



SQL Anywhere® 16

Änderungen und Upgrades

Version 16.0

Februar 2013

Version 16.0
Februar 2013

© 2013 SAP AG oder ein SAP-Konzernunternehmen. Alle Rechte vorbehalten.

Sie können diese Dokumentation (ganz oder teilweise) unter folgenden Bedingungen benutzen, reproduzieren und verteilen: 1) Sie müssen diese und alle anderen Urheberrechtsvermerke auf allen Kopien oder Auszügen der Dokumentation wiedergeben. 2) Sie dürfen die Dokumentation nicht verändern. 3) Sie dürfen nichts tun, aus dem abgeleitet werden könnte, dass Sie oder jemand anderer als SAP Verfasser oder Quelle der Dokumentation ist. Die hier enthaltenen Informationen können jederzeit ohne vorherigen Hinweis geändert werden.

Einige Softwareprodukte, die von der SAP AG oder einem ihrer Vertriebspartner vermarktet werden, enthalten Softwarekomponenten anderer Softwareanbieter. Die nationalen Produktspezifikationen können unterschiedlich sein.

Diese Dokumentationen werden von der SAP AG und ihren Tochtergesellschaften ("SAP Group") lediglich zu Informationszwecken bereitgestellt, ohne dass eine Gewährleistung oder eine Garantie irgendeiner Art gegeben wird. Die SAP Group übernimmt keine Verantwortung im Hinblick auf Fehler oder Auslassungen in den Dokumentationen. Die einzigen Garantien für Produkte und Dienstleistungen der SAP Group sind diejenigen, die in den mit den Produkten und Dienstleistungen eventuell gelieferten ausdrücklichen Garantieerklärungen enthalten sind. Keine der hier enthaltenen Informationen kann als Gewährung einer weitergehenden Garantie betrachtet werden.

SAP und weitere erwähnte SAP-Produkte und -Dienstleistungen sowie die entsprechenden Logos sind Marken oder eingetragene Marken der SAP AG in Deutschland und anderen Ländern. Weitere Hinweise finden Sie unter <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark>.

Inhalt

Über diese Dokumentation	vii
Neue Funktionen in Version 16.0	1
Produktweite neue Funktionen	1
Produktweite Verhaltensänderungen	7
Produktweite nicht mehr empfohlene und entfernte Funktionen	7
Neue Funktionen in SQL Anywhere	9
Verhaltensänderungen von SQL Anywhere	28
Nicht mehr empfohlene und nicht mehr weiterentwickelte SQL Anywhere-Funktionen	37
Neue MobiLink-Funktionen	40
Verhaltensänderungen von MobiLink	49
Neue Funktionen von QAnywhere	53
Neue Funktionen von Relay Server	53
Verhaltensänderungen von Relay Server	55
Neue Funktionen von UltraLite	56
Verhaltensänderungen und nicht mehr empfohlene Funktionen von UltraLite	65
Neue Funktionen der Administrationstools	67
Neue Funktionen in Version 12.0.1	73
12.0.1 Ergänzung: Neue Funktionen und Verhaltensänderungen	73
Neue Funktionen in SQL Anywhere	84
Performanceverbesserungen	90
Eigenschaften und Systemmonitorstatistiken	91
Verhaltensänderungen von SQL Anywhere	94
Nicht mehr empfohlene SQL Anywhere-Funktionen	96
Neue MobiLink-Funktionen	96
Verhaltensänderungen von MobiLink	101
Neue Funktionen in SQL Remote	104
Neue Funktionen von UltraLite	104
UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen	110

Neue Funktionen der Administrationstools	111
Erweiterungen der Dokumentation	116
Neue Funktionen in Version 12.0.0	117
Produktweite neue Funktionen	117
Produktweite Verhaltensänderungen	117
Neue Funktionen in SQL Anywhere	119
Verhaltensänderungen von SQL Anywhere	148
Nicht mehr empfohlene und nicht mehr weiterentwickelte SQL Anywhere-Funktionen	154
Neue MobiLink-Funktionen	156
Verhaltensänderungen von MobiLink	167
Neue Funktionen in SQL Remote	172
Neue Funktionen von UltraLite	173
UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen	177
Neue Funktionen von UltraLiteJ	180
Verhaltensänderungen von UltraLiteJ und nicht mehr empfohlene Funktionen	184
Neue Funktionen der Administrationstools	186
Erweiterungen der Dokumentation	196
Neue Funktionen in Version 11.0.1	199
Neue Funktionen in SQL Anywhere	199
Verhaltensänderungen von SQL Anywhere und nicht mehr empfohlene Funktionen	206
Neue MobiLink-Funktionen	210
MobiLink-Verhaltensänderungen und nicht mehr empfohlene Funktionen	212
Neue Funktionen von UltraLite	212
UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen	213
Neue Funktionen von UltraLiteJ	213
Verhaltensänderungen von UltraLiteJ und nicht mehr empfohlene Funktionen	214
Neue Funktionen des Administrationstools	214

Verhaltensänderungen des Administrationstools und nicht mehr empfohlene Funktionen	215
Produktweite neue Funktionen	216
Erweiterungen der Dokumentation	217
Neue Funktionen in Version 11.0.0	219
SQL Anywhere	220
MobiLink	263
SQL Remote	269
UltraLite	270
Sybase Central und Interactive SQL	276
Erweiterungen der Dokumentation	286
Produktweite Funktionen	287
Upgrade auf SQL Anywhere 16	289
Funktionen, die einen Neuaufbau (Entladen/Neuladen), ein Datenbank- Upgrade oder aktualisierte Clientbibliotheken erfordern	289
SQL Anywhere-Server-Upgrades	295
Vorsichtsmaßnahmen für Upgrade und Neuaufbau	297
Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind	298
Datenbanken neu aufbauen (entladen/neu laden)	298
Neuaufbauprozess für Datenbanken der Version 10 und höher	300
Neuaufbauprozess für Datenbanken der Version 9 und höher	303
Upgradevorgang für Datenbanken der Version 10 und später	310
Upgrade und Neuaufbau in einem Datenbankspiegelungssystem	314
Fehlerbehandlung: Datenbank-Upgrades	322
MobiLink-Upgrades	325
UltraLite-Upgrades	337
SQL Remote-Upgrades	341
Upgrade des SQL Anywhere-Monitors durchführen und Ressourcen migrieren	342
Index	347

Über diese Dokumentation

In diesem Handbuch werden die neuen Funktionen in SQL Anywhere 16 und in früheren Versionen der Software beschrieben.

Hinweise zu neuen Funktionen und Verhaltensänderungen in SQL Anywhere bis Version 10.0.1 finden Sie unter <http://dcx.sybase.com/html/dbwnde10/dbwnde10.html>.

Neue Funktionen in Version 16.0

Informationen über neue Funktionen und Verhaltensänderungen in den Versionen von SQL Anywhere vor der Version 10 finden Sie unter <http://dcx.sybase.com/html/dbwnde10/dbwnde10.html>.

Produktweite neue Funktionen

Nachstehend finden Sie eine Liste der produktweiten neuen Funktionen, die in Version 16.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Neues Sicherheitsmodell: RBAC (Role-Based Access Control - rollenbasierte Zugriffssteuerung)

In SQL Anywhere 16.0 wird ein neues rollen- und privilegbasiertes Sicherheitsmodell eingeführt, das an die Stelle des bisherigen Sicherheitsmodells tritt. Das neue rollenbasierte Sicherheitsmodell ermöglicht Ihnen eine genaue Kontrolle darüber, welche mit Privilegien verbundenen Aufgaben Benutzer ausführen können, und eine einfachere Verwaltung der Zugriffssteuerung.

Für vorhandene SQL Anywhere-Kunden, die ein Upgrade auf Version 16.0 durchführen, wurde Rückwärtskompatibilität bereitgestellt, damit die Anwendungen nach dem Upgrade der Datenbank weiterhin funktionieren. In einem speziellen Kapitel wird erläutert, wo die Unterschiede zwischen den Modellen liegen, was beim Upgrade automatisch geschieht und was Sie möglicherweise tun sollten, wenn Sie Anwendungen haben, die SQL Anywhere verwenden (z.B. Ihre Aufrufe für die Anweisungen GRANT und REVOKE aktualisieren). Siehe „Upgrade auf rollenbasierte Sicherheit“ [*SQL Anywhere Server - Datenbankadministration*].

Praktische Einführungen sind verfügbar unter „Praktische Einführung: Rollen und Privilegien erteilen (Sybase Central)“ [*SQL Anywhere Server - Datenbankadministration*] und „Praktische Einführung: Rollen und Privilegien erteilen (SQL)“ [*SQL Anywhere Server - Datenbankadministration*].

Im Folgenden finden Sie einen kurzen Überblick über die neuen und geänderten Funktionen zur Unterstützung des neuen rollenbasierten Sicherheitsmodells.

- **Überblick über das rollenbasierte Sicherheitsmodell** Vollständige Systemprivilegien und Rollen wurden hinzugefügt, um die Sicherheit zu erhöhen, indem Sie genau festlegen können, über welche Funktionen Ihre Benutzer verfügen sollen. Für jeden mit Privilegien verbundenen Vorgang, der im System durchgeführt werden kann, wurde ein Systemprivileg erstellt. Mit Rollen werden Privilegien zu logischen Gruppen kombiniert. SQL Anywhere bietet Ihnen zahlreiche vordefinierte Rollen, aber Sie können auch Ihre eigenen Rollen erstellen. Siehe „Benutzersicherheit (Rollen und Privilegien)“ [*SQL Anywhere Server - Datenbankadministration*].

Wenn Sie ein Upgrade vornehmen, werden Datenbankberechtigungen, Berechtigungen und Gruppen durch Rollen, Privilegien und benutzererweiterte Rollen ersetzt. Siehe „Upgrade auf rollenbasierte Sicherheit“ [*SQL Anywhere Server - Datenbankadministration*].

- **Neues TRUNCATE-Privileg auf Objektebene** Ein neues Privileg auf Objektebene, TRUNCATE, wurde hinzugefügt, damit ein Benutzer eine angegebene Tabelle oder materialisierte

Ansicht kürzen kann. Dieses Privileg gab es in früheren Versionen der Software nicht als Berechtigung auf Objektebene. Siehe „Objektprivilegien“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Neues LOAD-Privileg auf Objektebene** Ein neues Privileg auf Objektebene, LOAD, wurde hinzugefügt, damit ein Benutzer eine bestimmte Tabelle laden kann. Dieses Privileg gab es in früheren Versionen der Software nicht als Berechtigung auf Objektebene. Siehe „Objektprivilegien“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Automatisches Entsperren von Benutzerkonten** Mit den Login-Richtlinienoptionen root_auto_lock_time (nur Root-Login-Richtlinie) und auto_unlock_time können Sie festlegen, nach welcher Zeitspanne Benutzer, die aufgrund fehlgeschlagener Login-Versuche aus der Datenbank ausgesperrt wurden, automatisch entsperrt werden. Diese Optionen sind in den Anweisungen CREATE LOGIN POLICY und ALTER LOGIN POLICY verfügbar. Siehe „Automatisches Entsperren von Benutzerkonten“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Systemprozeduren** Die folgenden Systemprozeduren wurden hinzugefügt, um die rollenbasierte Sicherheit zu unterstützen:
 - **sp_objectpermission-Systemprozedur** Diese Prozedur generiert einen Bericht über die Objektprivilegien, die dem angegebenen Element (Objekt, DBSpace, Rolle oder Benutzername) erteilt wurden. Sie müssen vorhandene Datenbanken neu aufbauen, um diese Systemprozedur verwenden zu können. Siehe „sp_objectpermission-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_displayroles-Systemprozedur** Gibt alle Rollen zurück, die dem angegebenen Element (Systemprivileg, Systemrolle, benutzerdefinierte Rolle oder Benutzername) erteilt wurden, oder zeigt die gesamte Hierarchiestruktur der Rollen an. Siehe „sp_displayroles-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_has_role-Systemprozedur** Gibt die Angabe zurück, ob dem Aufrufer der Prozedur das angegebene Systemprivileg oder die angegebene benutzerdefinierte Rolle erteilt wurde. Siehe „sp_has_role-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_proc_priv-Systemprozedur** Gibt die Liste der Systemprivilegien zurück, die zum Ausführen einer Prozedur erforderlich sind. Siehe „sp_proc_priv-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_auth_sys_role_info-Systemprozedur** Gibt die Zuordnung von Datenbankberechtigungen aus früheren Versionen von SQL Anywhere zu den entsprechenden Kompatibilitätsrollen zurück. Siehe „sp_auth_sys_role_info-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_sys_priv_role_info-Systemprozedur** Gibt die Zuordnung von Systemprivilegien zu den zugrunde liegenden Systemrollen zurück. Siehe „sp_sys_priv_role_info-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Datenbankoptionen** Die folgenden Datenbankoptionen wurden hinzugefügt, um die rollenbasierte Sicherheit zu unterstützen:

- **min_role_admins-Option** Legt die minimale Anzahl von Administratoren fest, die für eine Rolle erforderlich ist. Siehe „min_role_admins-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **db_publisher-Option** Mit dieser Option wird die Benutzer-ID für den Publikationseigentümer der Datenbank gespeichert. Sie kann genauso festgelegt werden wie andere Datenbankoptionen, aber auch mit den Anweisungen GRANT PUBLISH und REVOKE PUBLISH. Siehe „db_publisher-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbankserveroptionen** Das Verhalten der folgenden Datenbankserveroptionen wurde geändert, um die rollenbasierte Sicherheit zu unterstützen:
 - **Datenbankserveroption -gu** Der Wert "DBA" bedeutet, dass nur ein Benutzer mit dem SERVER OPERATOR-Systemprivileg Datenbanken erstellen oder löschen kann. Siehe „Datenbankserveroption -gu“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Datenbankserveroption -gk** Der Wert "DBA" bedeutet, dass nur ein Benutzer mit dem SERVER OPERATOR-Systemprivileg einen Datenbankserver mit dem Dienstprogramm dbstop herunterfahren kann. Siehe „Datenbankserveroption -gk“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Datenbankserveroption -gd** Der Wert "DBA" bedeutet, dass nur ein Benutzer mit dem SERVER OPERATOR-Systemprivileg Datenbanken starten kann. Siehe „Datenbankserveroption -gd“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Datenbankserveroption -gl** Der Wert "DBA" bedeutet, dass nur ein Benutzer mit dem Privileg LOAD ANY TABLE oder ALTER ANY TABLE die LOAD-Anweisung ausführen kann. Der Benutzer muss das SELECT ANY TABLE-Privileg haben, um die UNLOAD-Anweisung ausführen zu können. Siehe „Datenbankserveroption -gl“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue und geänderte SQL-Anweisungen** Die folgenden SQL-Anweisungen wurden hinzugefügt oder geändert, um die rollenbasierte Sicherheit zu unterstützen:
 - **Änderungen der GRANT-Anweisung** Die GRANT-Anweisung wurde so erweitert, dass sie das Erteilen von Rollen und Privilegien ermöglicht. Siehe „GRANT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

 Die alte Syntax zum Erteilen von Datenbankberechtigungen, Berechtigungen und Mitgliedschaft in Gruppen wird zwar weiterhin unterstützt, aber nicht mehr empfohlen. Siehe „GRANT-Anweisung (Berechtigungen und Gruppen) (nicht mehr empfohlen)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Änderungen der REVOKE-Anweisung** Die REVOKE-Anweisung wurde so erweitert, dass sie das Entziehen von Rollen und Privilegien ermöglicht. Siehe „REVOKE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

 Die alte Syntax zum Erteilen von Datenbankberechtigungen, Berechtigungen und Mitgliedschaft in Gruppen wird zwar weiterhin unterstützt, aber nicht mehr empfohlen. Siehe „REVOKE-

Anweisung (Berechtigungen und Gruppen) (nicht mehr empfohlen)“ [\[SQL Anywhere Server - Datenbankadministration\]](#).

- **Neue CREATE ROLE-Anweisung** Erstellt oder ersetzt eine Rolle, konvertiert einen Benutzer in eine Rolle oder verwaltet Rollenadministratoren für eine Rolle. Siehe „[CREATE ROLE-Anweisung](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Neue ALTER ROLE-Anweisung** Migriert eine Kompatibilitätsrolle (eine der Rollen, die mit SYS_AUTH_ beginnen) in eine benutzerdefinierte Rolle und löscht dann die Kompatibilitätsrolle. Siehe „[ALTER ROLE-Anweisung](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Neue DROP ROLE-Anweisung** Entfernt eine benutzerdefinierte Rolle oder Kompatibilitätsrolle aus der Datenbank oder konvertiert eine benutzererweiterte Rolle in einen normalen Benutzer. Siehe „[DROP ROLE-Anweisung](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Neue GRANT ROLE SYS_RUN_REPLICATION_ROLE-Anweisung** Diese Anweisung erteilt die zum Ausführen der Replikation erforderliche Rolle. Siehe „[GRANT ROLE SYS_RUN_REPLICATION_ROLE-Anweisung](#) [\[MobiLink\]](#) [\[SQL Remote\]](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Neue GRANT ROLE SYS_REPLICATION_ADMIN_ROLE-Anweisung** Diese Anweisung erteilt die zum Verwalten der Replikation erforderliche Rolle. Siehe „[GRANT ROLE SYS_REPLICATION_ADMIN_ROLE-Anweisung](#) [\[MobiLink\]](#) [\[SQL Remote\]](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Änderungen der Anweisungen GRANT PUBLISH und REVOKE PUBLISH** Bisher wurden mit den Anweisungen GRANT PUBLISH und REVOKE PUBLISH die Publikationseigentümer-Informationen für eine Datenbank in der ISYSAUTHORITY-Systemtabelle aktualisiert. Im Rahmen der rollenbasierten Sicherheit wird jedoch die Benutzer-ID des Publikationseigentümers nun als db_publisher-Datenbankoption gespeichert. Diese Anweisungen aktualisieren nun den Wert der db_publisher-Datenbankoption, statt ISYSUSERAUTHORITY zu aktualisieren. Siehe „[GRANT PUBLISH-Anweisung](#) [\[SQL Remote\]](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und „[REVOKE PUBLISH-Anweisung](#) [\[SQL Remote\]](#)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Änderungen des Katalogs** Die im Folgenden beschriebenen Änderungen wurden am Katalog vorgenommen, um die rollenbasierte Sicherheit zu unterstützen. Die Änderungen werden an der Katalogtabelle vorgenommen, aber da Sie nur Zugriff auf die entsprechende Systemansicht haben, wird die Ansicht hier ebenfalls erwähnt.

Katalogelement	Änderung
ISYSROLEGRANT-Systemtabelle/„ SYSROLEGRANT-Systemansicht “ [SQL Anywhere Server - SQL-Referenzhandbuch]	Neu. Speichert Informationen zu Rollenmitgliedschaft und Typ der Mitgliedschaft.

Katalogelement	Änderung
„Konsolidierte Ansicht SYSROLEGRANTS“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Neu. Dasselbe wie SYSROLEGRANT, aber enthält zwei weitere Spalten: role_name und grantee_name.
ISYSROLEGRANTEXT-Systemtabelle/„SYSROLEGRANTEXT-Systemansicht“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Neu. Speichert die Syntaxerweiterungen für die Privilegien SET USER und CHANGE PASSWORD.
„SYSGROUP-Kompatibilitätsansicht“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Diese Ansicht war bisher eine Systemansicht und ist nun eine Kompatibilitätsansicht.
„SYSGROUPS-Kompatibilitätsansicht“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Diese Ansicht war bisher eine konsolidierte Ansicht und ist nun eine Kompatibilitätsansicht.
„SYSUSER-Systemansicht“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Neue-Spalte: dual_password
„SYSUSERAUTHORITY-Kompatibilitätsansicht (nicht mehr empfohlen)“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Diese Ansicht wurde von einer Systemansicht zu einer Kompatibilitätsansicht geändert. Die zugrunde liegende ISYSAUTHORITY-Systemtabelle wurde zwar entfernt, aber die Ansicht bleibt aus Kompatibilitätsgründen erhalten.
„SYSTABLEPERM-Systemansicht“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Neue Spalten: loadauth und truncateauth
„Konsolidierte Ansicht SYSTABAUTH“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>]	Neue Spalten: loadauth und truncateauth

- Änderungen am SQL Anywhere-Plug-In zur Unterstützung von Rollen und Privilegien**
An Sybase Central wurden viele Änderungen vorgenommen, um die rollenbasierte Sicherheit zu unterstützen. Beachten Sie, dass mit Privilegien verbundene Datenbankaufgaben, z.B. das Erstellen einer Tabelle, möglicherweise in Sybase Central mehr Privilegien erfordern als beim Ausführen über SQL-Anweisungen. Dies liegt daran, dass Sybase Central bei Verwendung zusätzliche mit Privilegien verbundene Aufgaben ausführt, z.B. das Auffüllen der Ordner (Ansichten, Benutzer usw.) für die Anzeige.

Wenn Sie nicht sicher sind, welche Privilegien für das Ausführen einer Aufgabe erforderlich sind, finden Sie in der Dokumentation weitere Hinweise zum Ausführen der Aufgabe in Sybase Central.

Unterstützung der LDAP-Benutzerauthentifizierung

SQL Anywhere bietet nun Unterstützung für die LDAP-Benutzerauthentifizierung. Die folgende Liste enthält Informationen zu den Funktionen, die hinzugefügt oder geändert wurden, um die LDAP-

Benutzerauthentifizierung zu unterstützen. Siehe „LDAP-Authentifizierung“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Änderungen der Root-Login-Richtlinie** Die folgenden Login-Richtlinienoptionen wurden zur Root-Login-Richtlinie hinzugefügt, um die LDAP-Benutzerauthentifizierung zu unterstützen:
 - ldap_primary_server
 - ldap_secondary_server
 - ldap_auto_failback_period
 - ldap_failover_to_std
 - ldap_refresh_dn

Siehe „Root-Login-Richtlinie“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Systemprozeduren** Die folgenden Systemprozeduren wurden hinzugefügt oder erweitert, um die LDAP-Benutzerauthentifizierung zu unterstützen:
 - „sa_get_ldapserver_status-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] (neu)
 - „sa_get_user_status-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] (erweitert)

Neue Spalten (user_dn, user_dn_cached_at, password_change_state, password_change_first_user, password_change_second_user) werden durch die sa_get_user_status-Systemprozedur gemeldet.

- **Datenbankserver- und Datenbankoptionen** Die folgenden Datenbankoptionen wurden hinzugefügt oder erweitert, um die LDAP-Benutzerauthentifizierung zu unterstützen:
 - „login_mode-Option“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „trusted_certificates_file-Option“ [[SQL Anywhere Server - Datenbankadministration](#)]
- **SQL-Anweisungen** Die folgenden SQL-Anweisungen wurden hinzugefügt, um die LDAP-Benutzerauthentifizierung zu unterstützen:
 - „CREATE LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „ALTER LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „VALIDATE LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Die folgenden SQL-Anweisungen wurden erweitert, um die LDAP-Benutzerauthentifizierung zu unterstützen:

- „ALTER USER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] (neue REFRESH DN-Syntax)
- „CREATE LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] (neue Login-Richtlinienoptionen)
- „ALTER LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] (neue Login-Richtlinienoptionen)
- „COMMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] (Möglichkeit zum Kommentieren von LDAP-Server-Konfigurationsobjekten)

- **Katalogänderungen** Die folgenden Änderungen wurden am Katalog vorgenommen, um die LDAP-Benutzerauthentifizierung zu unterstützen:
 - **SYSLDAPSERVER-Systemansicht (neu)** Die SYSLDAPSERVER-Systemansicht enthält eine Zeile für jedes in der Datenbank konfigurierte LDAP SERVER-Objekt. Siehe „SYSLDAPSERVER-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **SYSUSER-Systemansicht** Die SYSUSER-Systemansicht hat zwei neue Spalten für die LDAP-Benutzerauthentifizierung: `user_dn` und `user_dn_cached_at`. Siehe „SYSUSER-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Produktweite Verhaltensänderungen

Nachstehend finden Sie eine Liste der neuen, produktweiten Verhaltensänderungen, die in Version 16.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Weitere produktweite Verhaltensänderungen

Beispielzertifikate befinden sich in einem neuen Verzeichnis X.509-Beispielzertifikate für die TLS-Kommunikation wurden aus den Verzeichnissen *bin32* und *bin64* in das Verzeichnis `C:\Users\Public\Documents\SQL Anywhere 16\Samples\Certificates` verschoben.

Die folgenden Dateien wurden ebenfalls zu dem neuen Verzeichnis hinzugefügt:

- *rsacient.id*
- *rsacient_nopwd.id*
- *rsaroot.key*
- *rsaserver_nopwd.id*

Weitere Hinweise finden Sie in der Datei `C:\Users\Public\Documents\SQL Anywhere 16\Samples\Certificates\readme.txt`.

SQL Anywhere unterstützt TLS-Version 1.1 Server- und Clientbibliotheken für SQL Anywhere unterstützen nun TLS-Version 1.1 sowohl für die TLS-Verbindungsverschlüsselung als auch für HTTPS. TLS-Version 1.1 wird unter Mac OS X nicht unterstützt.

iAnywhere Solutions Oracle-ODBC-Treiber Für Version 16.0 lautet der Name des SQL Anywhere Oracle-ODBC-Treibers **SQL Anywhere 16 - Oracle**. Die vorherige Version hieß **iAnywhere Solutions 12 - Oracle**. Für Referenzzwecke lautet der Name des SQL Anywhere ODBC-Treibers **SQL Anywhere 16**.

Produktweite nicht mehr empfohlene und entfernte Funktionen

- **Elliptische Kurvenverschlüsselung (ECC)** Die Unterstützung für die ECC-Verschlüsselung wurde entfernt. Diese Änderung hat Auswirkungen auf die folgenden Funktionen:

- Die IsEccAvailable-Servereigenschaft wurde entfernt.
- Die Datenbankserveroption -ec akzeptiert nicht mehr "ECC" für die TLS_TYPE-Protokolloption. Die TLS_TYPE-Protokolloption für die Datenbankserveroption -ec wurde aus der Dokumentation entfernt. Diese Option wird jedoch aus Gründen der Rückwärtskompatibilität weiterhin von der Software unterstützt.
- Der ENCRYPTION-Verbindungsparameter akzeptiert nicht mehr "ECC" für das TLS_TYPE-Argument. Das TLS_TYPE-Argument für die ENCRYPTION-Verbindung wurde aus der Dokumentation entfernt. Diese Option wird jedoch aus Gründen der Rückwärtskompatibilität weiterhin von der Software unterstützt.
- Das Dienstprogramm zur Zertifikatserstellung (createcert) akzeptiert die Option -ec nicht mehr und die Option -t akzeptiert "ecc" für den Verschlüsselungstyp nicht mehr.
- Das Schlüsselpaargenerator-Dienstprogramm (createkey) erstellt nur RSA-Schlüsselpaare. Die Eingabeaufforderungen **Verschlüsselungstyp wählen** und **ECC-Kurve angeben** wurden entfernt.
- Der SQL Anywhere-Monitor unterstützt keine ECC-Verschlüsselung. Wenn Sie einen MobiLink-Server vor Version 16 überwachen möchten, der ECC-Verschlüsselung verwendet, müssen Sie Folgendes durchführen:
 - Starten Sie den MobiLink-Server mit einer zweiten Gruppe von Netzwerkprotokolloptionen, die kein ECC verwendet.
 - Fügen Sie den MobiLink-Server als zu überwachende Ressource hinzu. Wenn Sie dazu aufgefordert werden, die Portnummer für die MobiLink-Serverressource einzugeben, geben Sie die Nummer an, der keine ECC-Verschlüsselung zugeordnet ist.
- Die MobiLink-Protokolloption tls_type akzeptiert nicht mehr "ECC" für die tls_type-Option. Die MobiLink-Protokolloption tls_type wurde aus der Dokumentation entfernt. Diese Option wird jedoch weiterhin von der Software unterstützt.
- Die MobiLink-Clientprotokolloption e2ee_type akzeptiert nicht mehr "ECC" für die tls_type-Option. Die MobiLink-Clientprotokolloption e2ee_type wurde aus der Dokumentation entfernt. Diese Option wird jedoch weiterhin von der Software unterstützt.
- Im Outbound Enabler kann für die tls_type-Option nicht mehr "ECC" gewählt werden. Die tls_type-Protokolloption wurde aus der Dokumentation entfernt.
- API-Methoden und -Eigenschaften für UltraLite C/C++ und UltraLiteJ im Zusammenhang mit der ECC-Konfiguration wurden entfernt.
- **Spaltennamen senden** In MobiLink und UltraLite werden Spaltennamen nun immer während der Synchronisation gesendet. Versuche, das Senden von Spaltennamen zu deaktivieren, werden ignoriert. Diese Änderung hat Auswirkungen auf die folgenden Funktionen:
 - In MobiLink wird die erweiterte Option SendColumnNames (scn) für dbmlsync nicht mehr empfohlen und wird ignoriert, wenn sie auf OFF gesetzt ist. Die erweiterte Option SendColumnNames wurde aus der Dokumentation entfernt. Diese Option wird jedoch weiterhin von der Software unterstützt.

- UltraLite- und UltraLiteJ-Clients senden bei der Synchronisation mit einer MobiLink-Serverdatenbank immer Spaltennamen. Methoden und Vorgehensweisen, mit denen das Senden von Spaltennamen gestoppt werden soll, werden nun ignoriert und nicht mehr empfohlen. Die folgenden Mitglieder und Parameter sind davon betroffen:

- send_column_names-Synchronisationsparameter
- Synchronisationsprofiloptionen
- SyncParms.setSendColumnNames-Methode [UltraLiteJ]
- SyncParms.getSendColumnNames-Methode [UltraLiteJ]
- ULSyncParms.SendColumnNames-Eigenschaft [UltraLite.NET]
- send_column_names in der ul_sync_info-Struktur [UltraLite C- und Embedded SQL-Datentypen]

Neue Funktionen in SQL Anywhere

Im Folgenden finden Sie eine Liste neuer Funktionen in SQL Anywhere Version 16.0.

Weitere Hinweise zu den Änderungen der unterstützten Plattformen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Wichtigste Funktionen

Im Folgenden finden Sie eine Liste der wichtigsten Funktionen, die in SQL Anywhere Version 16.0 eingeführt wurden.

Performanceverbesserungen

Der Optimierer in SQL Anywhere wurde erweitert und berücksichtigt nun während der Abfrageoptimierung Zugriffspläne, die keine links-tiefen Bäume sind. Die Berücksichtigung zusätzlicher Zugriffspläne führt dazu, dass der Optimierer einen effizienteren besten Plan findet und dadurch die Laufzeit einiger Anweisungen in SQL Anywhere 16.0 erheblich verkürzt wird. Siehe „[Komponenten von Ausführungsplänen](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Unterstützung der Datentypen ROW und ARRAY

In SQL Anywhere wurde Unterstützung für die zusammengesetzten Datentypen ROW und ARRAY hinzugefügt. Diese Datentypen stellen eine effizientere Methode zum Speichern von Listen dar, weil sie die Möglichkeit bieten, Struktur und Datentyp ihrer Werte zu definieren. Außerdem erleichtern sie den Zugriff auf Listenelemente, entweder direkt mit doppelten eckigen Klammern oder als Ergebnismenge mit dem UNNEST-Operator. Ziehen Sie den ARRAY-Datentyp in Erwägung, wenn Sie Listen als begrenzte Zeichenfolgen in VARCHAR-Spalten speichern und bei der syntaktischen Analyse sa_split_list verwenden. ARRAY-Daten sind sehr hilfreich, wenn Sie verschiedene Objekte speichern möchten, die miteinander verknüpft sind. ROW-Daten sind hilfreich, wenn Sie mehrere Werte für ein Objekt speichern möchten.

Neue SQL-Funktionen wurden hinzugefügt, um zusammengesetzte Datentypen zu unterstützen. Dazu gehören ein ROW-Konstruktor und ein ARRAY-Konstruktor. Ein neuer Operator, UNNEST, wurde ebenfalls hinzugefügt.

SQL-Funktionen und -Prozeduren akzeptieren ROW- und ARRAY-Daten als IN-, OUT- oder INOUT-Argumente. Zeilen und Arrays können als Rückgabetyt einer benutzerdefinierten SQL-Funktion verwendet werden.

ROW- und ARRAY-Daten können in folgenden Kontexten verwendet werden:

- Anweisungen SELECT, INSERT, UPDATE, MERGE und DELETE
- Gespeicherte Transact-SQL- und Watcom-SQL-Prozeduren
- Funktionen
- Trigger
- Batches

ROW, ARRAY, VARRAY und UNNEST sind nun reservierte Wörter.

Siehe:

- „Zusammengesetzte Datentypen“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „Vergleiche von zusammengesetzten Typen“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „Array-Operatoren“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ARRAY-Konstruktor [zusammengesetzt]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ROW-Konstruktor [zusammengesetzt]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ARRAY_AGG-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ARRAY_MAX_CARDINALITY-Funktion [zusammengesetzt]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CARDINALITY-Funktion [zusammengesetzt]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „TRIM_ARRAY-Funktion [zusammengesetzt]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „Array-Operator UNNEST“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

OData-Unterstützung

SQL Anywhere umfasst nun einen OData Server, über den Webclients mit einem SQL Anywhere-Datenbankserver kommunizieren können. Webclients können OData-Anforderungen an einen konfigurierbaren OData Producer senden. Dieser wird auf einem HTTP-Server gehostet, der OData-Konzepte in relationale Datenbankvorgänge konvertiert. Die folgenden Funktionen wurden im Rahmen der OData-Unterstützung hinzugefügt oder geändert:

- Konformität mit OData-Protokollversion 2
- OData Producer
- Eingebetteter Jetty WebServer (mit Unterstützung für alternative HTTP-Server)
- OData-Serverdienstprogramm
- OData-Dienstprogramm Serverstopp
- Im Dienstprogramm für Dienste (dbsvc) können OData-Windows-Dienste erstellt werden.
- Das SQL Anywhere-Plug-In unterstützt die Verwaltung von OData-Diensten.

Siehe „[OData-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].

Unterstützung der Ereignisprotokollierung

Bei der Ereignisprotokollierung werden Informationen zu systemdefinierten und benutzerdefinierten Trace-Ereignissen in einem Ereignisprotokollierungsziel erfasst. Eine Trace-Sitzung besteht aus Trace-Ereignissen (bestimmten Punkten in der Datenbankserver-Software oder Ihrer SQL-Anwendung), deren

Informationen gesammelt und in einem Ziel protokolliert werden. Ziele sind der Speicherort (z.B. eine Datei), an dem der Datenbankserver Trace-Ereignisse protokolliert.

Die Ereignisprotokollierung wird für Produktionsumgebungen empfohlen und bietet detaillierte Steuerungsmöglichkeiten hinsichtlich der protokollierten Informationen. Sie können benutzer- und systemdefinierte Trace-Ereignisse sowohl für den Datenbankserver als auch für Ihre Anwendung protokollieren und die Trace-Ereignisse so anpassen, dass sich Performanceprobleme ermitteln lassen.

Mit der gesicherten Funktion `trace_system_event` können Sie steuern, ob benutzerdefinierte Trace-Ereignisse erstellt werden können.

Siehe:

- „Ereignisprotokollierung“ [*SQL Anywhere Server - Datenbankadministration*]
- „`sp_trace_events`-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „`sp_trace_event_fields`-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „`sp_trace_event_sessions`-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „`sp_trace_event_session_events`-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „`sp_trace_event_session_targets`-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „`sp_trace_event_session_target_options`-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „CREATE TEMPORARY TRACE EVENT-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „CREATE TEMPORARY TRACE EVENT SESSION-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „ALTER TRACE EVENT SESSION-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „DROP TRACE EVENT-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „DROP TRACE EVENT SESSION-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „NOTIFY TRACE EVENT-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „Dienstprogramm für die Verwaltung von ETD-Dateien (`dbmanageetd`)“ [*SQL Anywhere Server - Datenbankadministration*]
- „Gesicherte Funktionen“ [*SQL Anywhere Server - Datenbankadministration*]

SAP HANA-Unterstützung

Interactive SQL unterstützt Verbindungen mit SAP HANA-Datenbanken Sie können Interactive SQL verwenden, um eine Verbindung mit einer SAP HANA-Datenbank herzustellen. Klicken Sie im Fenster **Verbinden** auf **Datenbanktyp ändern** und klicken Sie dann auf **SAP HANA**. Siehe „Interactive SQL“ [*SQL Anywhere Server - Datenbankadministration*].

HANAODBC-Serverklasse für den Ferndatenzugriff Damit Sie SAP HANA als Backend-Server verwenden können, wurde die HANAODBC-Klasse in die Unterstützung für den Ferndatenzugriff einbezogen. Siehe „Serverklassen für den Ferndatenzugriff“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Erweiterungen von Datenbankspiegelung und Scale-Out mit Schreibschutz

Ereignisse können auf dem Spiegelserver oder auf einem Kopieknotten ausgeführt werden Ereignisse können nun in Spiegelungssystemen und in Scale-Out-Systemen mit Schreibschutz auf dem Spiegelserver und auf den Kopieknotten ausgeführt werden. Wenn Sie ein Ereignis erstellen

möchten, das auf einem beliebigen Server ausgeführt werden kann, geben Sie die FOR ALL-Klausel mit der CREATE EVENT-Anweisung oder der ALTER EVENT-Anweisung an. Siehe „[CREATE EVENT-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[ALTER EVENT-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Neue MIRROR-Serverklasse für den Ferndatenzugriff Die MIRROR-Serverklasse stellt eine Verbindung mit einem entfernten SQL Anywhere-Server über ODBC her. Beim Erstellen des Fremdservers enthält jedoch die USING-Klausel einen Spiegelservernamen aus der SYS.SYSMIRRORSERVER-Katalogtabelle. Siehe „[MIRROR-Serverklasse](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Den Arbiterserver in eine laufende Spiegelungskonfiguration verschieben Siehe „[Praktische Einführung: Arbiterserver verschieben](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Den Spiegelserver in einen Kopieknotten konvertieren Siehe „[Praktische Einführung: Partnerserver in Kopieknotten konvertieren](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Partner von einem Server auf einen anderen Server verschieben Siehe „[Praktische Einführung: Partnerserver verschieben](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Server können Kopieknotten und Arbiterserver für dieselbe Datenbank sein Siehe „[Praktische Einführung: Server sowohl als Kopieknotten als auch als Arbiterserver verwenden](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Spiegelungsdatenbanken oder Scale-Out-Datenbanken mit Schreibschutz auf laufenden Servern dynamisch starten Verwenden Sie die START DATABASE-Anweisung mit der MIRROR ON-Klausel, um eine Spiegeldatenbank oder einen Kopieknotten auf einem laufenden Partner zu starten. Siehe „[START DATABASE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SET MIRROR OPTION-Anweisung Die SET MIRROR OPTION-Anweisung verfügt über eine neue Option, promotion_time, mit der Sie angeben können, wie lang ein Kopieknotten nach dem Verlust einer Verbindung zum übergeordneten Knoten mit dem Stammdatenbankserver verbunden bleiben soll, bevor er sich selbst zum übergeordneten Knoten macht. Die max_disconnected_time-Option legt nun fest, wie viel Zeit seit der letzten Verbindung des Kopieknottes mit dem übergeordneten Knoten, dem alternativen übergeordneten Knoten oder der Stammdatenbank vergangen sein muss, bevor der Kopieknotten gestoppt wird. Siehe „[SET MIRROR OPTION-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Erweiterungen von Datenbankeigenschaften

CopyNodeParent Gibt den Namen des aktuellen übergeordneten Servers für einen Kopieknotten in einer Scale-Out-Konfiguration mit Schreibschutz zurück. Siehe [CopyNodeParent-Datenbankeigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].

PartnerState Wenn kein Partner definiert ist, gibt PartnerState den Wert NULL zurück. Bisher wurde der Wert "disconnected" zurückgegeben. Siehe [PartnerState-Datenbankeigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].

Neue Standardklausel WITH ROW LOGGING für LOAD TABLE Die WITH ROW LOGGING-Klausel bewirkt, dass jede geladene Zeile im Transaktionslog als INSERT-Anweisung erfasst wird. Diese Protokollierungsstufe wird für Datenbanken in einer Synchronisationsumgebung empfohlen und ist nun

der Standardwert bei Verwendung des Ausdrucks "FROM Dateiname" oder "USING FILE Dateiname" in einer gespiegelten Datenbank. Wenn Sie jedoch große Datenmengen laden, kann dieser Protokollierungstyp die Performance beeinträchtigen und erzeugt ein wesentlich längeres Transaktionslog.

Erweiterungen der Verwaltung

- **Datenbanken werden nach Upgrades neu gestartet** Standardmäßig wird die Datenbank nun nach einem Upgrade gestoppt und neu gestartet. Wenn Sie die Datenbank nach einem Upgrade stoppen und nicht neu starten möchten, verwenden Sie die RESTART OFF-Klausel der ALTER DATABASE-Anweisung oder die Option -nrs des Dienstprogramms zum Upgrade (dbupgrad). Nach erfolgreichem Upgrade einer Datenbank wird das Transaktionslog umbenannt. Siehe „[ALTER DATABASE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*] und „[Dienstprogramm zum Upgrade \(dbupgrad\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- **Eigentümer von Tabellen ändern** Sie können nun den Eigentümer einer Tabelle mit der ALTER TABLE-Anweisung ändern. Beim Ändern eines Tabelleneigentümers haben Sie die Möglichkeit, explizit erteilte Privilegien sowie Fremdschlüssel entweder beizubehalten oder zu löschen. Siehe „[ALTER TABLE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Kennwörter für die Doppelkontrolle** Die Login-Richtlinienoption change_password_dual_control erfordert, dass beim Ändern des Kennworts für einen Zielbenutzer zwei Administratoren beteiligt sein müssen. Die erste Benutzer gibt den ersten Teil eines Kennworts an, die zweite Person den zweiten Teil. Kennwortänderungen mit Doppelkontrolle verhindern, dass ein Kennwortadministrator das vollständige Kennwort eines anderen Benutzers kennt.

In der Login-Richtlinie des Zielbenutzers muss die change_password_dual_control-Option aktiviert sein.

Die durch die verify_password_function-Datenbankoption angegebene Funktion wird nicht für Benutzer aufgerufen, in deren Login-Richtlinie die change_password_dual_control-Option aktiviert ist.

- „[Kennwörter für die Doppelkontrolle](#)“ [*SQL Anywhere Server - Datenbankadministration*]
- „[verify_password_function-Option](#)“ [*SQL Anywhere Server - Datenbankadministration*]
- „[sa_verify_password-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]

Datenbankverbindungen

Die folgende Liste enthält die Erweiterungen für die Datenbankverbindungen in SQL Anywhere Version 16.0.

- **MaxRequestVars-Protokolloption (MAXVARS)** Diese Protokolloption legt die maximale Anzahl von HTTP-Eingabevariablen fest, die in Anforderungen an den Datenbankserver zulässig sind. Siehe „[MaxRequestVars-Protokolloption \(MAXVARS\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **ODBC-Verbindungsparameter SuppressInfoForDataTypes** Dieser Verbindungsparameter verhindert, dass der ODBC-Treiber Informationen zu den angegebenen Datentypen zurückgibt. Siehe „ODBC-Verbindungsparameter“ [[SQL Anywhere Server - Datenbankadministration](#)].

Sicherung und Wiederherstellung

Die folgende Liste enthält alle Erweiterungen zur Sicherung und Wiederherstellung in SQL Anywhere Version 16.0.

- **Erweiterungen des Sicherungsdienstprogramms (dbbackup)** Die folgenden Optionen wurden zum Sicherungsdienstprogramm hinzugefügt. Alle diese Optionen müssen mit der Option -s verwendet werden. Siehe „Sicherungsdienstprogramm (dbbackup)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Option	Beschreibung
-aw[-]	Aktiviert die automatische Optimierung von Verfassern. Durch Angeben von -aw wird die automatische Optimierung von Verfassern deaktiviert.
-bc <i>comment</i>	Erfasst einen Kommentar in der Sicherungsverlaufsdatei.
-h[-]	Aktiviert den Sicherungsverlauf. Durch Angeben von -h wird der Sicherungsverlauf deaktiviert.
-wa	Wartet den Abschluss von Transaktionen ab, bevor das Transaktionslog umbenannt oder gekürzt wird.
-wb	Verzögert die Sicherung der Datenbank, bis keine aktiven Transaktionen mehr vorhanden sind.

Sicherheit

Die folgende Liste enthält alle Erweiterungen zur Sicherheit in SQL Anywhere Version 16.0.

- **RAW-Verschlüsselung** Die ENCRYPT-Funktion kann Daten im Datenbankserver verschlüsseln und anschließend in ein RAW-Format ausgeben, damit sie außerhalb der Datenbank entschlüsselt werden können. Die DECRYPT-Funktion kann Daten entschlüsseln, die außerhalb des Datenbankservers verschlüsselt wurden. Siehe „ENCRYPT-Funktion [Zeichenfolge]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „DECRYPT-Funktion [Zeichenfolge]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Unterstützung für TDS RSA-Verschlüsselung mit Nonce-Kennwort austausch** Die neue Datenbankserveroption -tdsl beschränkt den TDS-Login-Anforderungstyp, den ein Datenbankserver unterstützt. Legen Sie den TDS-Login-Modus so fest, dass alle Login-Anforderungen, nur RSA-Login-Anforderungen oder nur RSA-Login-Anforderungen mit Nonce unterstützt werden. Siehe „Datenbankserveroption -tdsl“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Gesicherte Funktionen**

- **Schlüssel für gesicherte Funktion können nun angepasst werden** Sie können nun benutzerdefinierte Schlüssel für gesicherte Funktionen erstellen und anzeigen, die mithilfe der folgenden Systemprozeduren einzelnen Benutzern zugeordnet werden können:
 - **sp_create_secure_feature_key-Systemprozedur** Erstellt einen neuen Schlüssel für gesicherte Funktionen. Siehe „[sp_create_secure_feature_key-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
 - **sp_alter_secure_feature_key-Systemprozedur** Ändert einen zuvor definierten Schlüssel für gesicherte Funktionen, indem der Autorisierungsschlüssel bzw. die Funktionsliste geändert wird. Siehe „[sp_alter_secure_feature_key-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
 - **sp_drop_secure_feature_key-Systemprozedur** Löscht einen Schlüssel für gesicherte Funktionen. Siehe „[sp_drop_secure_feature_key-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
 - **sp_list_secure_feature_keys-Systemprozedur** Gibt eine Liste der definierten Schlüssel für gesicherte Funktionen zurück. Siehe „[sp_list_secure_feature_keys-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
 - **sp_use_secure_feature_key-Systemprozedur** Ermöglicht den Zugriff auf die gesicherten Funktionen, die dem angegebenen Schlüssel für gesicherte Funktionen zugeordnet sind. Siehe „[sp_use_secure_feature_key-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Gesicherte Funktionen** Sie können verhindern, dass Benutzer Verzeichnisse und Dateien auf dem Computer bearbeiten, der auch als Server fungiert. Die folgenden Funktionen, die Systemprozeduren entsprechen, wurden der **local_io**-Funktionsgruppe hinzugefügt:
 - sp_list_directory
 - sp_create_directory
 - sp_copy_directory
 - sp_move_directory
 - sp_delete_directory
 - sp_copy_file
 - sp_move_file
 - sp_delete_file

Die manage_server-Funktionsgruppe hindert Benutzer am Zugriff auf Funktionen im Zusammenhang mit dem Datenbankserver.

Die Funktionen create_trace_file und _trace_system_event verhindern, dass Benutzer Ereignisprotokollierungsziele bzw. benutzerdefinierte Ereignisse erstellen.
- **SQL-Code erneut ausführen, nachdem der Schlüssel des Datenbankservers für gesicherte Funktionen angegeben wurde** Wenn Sie in Sybase Central versuchen, SQL-Code auszuführen, der eine gesicherte Funktion verwendet, erhalten Sie die Möglichkeit, den

Schlüssel des Datenbankservers für gesicherte Funktionen anzugeben. Anschließend wird der SQL-Code erneut ausgeführt. Der Datenbankserver muss mit einem Schlüssel für gesicherte Funktionen (mit der Option -sk) gestartet werden, damit diese Funktion verwendet werden kann.

- **Sandboxing** Die Sandboxing-Funktion beschränkt Dateivorgänge mit Lese-/Schreibzugriff aus der Datenbank auf das Verzeichnis, in dem sich die Hauptdatenbankdatei befindet.

Wenn Sandboxing aktiviert ist, werden relative Pfadnamen als relativ zu dem Verzeichnis behandelt, in dem sich die Hauptdatenbankdatei befindet. Wenn Sandboxing nicht aktiviert ist, sind relative Pfadnamen relativ zum Arbeitsverzeichnis des Datenbankservers. Siehe „[Verhaltensänderungen von SQL Anywhere](#)“ auf Seite 28.

Die folgenden Funktionen wurden hinzugefügt, um die Sandboxing-Funktion zu unterstützen:

Funktion	Beschreibung
Datenbank-serveroptionen (dbsrv16)	<ul style="list-style-type: none"> ○ Datenbankserver- und Datenbankoption -sbx Diese Option legt das standardmäßige Sandboxing-Verhalten für alle Datenbanken fest, die auf dem Datenbankserver ausgeführt werden, oder für eine einzelne Datenbank. Siehe „Datenbankserveroption -sbx“ [SQL Anywhere Server - Datenbankadministration] und „Datenbankoption -sbx“ [SQL Anywhere Server - Datenbankadministration].
Datenbankoptionen	<ul style="list-style-type: none"> ○ disk_sandbox-Datenbankoption Diese Option steuert, ob Dateivorgänge der Datenbank mit Lese- und Schreibzugriff auf das Verzeichnis beschränkt werden, in dem sich die Hauptdatenbankdatei befindet. Siehe „disk_sandbox-Option“ [SQL Anywhere Server - Datenbankadministration].
Anweisungen	<ul style="list-style-type: none"> ○ START DATABASE-Anweisung Geben Sie beim Ausführen der START DATABASE-Anweisung die DISKSANDBOX-Klausel an, wenn Sie Dateivorgänge der Datenbank mit Lese- und Schreibzugriff auf das Verzeichnis beschränken möchten, in dem sich die Hauptdatenbankdatei befindet. Siehe „START DATABASE-Anweisung“ [SQL Anywhere Server - SQL-Referenzhandbuch].
Gesicherte Funktionen	<ul style="list-style-type: none"> ○ Gesicherte Funktion manage_disk_sandbox Verwenden Sie die Datenbankserveroption -sf, um die gesicherte Funktion manage_disk_sandbox zu steuern. Die gesicherte Funktion manage_disk_sandbox verhindert, dass Benutzer Sandboxing-Einstellungen ändern. Standardmäßig ist die gesicherte Funktion manage_disk_sandbox gesichert. Siehe „Datenbankserveroption -sf“ [SQL Anywhere Server - Datenbankadministration]. ○ Gesicherte Funktion disk_sandbox Verwenden Sie die Datenbankserveroption -sf, um die gesicherte Funktion disk_sandbox zu steuern. Die gesicherte Funktion disk_sandbox gibt verbundenen Benutzern die Möglichkeit, Sandboxing für ihre Verbindung zu aktivieren bzw. zu deaktivieren. Standardmäßig ist die gesicherte Funktion disk_sandbox gesichert. Siehe „Sandboxing“ [SQL Anywhere Server - Datenbankadministration].

Funktion	Beschreibung
Systemprozeduren	<p>Wenn Sie die Systemprozedur <code>sa_server_option</code> oder <code>sa_db_option</code> verwenden möchten, um Sandboxing-Einstellungen zu ändern, geben Sie den Schlüssel für die gesicherte Funktion <code>manage_disk_sandbox</code> an.</p> <ul style="list-style-type: none"> ○ sa_server_option-Systemprozedur Ändert das standardmäßige Sandboxing-Verhalten des Datenbankservers, während der Datenbankserver läuft. Siehe „sa_server_option-Systemprozedur“ [SQL Anywhere Server - SQL-Referenzhandbuch]. ○ sa_db_option-Systemprozedur Ändert das standardmäßige Sandboxing-Verhalten der Datenbank, während die Datenbank läuft. Siehe „sa_db_option-Systemprozedur“ [SQL Anywhere Server - SQL-Referenzhandbuch].
Eigenschaft	<ul style="list-style-type: none"> ○ disk_sandbox-Verbindungseigenschaft Gibt die Angabe zurück, ob die Sandboxing-Funktion für die Benutzerverbindung auf ON oder OFF gesetzt ist. Siehe disk_sandbox-Verbindungseigenschaft [SQL Anywhere Server - Datenbankadministration]. ○ DiskSandbox-Datenbankservereigenschaft Gibt die Angabe zurück, ob die Sandboxing-Funktion für den Datenbankserver auf ON oder OFF gesetzt ist. Siehe DiskSandbox-Servereigenschaft [SQL Anywhere Server - Datenbankadministration]. ○ DiskSandbox-Datenbankeigenschaft Gibt die Angabe zurück, ob die Sandboxing-Funktion für die Datenbank auf ON oder OFF gesetzt ist. Siehe DiskSandbox-Datenbankeigenschaft [SQL Anywhere Server - Datenbankadministration].

Datenbank-Dienstprogramme

Die folgende Liste enthält die Erweiterungen der Datenbank-Dienstprogramme in SQL Anywhere Version 16.0.

- **Dienstprogramm für Serverstart im Hintergrund (dbspawn)** Bei Verwendung von `dbspawn` zum Starten eines Datenbankservers mit einem Namen, der nicht eindeutig ist, wird ein neuer Fehler- und Exit-Code (`EXIT_SERVER_NAME_IN_USE`) zurückgegeben. Siehe „Exit-Codes der Softwarekomponenten“ [[SQL Anywhere Server - Programmierung](#)].
- **Dienstprogramm zum Entladen (dbunload)** Mit der Option `-ss` können Sie die Generierung von Spaltenstatistiken in der Reload-SQL-Skriptdatei unterdrücken. Siehe „Dienstprogramm zum Entladen (dbunload)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Dienstprogramm für die Zertifikatserstellung (createcert) und Dienstprogramm zum Anzeigen von Zertifikaten (viewcert)**

- Mit der Option `-p1` für `createcert` und `viewcert` können Sie angeben, dass PKCS #1 zum Kodieren unverschlüsselter privater RSA-Schlüssel verwendet werden soll.
- Mit der Option `-sa` für `createcert` können Sie angeben, welcher Signaturalgorithmus beim Erstellen oder Signieren von Zertifikaten verwendet werden soll.
- Durch Angeben der Option `@data` für `viewcert` können Sie `viewcert` mit den Befehlszeilenspezifikationen ausführen, die in einer Konfigurationsdatei oder Umgebungsvariablen gespeichert sind.

Siehe „Dienstprogramm zum Erstellen von Zertifikaten (`createcert`)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Dienstprogramm zum Anzeigen von Zertifikaten (`viewcert`)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbankoptionen

Die folgende Liste enthält die Erweiterungen der Datenbankoptionen in SQL Anywhere Version 16.0.

- **auto_commit_on_create_local_temp_index-Option** Diese Option steuert, ob der Datenbankserver eine COMMIT-Anweisung ausführt, bevor ein Index in einer lokalen temporären Tabelle erstellt wird. Sie müssen ein Upgrade bestehender Datenbanken durchführen, um diese Funktion verwenden zu können. Siehe „`auto_commit_on_create_local_temp_index-Option`“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **extern_login_credentials-Option** Steuert, ob Verbindungen mit entfernten Datenbanken mithilfe der externen Anmeldeinformationen des angemeldeten Benutzers oder der externen Anmeldeinformationen des effektiven Benutzers hergestellt werden. Siehe „`extern_login_credentials-Option`“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **webservice_sessionid_name-Option** Diese Option legt neu fest, anhand welcher Informationen der SQL Anywhere-Webserver ermittelt, ob die Sitzungsverwaltung verwendet wird. Siehe „`webservice_sessionid_name-Option`“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbankserveroptionen

Die folgende Liste enthält die Erweiterungen der Datenbankserveroptionen in SQL Anywhere Version 16.0.

- **Datenbankserveroption -al** Lässt die Standard-Benutzerauthentifizierung für angegebene Benutzer zu. Siehe „`Datenbankserveroption -al`“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbankoption -al** Lässt die Standard-Benutzerauthentifizierung für angegebene Benutzer der angegebenen Datenbank zu. Siehe „`Datenbankoption -al`“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbankserveroption -gta** Legt fest, welche logischen Prozessoren der Datenbankserver verwenden kann. Siehe „`Datenbankserveroption -gta`“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Datenbankserveroption -uf** Die Datenbankserveroption -uf gilt nun für alle Betriebssysteme. Siehe „Datenbankserveroption -uf“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbankserveroption -ufd** Legt die Aktion fest, die der Datenbankserver ausführt, wenn ein schwerwiegender Fehler oder ein Assertierungsfehler in einer Datenbank auftritt. Siehe „Datenbankserveroption -ufd“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbankserveroption -xs** Die Datenbankserveroption -xs kann mehrfach angegeben werden, damit der Datenbankserver auf Verbindungen über mehrere Ports bzw. Protokolle warten kann. Bisher konnte die Datenbankserveroption -xs nur einmal angegeben werden, sodass Sie alle Ports und Protokolle innerhalb des Parameters der einzelnen Option -xs angeben mussten. Siehe „Datenbankserveroption -xs“ [[SQL Anywhere Server - Datenbankadministration](#)].

Eigenschaften und Systemmonitorstatistiken

Die folgende Liste enthält Erweiterungen von Eigenschaften und Systemmonitorstatistiken in SQL Anywhere Version 16.0.

- **Neue Verbindungseigenschaften** Folgende Verbindungseigenschaften wurden in dieser Version hinzugefügt:
 - `auto_commit_on_create_local_temp_index`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
 - `extern_login_credentials`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
 - `NumLocalTempTables`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
 - `trusted_certificates_file`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
 - `webservice_sessionid_name`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
- **Neue Datenbankeigenschaften** Folgende Datenbankeigenschaften wurden in dieser Version hinzugefügt:
 - `LastCommitRedoPos`
 - `LastSyncedRedoPos`
 - `LastWrittenRedoPos`
 - `UTCTimestampCatalog`
- **Neue Datenbankservereigenschaften** Folgende Datenbankservereigenschaften wurden in dieser Version hinzugefügt:
 - `IsAesniAvailable`
 - `ProcessorAffinity`

Systemprozeduren und Funktionen

Die folgende Liste enthält die Erweiterungen der Systemprozeduren und Funktionen in SQL Anywhere Version 16.0.

- **sa_certificate_info-Systemprozedur** Diese Systemprozedur gibt Informationen zu in der Datenbank gespeicherten Zertifikaten zurück. Siehe „sa_certificate_info-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_cpu_topology-Systemprozedur** Diese Systemprozedur gibt Informationen zur Prozessortopologie des Computers zurück, auf dem der Datenbankserver läuft. Siehe „sa_cpu_topology-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_db_option-Systemprozedur** Mit dieser Systemprozedur können Sie eine Datenbankoption aufheben, während die Datenbank läuft. Siehe „sa_db_option-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_parse_json-Systemprozedur** Diese Systemprozedur gibt eine Darstellung von JSON-Daten in den SQL-Typen ROW und ARRAY zurück. Siehe „sp_parse_json-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_server_option-Systemprozedur** Unter Windows und Linux können Sie die Anzahl der logischen Prozessoren, die der Datenbankserver nutzen kann, nach dem Starten des Datenbankservers ändern, indem Sie die ProcessorAffinity-Eigenschaft der sa_server_option-Systemprozedur verwenden. Siehe „sa_server_option-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Gespeicherte Prozeduren für Verzeichnisse und Dateien** Neue gespeicherte Prozeduren ermöglichen es Ihnen, Verzeichnisse und Dateien, auf die der Server Zugriff hat, aufzulisten und zu bearbeiten, ohne dass Sie Verzeichniszugriffsserver, externe Logins und Verzeichniszugriffs-Proxy-Tabellen einrichten müssen. Diese Systemprozeduren können als gesicherte Funktionen aktiviert und deaktiviert werden.
 - **sp_copy_directory-Systemprozedur** Diese Systemprozedur kopiert ein Verzeichnis. Siehe „sp_copy_directory-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_copy_file-Systemprozedur** Diese Systemprozedur kopiert eine Datei. Siehe „sp_copy_file-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_create_directory-Systemprozedur** Diese Systemprozedur erstellt ein Verzeichnis. Siehe „sp_create_directory-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_delete_directory-Systemprozedur** Diese Systemprozedur löscht ein Verzeichnis. Siehe „sp_delete_directory-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_delete_file-Systemprozedur** Diese Systemprozedur löscht eine Datei. Siehe „sp_delete_file-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **sp_move_directory-Systemprozedur** Diese Systemprozedur verschiebt ein Verzeichnis. Siehe „sp_move_directory-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **sp_move_file-Systemprozedur** Diese Systemprozedur verschiebt eine Datei. Siehe „sp_move_file-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **xp_get_mail_error_code-Systemprozedur** Diese Systemprozedur gibt bei Verwendung der xp_mail-Prozeduren Informationen zum letzten SMTP- oder MAPI-Fehler zurück. Siehe „xp_get_mail_error_code-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **xp_get_mail_error_text-Systemprozedur** Diese Systemprozedur gibt bei Verwendung der xp_mail-Prozeduren Informationen zum letzten SMTP-Fehler zurück. Siehe „xp_get_mail_error_text-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **xp_getenv-Systemprozedur** Diese Systemprozedur gibt den Wert einer Umgebungsvariablen zurück. Siehe „xp_getenv-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **xp_startsmtp-Systemprozedur** Der trusted_certificates-Parameter dieser Systemprozedur kann nun die Optionen **file=file-path** und **cert_name=certificate-name** enthalten. Siehe „xp_startsmtp-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Unterabfrage-Bedingung und Unterabfragen werden für Parameterausdrücke in Prozeduren und Funktionen unterstützt** Unterabfrage-Bedingung und Unterabfragen werden für Parameterausdrücke in Prozeduren und Funktionen unterstützt.
- **BINTOHEX-Funktion** Diese Funktion gibt das hexadezimale Äquivalent einer binären Zeichenfolge zurück. Siehe „BINTOHEX-Funktion [Datentypkonvertierung]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **HEXTOBIN-Funktion** Diese Funktion gibt das LONG BINARY-Äquivalent einer hexadezimalen Zeichenfolge zurück. Siehe „HEXTOBIN-Funktion [Datentypkonvertierung]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SQL-Anweisungen

Die folgende Liste enthält die neuen und erweiterten SQL-Anweisungen in SQL Anywhere Version 16.0.

ALTER TABLE-Anweisung Sie können nun den Eigentümer einer Tabelle ändern. Beim Ändern eines Tabelleneigentümers haben Sie die Möglichkeit, Eigentümerberechtigungen sowie Fremdschlüssel entweder beizubehalten oder zu löschen. Siehe „ALTER TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Anweisungen CREATE CERTIFICATE und DROP CERTIFICATE Sie können jetzt Zertifikate in der Datenbank speichern. Die CREATE CERTIFICATE-Anweisung kann verwendet werden, um Zertifikate in einer Datenbank hinzuzufügen oder zu ersetzen. Die DROP CERTIFICATE-Anweisung kann verwendet werden, um Zertifikate aus einer Datenbank zu entfernen. Siehe „CREATE CERTIFICATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „DROP CERTIFICATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Anweisungen CREATE EVENT und ALTER EVENT Ereignisse können nun in Spiegelungssystemen und in Scale-Out-Systemen mit Schreibschutz auf Spiegelservers und auf Kopieknotten ausgeführt werden. Wenn Sie ein Ereignis erstellen möchten, das auf einem beliebigen

Server ausgeführt werden kann, geben Sie die FOR ALL-Klausel mit der CREATE EVENT-Anweisung oder der ALTER EVENT-Anweisung an. Siehe „[CREATE EVENT-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*] und „[ALTER EVENT-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE FUNCTION-Anweisung [Webdienst] Sie können nun ein in der Datenbank gespeichertes Zertifikat angeben, indem Sie die **certificate_name** Option der CERTIFICATE-Klausel verwenden. Außerdem können Sie nun eine Option mit Kriterien für Keepalive-Timeouts (**kto**) angeben, um eine Keepalive-Verbindung (HTTP/HTTPS) für einen bestimmten Zeitraum zu instanziiieren und im Cache abzulegen. Siehe „[CREATE FUNCTION-Anweisung \[Webdienst\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE INDEX-Anweisung Sie können nun beim Ausführen der Anweisung WITH NULLS DISTINCT angeben. Siehe „[CREATE INDEX-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE PROCEDURE-Anweisung [Webdienste] Sie können nun ein in der Datenbank gespeichertes Zertifikat angeben, indem Sie die **certificate_name** Option der CERTIFICATE-Klausel verwenden. Außerdem können Sie nun eine Option mit Kriterien für Keepalive-Timeouts (**kto**) angeben, um eine Keepalive-Verbindung (HTTP/HTTPS) für einen bestimmten Zeitraum zu instanziiieren und im Cache abzulegen. Siehe „[CREATE PROCEDURE-Anweisung \[Webdienste\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE SERVER-Anweisung Die USING-Klausel der DELIMITER-Option dient dazu, die Verwendung von entweder Schrägstrich (/) oder Backslash (\) als Pfadtrennzeichen in Dateinamen zu erzwingen, die der Verzeichniszugriffsserver zurückgibt. Siehe „[CREATE SERVER-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE TEXT INDEX-Anweisung Sie können jetzt Textindizes für materialisierte Ansichten erstellen. Verwenden Sie die IMMEDIATE REFRESH-Klausel, um einen Textindex für eine materialisierte Ansicht zu erstellen. Siehe „[CREATE TEXT INDEX-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

DROP REMOTE CONNECTION-Anweisung Die DROP REMOTE CONNECTION-Anweisung trennt Ferndatenzugriffsverbindungen mit einem Fremdserver. Siehe „[DROP REMOTE CONNECTION-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE TEMPORARY TRACE EVENT SESSION-Anweisung Erstellt eine Benutzer-Trace-Ereignissitzung. Siehe „[CREATE TEMPORARY TRACE EVENT SESSION-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

CREATE TEMPORARY TRACE EVENT Erstellt ein Benutzer-Trace-Ereignis, das bestehen bleibt, bis die Datenbank gestoppt wird. Siehe „[CREATE TEMPORARY TRACE EVENT-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

ALTER TRACE EVENT SESSION-Anweisung Ermöglicht das Hinzufügen oder Entfernen von Trace-Ereignissen in einer Sitzung, das Hinzufügen oder Entfernen von Zielen in einer Sitzung sowie das Starten und Stoppen einer Trace-Sitzung. Siehe „[ALTER TRACE EVENT SESSION-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

DROP TRACE EVENT-Anweisung Löscht ein benutzerdefiniertes Trace-Ereignis. Siehe „[DROP TRACE EVENT-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

DROP TRACE EVENT SESSION-Anweisung Löscht eine Trace-Ereignissitzung. Siehe „[DROP TRACE EVENT SESSION-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

LOAD TABLE-Anweisung Für die Klausel *load-option* können Sie nun Folgendes angeben:

- **ALLOW ERRORS-Klausel** Ermöglicht das Angeben der Anzahl von Fehlern, die zulässig sind, bevor ein Rollback des Ladevorgangs durchgeführt wird. Der Standardwert ist 0.
- **ROW LOG-Klausel** Wenn während des Einfügens oder der syntaktischen Analyse einer Zeile ein Fehler auftritt, schreibt der Datenbankserver ein Bild der Eingabezeile an den angegebenen Speicherort und meldet zusätzlich die Zeile an den Benutzer.
- **MESSAGE LOG-Klausel** Wenn während des Einfügens oder der syntaktischen Analyse einer Zeile ein Fehler auftritt, schreibt der Datenbankserver den Fehler an den angegebenen Speicherort.

NOTIFY TRACE EVENT-Anweisung Protokolliert ein benutzerdefiniertes Trace-Ereignis in einer Trace-Sitzung. Siehe „[NOTIFY TRACE EVENT-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

TRY...CATCH-Anweisung Sie können eine BEGIN TRY...END TRY-Anweisung (mit BEGIN CATCH...END CATCH) als Fehlerbehandlungsroutine verwenden, um Informationen zu Fehlern innerhalb der kombinierten Anweisungen abzurufen. Siehe „[TRY-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Die folgenden Funktionen und Systemprozeduren wurden hinzugefügt, um Informationen zu Fehlern abzurufen. Die Prozeduren und Funktionen können überall innerhalb einer Anweisung verwendet werden, nicht nur innerhalb einer Fehlerbehandlungsroutine.

- „[ERROR_LINE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[ERROR_MESSAGE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[ERROR_PROCEDURE-Funktion \[Funktionstyp\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[ERROR_SQLCODE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[ERROR_SQLSTATE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[ERROR_STACK_TRACE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[sa_error_stack_trace-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[sa_stack_trace-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „[STACK_TRACE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]

Anweisungen mit ENCRYPTED KEY- oder KEY-Klausel können nun Variablennamen verwenden Sie können für die ENCRYPTED KEY- oder KEY-Klausel entweder eine Zeichenfolge oder einen Variablennamen angeben. Diese Änderung hat Auswirkungen auf die folgenden Anweisungen:

- „ALTER DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE DECRYPTED DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE DECRYPTED FILE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE ENCRYPTED DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE ENCRYPTED FILE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „RESTORE DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „START DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „UNLOAD-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- Openstring-Ausdrücke in einer FROM-Klausel [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Datentypen

Die folgende Liste enthält die Erweiterungen von Datentypen in SQL Anywhere Version 16.0.

- **Unterstützung für nachgestelltes UNSIGNED-Schlüsselwort hinzugefügt** Der Datenbankserver bietet nun Unterstützung für die Typdeklarationen BIGINT UNSIGNED, INTEGER UNSIGNED, SMALLINT UNSIGNED und TINYINT UNSIGNED. Siehe „[Numerische Datentypen](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Programmierschnittstellen

Die folgende Liste enthält die Erweiterungen von Programmierschnittstellen in SQL Anywhere Version 16.0.

- **Unterstützung für Visual Studio 2012 hinzugefügt** Der SQL Anywhere .NET-Datenprovider bietet nun Unterstützung für Microsoft Visual Studio 2012. Siehe „[SQL Anywhere .NET-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Providerunterstützung für .NET Compact Framework 3.5 hinzugefügt** Der SQL Anywhere .NET-Datenprovider bietet nun Unterstützung für Compact Framework 3.5. Siehe „[Funktionen des SQL Anywhere .NET-Datenproviders](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Providerunterstützung für .NET Framework 4.5 hinzugefügt** Der SQL Anywhere .NET 4.0-Datenprovider bietet nun Unterstützung für Compact Framework 4.5. Siehe „[Funktionen des SQL Anywhere .NET-Datenproviders](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Providerunterstützung für Entity Framework 5.0 hinzugefügt** Der SQL Anywhere .NET-Datenprovider bietet nun Unterstützung für Entity Framework 5.0, die neueste Version von Microsoft

Entity Framework. Siehe „Unterstützung für Entity Framework“ [[SQL Anywhere Server - Programmierung](#)].

- **OLE DB-Provider unterstützt nun den DBTYPE_DBTIME2-Datentyp** Der SQL Anywhere OLE DB-Provider unterstützt nun den DBTYPE_DBTIME2-Datentyp. DBTYPE_DBTIME2 (145) ist ein OLE DB-Typ, der den TIME-Datentyp mit Sekundenbruchteilen unterstützt. (DBTYPE_DBTIME unterstützt keine Sekundenbruchteile.) Die Unterstützung für diesen Datentyp erleichtert die Übertragung von Spalten vom Typ TIME zwischen SQL Anywhere-Datenbanken und anderen Datenbankmanagementsystemen (einschließlich SQL Anywhere) ohne Genauigkeitsverlust.
- **Aktualisierte PHP-Treiberunterstützung** Unterstützung für neuere PHP-Versionen bis einschließlich 5.3.18 und 5.4.8 wurde hinzugefügt.

Katalogänderungen

Die folgende Liste enthält alle Katalogänderungen in SQL Anywhere Version 16.0.

- **Neue ISYSCERTIFICATE-Systemtabelle und SYSCERTIFICATE-Systemansicht** In jeder Zeile der ISYSCERTIFICATE-Systemtabelle wird ein Zertifikat als Text im PEM-Format gespeichert. Die SYSCERTIFICATE-Systemansicht ist eine Ansicht dieser Systemtabelle. Siehe „[ISYSCERTIFICATE-Systemtabelle](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[SYSCERTIFICATE-Systemansicht](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue Spalten in ISYSTABCOL und ISYSTAB** Die Spalten nonmaterialized_value und start_schema wurden zu ISYSTABCOL hinzugefügt. Die current_schema-Spalte wurde zu ISYSTAB hinzugefügt. Siehe „[SYSTAB-Systemansicht](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[SYSTABCOL-Systemansicht](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue Spalten in ISYSUSER** Die Spalten user_type, user_dn, user_dn_cached_at, password_creation_time_utc, last_login_time_utc und dual_password wurden zu ISYSUSER hinzugefügt. Siehe „[SYSUSER-Systemansicht](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Im Katalog werden UTC-Zeitstempel verwendet** In früheren Versionen wurden für Katalogtabellen Ortszeitstempel gespeichert und bereitgestellt, die vom Standort des Datenbankservers abhängig waren. Zeitstempel werden nun als UTC-basiertes Äquivalent (Coordinated Universal Time - koordinierte Weltzeit) gespeichert, aber in Ansichten werden sowohl UTC-Zeitstempel als auch Ortszeitstempel bereitgestellt. Anwendungen, die auf der Anzahl der Spalten in den folgenden Systemtabellen und Ansichten basieren, müssen aktualisiert werden:

Name der Systemtabelle	Name der Systemansicht	Hinzugefügte Spalte(n)
ISYSTAB	SYSTAB	last_modified_at_utc
ISYSVIEW	SYSVIEW	mv_last_refreshed_at_utc, mv_known_stale_at_utc
ISYSOBJECT	SYSOBJECT	creation_time_utc

Name der Systemtabelle	Name der Systemansicht	Hinzugefügte Spalte(n)
ISYSUSER	SYSUSER	password_creation_time, last_login_time
ISYSTEXTIDX	SYSTEXTIDX	last_refresh_utc
ISYSEXTERNENVOBJECT	SYSEXTERNENVOBJECT	update_time_utc
ISYSJAR	SYSJAR	update_time_utc
ISYSJAVACLASS	SYSJAVACLASS	update_time_utc
ISYSHISTORY	SYSHISTORY	first_time_utc, last_time_utc
ISYSREMOTEUSER	SYSREMOTEUSER	time_sent_utc, time_received_utc
	Konsolidierte Ansicht SYSREMO- TEUSERS	time_sent_utc, time_received_utc
ISYSCOLSTAT	SYSCOLSTAT	update_time_utc
	Konsolidierte Ansicht SYSCOLS- TATS	update_time_utc

Unix- bzw. Linux-Erweiterungen

Die folgende Liste enthält die Erweiterungen für Unix und Linux in SQL Anywhere Version 16.0.

- **Lokalisierte französische SQL Anywhere-Ressourcen für Linux** Lokalisierte französische SQL Anywhere-Ressourcen sind nun für Linux verfügbar. Eine Lokalisierung wirkt sich auf viele Komponenten aus, darunter Verpackung, Installation, Dokumentation, Software-Benutzeroberfläche sowie Fehler-, Warn- und Informationsmeldungen. Siehe „[Lokalisierte Versionen von SQL Anywhere](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Performanceverbesserungen

Die folgende Liste enthält die Performanceverbesserungen in Version 16.0, die nicht mit sichtbaren Änderungen der Benutzeroberfläche einhergehen.

- **AES-Verschlüsselung** Die Performance wurde verbessert, wenn AES zur Verschlüsselung der Client/Server-Kommunikation und zum Verschlüsseln von Datenbankdateien auf Intel/AMD-Prozessoren für Windows und Linux verwendet wird. Sie können überprüfen, ob eine erweiterte Implementierung des AES-Algorithmus verwendet wird, indem Sie die IsAesniAvailable-

Datenbankservereigenschaft abfragen. Siehe [IsAesniAvailable-Servereigenschaft \[SQL Anywhere Server - Datenbankadministration\]](#).

- **Spalten zu Tabellen hinzufügen** Die Performance bei Verwendung der ALTER TABLE...ADD COLUMN-Anweisung wurde verbessert. In den meisten Situationen, in denen der Standardwert zu einer Konstante ausgewertet werden kann, sollte die ALTER TABLE-Anweisung erheblich schneller ausgeführt werden als in früheren Versionen. Sie müssen ein Upgrade der vorhandenen Datenbanken der Version 11 oder höher durchführen, um diese Funktion verwenden zu können. Datenbanken der Version 10 und früher müssen Sie neu aufbauen, um diese Funktion verwenden zu können.

Verschiedenes

Die folgende Liste enthält verschiedene Erweiterungen in SQL Anywhere Version 16.0.

- **Neue und erweiterte Ausdrücke für die CONTAINS-Suchbedingung** Ein neuer *before-expression* wurde zur CONTAINS-Suchbedingung hinzugefügt. Verwenden Sie den *before-expression*, um eine Nachbarschaftssuche durchzuführen, bei der die relative Reihenfolge der Begriffe beachtet wird. Siehe „CONTAINS-Suchbedingung“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

Der *near-expression* wurde so erweitert, dass jetzt Bedingungen zum Mindestabstand zwischen Suchbegriffen akzeptiert werden. Siehe „CONTAINS-Suchbedingung“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

- **Clientnachrichten zur Verschlüsselung** Beim Erstellen einer Datenbank mit dem Dienstprogramm dbinit oder der CREATE DATABASE-Anweisung werden nun Nachrichten an den Client gesendet, die angeben, welcher Typ der Datenbankverschlüsselung verwendet wird. Falls eine Verschlüsselung erfolgt, wird der verwendete Algorithmus ebenfalls angezeigt.
- **Grafische Pläne und Langtextpläne** Grafische Pläne und Langtextpläne wurden um Optimierungsstatistiken erweitert, die während der Abfrageoptimierung gesammelt wurden. Siehe „Grafische Pläne“ [\[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#) und „Ausführlicher Textplan“ [\[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#).
- **CESU-8-Unterstützung** SQL Anywhere unterstützt nun den CESU-8-Zeichensatz (mit Alias ibm-9400) und die CESU8BIN-Kollation. Verwenden Sie das Dienstprogramm dblank mit der Option -le+, um eine vollständige Liste der unterstützten Zeichensätze abzurufen. Weitere Hinweise zu unterstützten Kollationen finden Sie unter „Alternative Kollationen“ [\[SQL Anywhere Server - Datenbankadministration\]](#).

- **Reservierte Wörter** Im Folgenden finden Sie eine Liste der reservierten Wörter, die der Datenbank in SQL Anywhere Version 16.0 hinzugefügt wurden. Siehe „Reservierte Wörter“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - array
 - json
 - row
 - rowtype
 - unnest
 - varray
- **Bezeichner und Aliasnamen** Bezeichner und Aliasnamen dürfen nicht mehr folgende Zeichen enthalten:
 - Eckige Klammern
 - Invertierte Hochkommata

Hinweis

Wenn Sie eine Datenbank neu laden, die eine frühere Version aufweist als 16.0, entfernen Sie eckige Klammern oder invertierte Hochkommata in Bezeichnern. Andernfalls schlägt das Neuladen fehl.

Weitere Hinweise zu Zeicheneinschränkungen für Bezeichner finden Sie unter „Bezeichner“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]. Weitere Hinweise zu Zeicheneinschränkungen für Aliasnamen finden Sie in der Beschreibung der Klausel *select-list* im „SELECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Verhaltensänderungen von SQL Anywhere

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von SQL Anywhere, die in Version 16.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Standardsicherheitsmodell für einige Systemprozeduren wurde geändert** Einige Systemprozeduren vor Version 16.0 führen Vorgänge in der Datenbank durch, die Berechtigungen erfordern. Im Sicherheitsmodell vor Version 16.0 wurden diese Prozeduren mit den Berechtigungen des Definierers (Eigentümer) ausgeführt, sodass der Benutzer nur die Berechtigung zum Ausführen der Prozedur selbst benötigte und nicht die Berechtigungen für alle von der Prozedur durchgeführten Vorgänge. In einigen Ausnahmefällen war außerdem die DBA-Berechtigung erforderlich.

Ab Version 16.0 ändert sich das Standard-Sicherheitsmodell für diese Prozeduren. Standardmäßig werden diese Prozeduren in neu erstellten Datenbanken mit den Privilegien des aufrufenden Benutzers ausgeführt. Um eine Prozedur ausführen zu können, muss der Aufrufer daher die in der Dokumentation für die Prozedur angegebenen Privilegien haben. Der Aufrufer benötigt außerdem das EXECUTE-Privileg für die Prozedur, das er jedoch als Mitglied von PUBLIC erbt.

Sie können bei der Erstellung der Datenbank steuern, ob das alte Modell oder das neue Modell verwendet wird, indem Sie die neue SYSTEM PROCEDURE AS DEFINER-Klausel der CREATE DATABASE-Anweisung oder die Option -pd des Dienstprogramms Initialisierung (dbinit) angeben.

Manche Systemprozeduren erfordern jedoch immer die in der Dokumentation angegebenen Privilegien, unabhängig von der Einstellung für das Sicherheitsmodell. Eine Liste dieser Prozeduren finden Sie unter „Systemprozeduren vor Version 16.0 als Aufrufer oder Definierer ausführen“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Beim Upgrade einer Datenbank besteht das Standardverhalten darin, das bereits eingerichtete Sicherheitsmodell beizubehalten. Das heißt beispielsweise, dass beim Upgrade einer Datenbank der Version 12.0.1 die Datenbank nach dem Upgrade weiterhin das alte Sicherheitsmodell verwendet, sofern Sie nichts anderes angeben.

Wenn Sie zum Zeitpunkt des Upgrades steuern möchten, welches Sicherheitsmodell verwendet wird, können Sie die SYSTEM PROCEDURE AS DEFINER-Klausel der ALTER DATABASE-Anweisung oder die Option -pd des Dienstprogramms zum Upgrade (dbupgrad) verwenden. Dieselben Ausnahmen bezüglich der Systemprozeduren, die immer sowohl EXECUTE als auch spezifische Privilegien erfordern, gelten für Upgrade wie Erstellung.

Hinweis

Die Entscheidung über das Sicherheitsmodell (Aufrufer oder Definierer) wirkt sich nicht auf das Standardverhalten von benutzerdefinierten Prozeduren aus, die sich weiterhin nach dem Definierer richten. Auch wenn die Standardeinstellung für Systemprozeduren in den Aufrufer geändert wurde, bleibt die Standardeinstellung für benutzerdefinierte Prozeduren der Definierer.

- „Änderungen für Systemprozeduren, die mit Privilegien verbundene Vorgänge ausführen“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „Systemprozeduren vor Version 16.0 als Aufrufer oder Definierer ausführen“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
 - Liste der Prozeduren, die durch die Aufrufer/Definierer-Einstellung beeinflusst werden [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
 - „ALTER DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „Dienstprogramm Initialisierung (dbinit)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „Dienstprogramm zum Upgrade (dbupgrad)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- **Relative Pfade und Sandboxing** In früheren Versionen war der Standardwert für relative Pfade immer das Arbeitsverzeichnis des Datenbankservers. Wenn Sandboxing für eine Datenbank aktiviert ist, sind relative Pfade nur relativ zum Verzeichnis der Datenbank und nicht zum Verzeichnis des Datenbankservers. Siehe „[Sandboxing](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **SELECT * wird in Ansichtsdefinitionen unterstützt** In früheren Versionen wurde SELECT * nur in der Hauptabfrage der CREATE VIEW-Anweisung unterstützt. Nun wird es in der Hauptabfrage unterstützt sowie in einer Unterabfrage, einer abgeleiteten Tabelle oder einer Unterabfrage-Bedingung der CREATE VIEW-Anweisung. Siehe „[CREATE VIEW-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Mindestkennwortlänge geändert** In früheren Versionen betrug die Mindestkennwortlänge standardmäßig 0 Zeichen. Der Standardwert für die Mindestkennwortlänge wurde auf 3 Zeichen geändert. Wenn Ihre Anwendung 0 Zeichen lange Kennwörter zulässt, können Sie die folgende Anweisung in einer neuen SQL Anywhere-Datenbank ausführen, um die Standardeinstellung an frühere Versionen anzupassen:

```
SET OPTION PUBLIC.min_password_length=0;
```

Weitere Hinweise finden Sie unter „min_password_length-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Standardverhalten beim Erstellen von Indizes für lokale temporäre Tabellen geändert** In früheren Versionen hat der Datenbankserver vor dem Erstellen eines Indexes für eine lokale temporäre Tabelle immer eine COMMIT-Anweisung ausgeführt. Nun führt der Datenbankserver vor dem Erstellen eines Indexes für eine lokale temporäre Tabelle keine COMMIT-Anweisung mehr aus. Sie können dieses Verhalten steuern, indem Sie die auto_commit_on_create_local_temp_index-Datenbankoption setzen. Siehe „auto_commit_on_create_local_temp_index-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue maximale Paketgröße** Die maximale Paketgröße wurde von 16000 auf 65535 Byte erhöht. Clients der Versionen SQL Anywhere 12 und früher sind auf 16000 Byte beschränkt, wenn sie Verbindungen mit SQL Anywhere 16-Datenbankservern herstellen. Die Standardpaketgröße wurde nicht geändert. Siehe „Verbindungsparameter CommBufferSize (CBSIZE)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Grenze der Datenbankserveroption -ch geändert** Wenn Sie mit der Option -ch eine maximale Cachegröße angeben, die kleiner ist als 64 MB, passt der Datenbankserver die maximale Cachegröße auf 64 MB an. Wenn Sie eine maximale Cachegröße von weniger als 64 MB benötigen (wobei diese Einstellung nicht empfohlen wird), können Sie die Option -chx verwenden. Diese Änderungen gelten nicht unter Windows Mobile. Siehe „Datenbankserveroption -ch“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Gleichzeitige Indexerstellung** In früheren Versionen musste die CREATE INDEX-Anweisung eine exklusive Tabellensperre setzen, wenn ein Index erstellt wurde. Nun setzt der Vorgang für kurze Zeiträume am Anfang und am Ende des Vorgangs eine exklusive Tabellensperre und für die meiste Zeit des Vorgangs eine gemeinsame Sperre, sodass andere Verbindungen auf die Tabelle zugreifen können, während der Index erstellt wird. Für die Verbindung, die den Index erstellt, wird der Zugriff auf die Tabelle blockiert, bis der Index erstellt ist. Sie müssen ein Upgrade bestehender Datenbanken durchführen, um diese Funktion verwenden zu können. Siehe „CREATE INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Spiegelungsverbindungen können getrennt werden, wenn die Verbindungen das Übernehmen des Transaktionslogs verhindern** Verbindungen mit einem Kopieknoten oder der Spiegeldatenbank werden in manchen Fällen getrennt, wenn diese Verbindungen das Übernehmen des Transaktionslogs verhindern. Wenn beispielsweise eine Verbindung eine Prozedur verwendet, die das Transaktionslog zu ändern oder zu löschen versucht, wird die Verbindung getrennt, die das Übernehmen des Transaktionslogs blockiert, und eine Meldung wird in die Serverkonsole ausgegeben. Siehe „Abfragen in der Spiegeldatenbank“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Änderungen am Plan-Caching** Abfrageausführungspläne werden für Abfragen mit langen Ausführungszeiten nicht in Cachespeicher geschrieben, weil die Vorteile durch das Vermeiden der Abfrageoptimierung im Vergleich zu den Gesamtkosten der Abfrage klein sind. Außerdem versucht der Datenbankserver nicht, wiederverwendbare Abfragepläne für Abfragen zu rekonstruieren, die sehr empfindlich auf die Werte ihrer Hostvariablen reagieren.

- **Neuer Standardwert für Anforderungsvariablen an HTTP-Server** In früheren Versionen gab es keine Grenze für die Anzahl von HTTP-Eingabevariablen, die in einer Anforderung gesendet werden konnten. Die MaxRequestVars-Protokolloption beschränkt die Anzahl von HTTP-Eingabevariablen. Der Standardwert für die MaxRequestVars-Protokolloption ist 10000. Siehe „MaxRequestVars-Protokolloption (MAXVARS)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neues Standardverhalten für Datenbankassertierungen** In früheren Versionen wurden Datenbank-Assertierungsfehler als Datenbankserver-Assertierungsfehler behandelt. Dies führte dazu, dass der Datenbankserver heruntergefahren wurde oder auf allen Clientverbindungen einen Fehler zurückgegeben hat. Datenbank-Assertierungsfehler werden nun getrennt von Datenbankserver-Assertierungsfehlern behandelt und führen dazu, dass die Datenbank heruntergefahren wird, während der Datenbankserver weiterläuft. Siehe „Datenbankserveroption -ufd“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verhaltensänderungen von gesicherten Funktionen** Schlüssel für gesicherte Funktionen haben nun eine Mindestlänge von 6 Zeichen und einige gesicherte Funktionen können nicht global deaktiviert werden, sondern nur für einzelne Verbindungen. Siehe „Gesicherte Funktionen“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Vertrauliche Daten werden in der Ausgabe verschleiert** Kennwörter und Chiffrierschlüssel werden verschleiert, wenn Anweisungen, die sie enthalten, in das Anforderungslog geschrieben, durch die Diagnoseprotokollierung protokolliert oder von DESCRIBE-Anweisungen als Spaltennamen verwendet werden. Dieses Verhalten gilt auch für die Ausgabe der REWRITE-Funktion und der LastStatement-Verbindungseigenschaft sowie für in der Ereignisprotokollierung erfasste Anweisungen.

Vertrauliche Parameter sowie Kennwörter und Schlüssel werden für die folgenden Funktionen, Prozeduren und Anweisungen ausgeblendet:

- ENCRYPT-Funktion
 - DECRYPT-Funktion
 - sa_verify_password-Systemprozedur
 - sp_addlogin-Prozedur (Adaptive Server Enterprise-Kompatibilitätsprozedur)
 - sp_password-Systemprozedur
 - xp_startmail-Systemprozedur
 - xp_startsmtp-Systemprozedur
 - sp_create_secure_feature_key-Systemprozedur
 - sp_alter_secure_feature_key-Systemprozedur
 - sp_use_secure_feature_key-Systemprozedur
 - GRANT CONNECT-Anweisung
 - CREATE DATABASE-Anweisung
 - CREATE ENCRYPTED DATABASE-Anweisung
 - CREATE ENCRYPTED TABLE DATABASE-Anweisung
 - CREATE ENCRYPTED FILE-Anweisung
 - CREATE DECRYPTED DATABASE-Anweisung
 - CREATE DECRYPTED FILE-Anweisung
 - START DATABASE-Anweisung
 - DROP DATABASE-Anweisung
 - CREATE EXTERNLOGIN-Anweisung
 - SET TEMPORARY OPTION secure_feature_key = key-Anweisung
- **Lizenzierungsänderung für Personal Server (dbeng16)** Der Personal Server ist auf vier Prozessorkerne in einer CPU beschränkt. Bisher war der Personal Server auf eine CPU beschränkt. Siehe „[Dienstprogramm für die Serverlizenzierung \(dblic\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Verhaltensänderungen von Datenbankdienstprogrammen

- **Verhaltensänderung des Dienstprogramms zum Upgrade (dbupgrad)** Standardmäßig wird die Datenbank nach einem Upgrade neu gestartet. Wenn Sie verhindern möchten, dass die Datenbank neu gestartet wird, verwenden Sie die Option -nrs, um die Datenbank nach dem Upgrade zu stoppen. Nach erfolgreichem Abschluss eines Upgrades wird das Transaktionslog umbenannt. Siehe „[Dienstprogramm zum Upgrade \(dbupgrad\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- **Verhaltensänderung des Dienstprogramms für Supportanfragen (dbsupport)**
 - **Option -cc autosubmit** Bisher wurden mit der Option -cc autosubmit Crash-Berichte und Diagnoseinformationen nur nach einem Absturz automatisch eingesendet. Nun werden mit der Option -cc autosubmit Crash-Berichte automatisch eingesendet und Diagnoseinformationen regelmäßig an das Software-Entwicklungsteam gesendet, um zur Verbesserung des Produkts beizutragen.

Während der Installation wird diese Option standardmäßig aktiviert. Siehe „[Regelmäßig Performancedaten senden](#)“ [*SQL Anywhere Server - Datenbankadministration*]
 - **Option -cp autodetect** Die Option -cp autodetect wird nun unter Unix unterstützt. Unter Unix kann dbsupport mithilfe der Option -cp autodetect Proxy-Server und Port konfigurieren, indem es

auf die entsprechenden Einstellungen der HTTP_PROXY-Umgebungsvariablen zurückgreift. Siehe „Dienstprogramm für Supportanfragen (dbsupport)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Option -ch (nicht mehr empfohlen)** Die Optionen -ch und -ch- von dbsupport werden nicht mehr unterstützt. Außerdem akzeptiert die Option -e von dbsupport nicht mehr die Option -ch. Das heißt, Sie können -ech nicht angeben. Siehe „Dienstprogramm für Supportanfragen (dbsupport)“ [*SQL Anywhere Server - Datenbankadministration*].

Verhaltensänderungen von Systemprozeduren und Funktionen

In der folgenden Liste sind die Verhaltensänderungen der Systemprozeduren und Funktionen in SQL Anywhere Version 16.0 aufgeführt.

- **Das Zurückgeben von Informationen für andere Verbindungen erfordert nun Privilegien.** Bei Datenbanken der Version 12 oder früher benötigen Sie keine Berechtigungen, um die folgenden Systemprozeduren und Funktionen ausführen und Informationen für eine beliebige Verbindung zurückgeben zu können.

Bei Datenbanken der Version 16 kann jeder Benutzer diese Systemprozeduren und Funktionen ausführen, um Informationen für die aktuelle Verbindung zurückzugeben. Um diese Systemprozeduren und Funktionen für die Rückgabe von Informationen für andere Verbindungen ausführen zu können, benötigen Sie jedoch das Systemprivileg SERVER OPERATOR, MONITOR oder DROP CONNECTION.

- „CONNECTION_EXTENDED_PROPERTY-Funktion [Zeichenfolge]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „CONNECTION_PROPERTY-Funktion [System]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „NEXT_CONNECTION-Funktion [System]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_conn_activity-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_conn_compression_info-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_conn_info-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_conn_list-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_conn_options-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_conn_properties-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]

- **Das Zurückgeben von Informationen für andere Datenbanken erfordert nun Privilegien.** Bei Datenbanken der Version 12 oder früher benötigen Sie keine Berechtigungen, um die folgenden Systemprozeduren und Funktionen ausführen und Informationen für eine beliebige Datenbank zurückgeben zu können.

Bei Datenbanken der Version 16 kann jeder Benutzer diese Systemprozeduren und Funktionen ausführen, um Informationen für die aktuelle Datenbank zurückzugeben. Um diese Systemprozeduren

und Funktionen für die Rückgabe von Informationen für andere Datenbanken ausführen zu können, benötigen Sie jedoch das Systemprivileg SERVER OPERATOR oder MONITOR.

- „DB_EXTENDED_PROPERTY-Funktion [System]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „DB_NAME-Funktion [System]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „DB_PROPERTY-Funktion [System]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „NEXT_DATABASE-Funktion [System]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_db_info-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_db_list-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_db_properties-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- **Erweiterungen der TRACEBACK-Funktion** Die TRACEBACK-Funktion gibt einen Aufruf-Stack zurück, der mit Objektnamen und Zeilennummern versehen ist. Dies vereinfacht das Auffinden der eigentlichen Anweisungen, die im Aufruf-Stack gemeldet werden. Siehe „TRACEBACK-Funktion [Verschiedene]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Änderungen der SMTP- und MAPI-Rückgabecodes** Die folgenden Änderungen wurden an den Rückgabecodes vorgenommen, die von den Systemprozeduren xp_startmail, xp_startsmtp, xp_sendmail, xp_stopmail und xp_stopsmtp verwendet werden:

Rückgabecode	Bedeutung in früheren Versionen	Bedeutung in Datenbanken der Version 16.0	Zusätzliche Informationen
-1		Unbekannter Fehler	Neuer Fehlercode
0	Erfolg	Erfolg	
1		Ein ungültiger Parameter wurde angegeben	Neuer Fehlercode
2	xp_startmail oder xp_startsmtp fehlgeschlagen	Kein Speicher mehr	
3	xp_stopmail oder xp_stopsmtp fehlgeschlagen	xp_startmail oder xp_startsmtp wurde nicht aufgerufen	
4		Ungültiger Hostname	Neuer Fehlercode
5	xp_sendmail fehlgeschlagen	Verbindungsfehler	
6		Fehler der sicheren Verbindung	Neuer Fehlercode
7		MAPI-Funktionen sind nicht verfügbar	Neuer Fehlercode

Die Systemprozeduren `xp_get_mail_error_code` und `xp_get_mail_error_text` geben nun zusätzliche Informationen zu den Rückgabecodes zurück. Sie müssen ein Upgrade bestehender Datenbanken durchführen, um die neuen Systemprozeduren und Fehlercodes zu erhalten. Siehe:

- „`xp_get_mail_error_code`-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „`xp_get_mail_error_text`-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- Rückgabecodes für MAPI- und SMTP-Systemprozeduren [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Wenn Ihre Anwendung nicht so geändert werden kann, dass sie nach den aktualisierten Rückgabecodes sucht, können Sie den Datenbankserver so konfigurieren, dass er auf die Rückgabecodes aus früheren Versionen der Software zurückgreift, indem Sie die Datei `dbext.dll` der Version 12 verwenden und das Skript `use_old_dbextf.sql`, das sich im Unterverzeichnis `scripts` Ihrer SQL Anywhere-Installation befindet. Siehe „[Deployment des Datenbankservers](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **Funktionen `USER_NAME` und `SUSER_NAME` geben nun `VARCHAR` zurück** Die Funktionen `user_name` und `suser_name` geben nun Daten vom Typ `VARCHAR` statt `LONG VARCHAR` zurück. Wenn Sie materialisierte Ansichten haben, die diese Funktionen verwenden, müssen Sie sie neu aufbauen.
- **Fehlerbehandlung** Damit der `LOAD TABLE`-Vorgang durchgeführt werden kann, wenn die Anweisung beim Einfügen oder bei der syntaktischen Analyse von Zeilen Fehler feststellt, können problematische Zeilen in ein `ROW LOG` und eine Fehlermeldung in ein `MESSAGE LOG` geschrieben werden. Nachdem eine festlegbare Anzahl von Fehlern aufgetreten ist, schlägt die Anweisung fehl und wird zurückgesetzt. Die folgenden Klauseln wurden hinzugefügt, um diese Funktion zu unterstützen:
 - `ALLOW...ERRORS`
 - `ROW LOG...`
 - `MESSAGE LOG...`

Siehe „[LOAD TABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Verhaltensänderungen der Programmierschnittstellen

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von Programmierschnittstellen, die in SQL Anywhere Version 16.0 eingeführt wurden.

- **Schwerwiegender Fehler "Kein Speicher mehr"** Unter Windows konnten frühere Versionen der Clientbibliotheken versuchen, das Fenster `Kein Speicher mehr` anzuzeigen, wenn eine Clientbibliotheksroutine keinen dynamischen Speicher belegen konnte. Eine Antwort des Benutzers wurde abgewartet und anschließend die Anwendung beendet. Dies tun die Clientbibliotheken nicht mehr. Wenn jetzt kein Speicher mehr verfügbar ist, wird ein Eintrag im Windows-Ereignisprotokoll geschrieben, und zwar mit der Ereignisquelle `SQLANY16.0` (32-Bit-Anwendungen) bzw. `SQLANY64 16.0` (64-Bit-Anwendungen). Nach dem Schreiben des Ereignisprotokolleintrags wird die Software weiter ausgeführt, sodass die anfordernde Routine die Fehlerbedingung verarbeiten kann. Diese Änderung verhindert, dass als Windows-Dienste geschriebene Clientanwendungen hängen. Es

ist Aufgabe des Entwicklers der Clientanwendung, zu ermitteln, warum eine Anwendung den gesamten verfügbaren Speicher belegt. Siehe „[Ereignisprotokollmeldungen formatieren](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **ODBC-Funktion SQLTables** Die **SQLTables**-Funktion wurde so korrigiert, dass die dem Datenbankeigentümer gehörenden Systemtabellen, die von der CREATE DATABASE-Anweisung installiert wurden, als SYSTEM TABLE und nicht als TABLE kategorisiert werden (in der TABLE_TYPE-Spalte der Ergebnismenge). Tabellen wie dbo.jdbc_function_escapes und dbo.spt_jdatatype_info werden nun als Systemtabellen klassifiziert.

Die **SQLTables**-Funktion wurde so korrigiert, dass die dem Datenbankeigentümer gehörenden Systemansichten, die von der CREATE DATABASE-Anweisung installiert wurden, als SYSTEM VIEW und nicht als VIEW kategorisiert werden (in der TABLE_TYPE-Spalte der Ergebnismenge). Ansichten wie dbo.systypes und dbo.sysusers werden nun als Systemansichten klassifiziert.

- **JDBC-Methode DatabaseMetaData.getTables** Die **getTables**-Methode der DatabaseMetaData-Klasse wurde so korrigiert, dass die dem Datenbankeigentümer gehörenden Systemtabellen, die von der CREATE DATABASE-Anweisung installiert wurden, als SYSTEM TABLE und nicht als TABLE kategorisiert werden (in der TABLE_TYPE-Spalte der Ergebnismenge). Tabellen wie dbo.jdbc_function_escapes und dbo.spt_jdatatype_info werden nun als Systemtabellen klassifiziert.

Die **getTables**-Methode wurde so korrigiert, dass die dem Datenbankeigentümer gehörenden Systemansichten, die von der CREATE DATABASE-Anweisung installiert wurden, als SYSTEM VIEW und nicht als VIEW kategorisiert werden (in der TABLE_TYPE-Spalte der Ergebnismenge). Ansichten wie dbo.systypes und dbo.sysusers werden nun als Systemansichten klassifiziert.

- **Änderungen der C-API für Datenbanktools** Das **script_full_path**-Feld wurde aus der **a_sync_db**-Struktur entfernt. Die Verwendung dieses Felds wird seit Version 8.x nicht mehr unterstützt. Siehe **a_sync_db-Struktur** [[Datenbanktools](#)] [[SQL Anywhere Server - Programmierung](#)].

Neue Felder (wait_before_start, wait_after_end, backup_comment, auto_tune_writers und backup_history) und Enumerationen wurden zur **a_backup_db**-Struktur hinzugefügt. Siehe **a_backup_db-Struktur** [[Datenbanktools](#)] [[SQL Anywhere Server - Programmierung](#)].

- **Bezeichnerregeln und Datentypeinschränkungen werden nun für JSON-Objekte erzwungen** JSON-Objektbezeichner müssen den in der Datenbank definierten Bezeichnerregeln entsprechen. Außerdem erzwingt der Datenbankserver dieselben Grenzen für JSON-Datentypen wie für die zugrunde liegenden Datentypen ROW und ARRAY.

Verhaltensänderungen von SQL-Anweisungen

Die folgende Liste enthält alle SQL-Verhaltensänderungen in SQL Anywhere Version 16.0.

ALTER DATABASE-Anweisung Standardmäßig wird die Datenbank nun nach einem Upgrade gestoppt und neu gestartet. Wenn Sie verhindern möchten, dass die Datenbank neu gestartet wird, verwenden Sie die RESTART OFF-Klausel, um die Datenbank nach dem Upgrade zu stoppen. Siehe „[ALTER DATABASE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SELECT * wird in Ansichtsdefinitionen unterstützt In früheren Versionen wurde SELECT * nur in der Hauptabfrage der CREATE VIEW-Anweisung unterstützt. Nun wird es in der Hauptabfrage

unterstützt sowie in einer Unterabfrage, einer abgeleiteten Tabelle oder einer Unterabfrage-Bedingung der CREATE VIEW-Anweisung. Siehe „CREATE VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

START DATABASE-Anweisung Die ON-Klausel der START DATABASE-Anweisung wird nicht mehr empfohlen. Mit der START DATABASE-Anweisung können nur Datenbanken auf dem aktuellen Datenbankserver gestartet werden. Bisher konnten Sie die ON-Klausel in Interactive SQL verwenden, um einen zu verwendenden Datenbankserver anzugeben. Siehe „START DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

ALTER SERVER-Anweisung Die CONNECTION CLOSE-Klausel der ALTER SERVER-Anweisung wird nicht unterstützt. Verwenden Sie die DROP REMOTE CONNECTION-Anweisung. Siehe „DROP REMOTE CONNECTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

LOAD TABLE-Anweisung

- **STRIP ON-Klausel** Die STRIP ON-Klausel der LOAD TABLE-Anweisung wird nicht unterstützt. Verwenden Sie stattdessen die STRIP RTRIM-Klausel. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SYSUSER-Systemansicht speichert nun eigenständige Rolleninformationen Bisher wurden in der SYSUSER-Systemansicht nur Benutzer im System gespeichert. Jetzt werden eigenständige Rollen (Rollen, die nicht Benutzern zugeordnet sind) ebenfalls in dieser Ansicht gespeichert. Für diese Rollen sind jedoch nur die Spalten user_id, object_id, user_name und user_type aussagekräftig. Siehe „SYSUSER-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Nicht mehr empfohlene und nicht mehr weiterentwickelte SQL Anywhere-Funktionen

Nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste nicht mehr empfohlener Funktionen in Version 16.0:

- **Berechtigungen nicht mehr empfohlen** In SQL Anywhere 16.0 wird ein neues rollen- und privilegbasiertes Sicherheitsmodell eingeführt, das an die Stelle des bisherigen Sicherheitsmodells tritt. Das neue rollenbasierte Sicherheitsmodell ermöglicht Ihnen eine genaue Kontrolle darüber, welche mit Privilegien verbundenen Aufgaben Benutzer ausführen können, und eine einfachere

Verwaltung der Zugriffssteuerung. Die früheren Datenbankberechtigungen werden nun nicht mehr empfohlen und wurden entweder durch Systemrollen oder durch Kompatibilitätsrollen ersetzt:

- DBA-Berechtigung
- REMOTE DBA-Berechtigung
- BACKUP-Berechtigung
- VALIDATE-Berechtigung
- PROFILE-Berechtigung
- READCLIENTFILE-Berechtigung
- WRITECLIENTFILE-Berechtigung
- READFILE-Berechtigung
- WRITEFILE-Berechtigung
- RESOURCE-Berechtigung

Siehe „[Was ist aus den Berechtigungen und Gruppen geworden?](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Optionen java_location und java_main_userid** Die Optionen java_location und java_main_userid werden seit Version 11.0.0 mit der Einführung der ISYSEXTERNENV-Systemtabelle nicht mehr empfohlen. Anwendungen, die mit der java_location-Option angeben, welche Java VM zu verwenden ist, sollten stattdessen die ALTER EXTERNAL ENVIRONMENT-Anweisung verwenden, um den Speicherort der Java VM in der ISYSEXTERNENV-Tabelle festzulegen. Für die java_main_userid-Option gibt es keine Alternative. Siehe „[Unterstützung für externe Umgebungen in SQL Anywhere](#)“ [*SQL Anywhere Server - Programmierung*].
- **secure_feature_key-Datenbankoption** Die secure_feature_key-Datenbankoption wird nicht mehr empfohlen. Verwenden Sie die sp_use_secure_feature_key-Systemprozedur. Siehe „[sp_use_secure_feature_key-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **PHP-Funktionen** Die folgenden PHP-Funktionen werden unterstützt, aber nicht mehr empfohlen. Für jede dieser Funktionen gibt es ein neues Äquivalent mit einem Namen, der mit sasql_ statt mit sqlanywhere_ beginnt.

Nicht mehr empfohlene PHP-Funktionen	Äquivalente neue PHP-Funktionen
sqlanywhere_commit	„sasql_commit“ [<i>SQL Anywhere Server - Programmierung</i>]
sqlanywhere_connect	„sasql_connect“ [<i>SQL Anywhere Server - Programmierung</i>]
sqlanywhere_data_seek	„sasql_data_seek“ [<i>SQL Anywhere Server - Programmierung</i>]
sqlanywhere_disconnect	„sasql_disconnect“ [<i>SQL Anywhere Server - Programmierung</i>]
sqlanywhere_error	„sasql_error“ [<i>SQL Anywhere Server - Programmierung</i>]
sqlanywhere_errorcode	„sasql_errorcode“ [<i>SQL Anywhere Server - Programmierung</i>]
sqlanywhere_execute	

Nicht mehr empfohlene PHP-Funktionen	Äquivalente neue PHP-Funktionen
sqlanywhere_fetch_array	„sasql_fetch_array“ [SQL Anywhere Server - Programmierung]
sqlanywhere_fetch_field	„sasql_fetch_field“ [SQL Anywhere Server - Programmierung]
sqlanywhere_fetch_object	„sasql_fetch_object“ [SQL Anywhere Server - Programmierung]
sqlanywhere_fetch_row	„sasql_fetch_row“ [SQL Anywhere Server - Programmierung]
sqlanywhere_free_result	„sasql_free_result“ [SQL Anywhere Server - Programmierung]
sqlanywhere_identity	
sqlanywhere_num_fields	„sasql_num_fields“ [SQL Anywhere Server - Programmierung]
sqlanywhere_num_rows	„sasql_num_rows“ [SQL Anywhere Server - Programmierung]
sqlanywhere_pconnect	„sasql_pconnect“ [SQL Anywhere Server - Programmierung]
sqlanywhere_query	„sasql_query“ [SQL Anywhere Server - Programmierung]
sqlanywhere_result_all	„sasql_result_all“ [SQL Anywhere Server - Programmierung]
sqlanywhere_rollback	„sasql_rollback“ [SQL Anywhere Server - Programmierung]
sqlanywhere_set_option	„sasql_set_option“ [SQL Anywhere Server - Programmierung]

Nicht mehr weiterentwickelte Funktionen

Im Folgenden finden Sie eine Liste nicht mehr weiterentwickelter Funktionen in Version 16.0:

- SQL Anywhere JDBC 3.0-Treiber** Der SQL Anywhere JDBC 3.0-Treiber ist nicht in dieser Softwareversion enthalten. Anwendungen, die die Datei *sajdbc.jar* verwenden, müssen auf die Datei *sajdbc4.jar* umgestellt werden. Siehe „[JDBC-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].
- JDBC-basierte Serverklassen für den Ferndatenzugriff** Die JDBC-basierten Fremdserverklassen werden in dieser Softwareversion nicht unterstützt. Zu diesen Klassen gehören die Fremdserverklassen ASEJDBC, IQJDBC und SAJDBC. Siehe „[CREATE SERVER-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- OLE DB-Provider unterstützt Borland Delphi nicht mehr** Der SQL Anywhere OLE DB-Provider kann nicht mehr mit älteren Versionen von Borland Delphi verwendet werden. Borland Delphi unterstützt die Datentypen DBTYPE_DBTIME2 und DBTYPE_DBTIMESTAMPOFFSET nicht. Embarcadero unterstützt die Versionen von Delphi aus den Jahren 2006 und früher nicht mehr. Siehe <http://www.embarcadero.com/products/delphi/previous-versions>.

- **Address Windowing Extensions (AWE) nicht unterstützt** Die Verwendung von Address Windowing Extensions für 32-Bit-Windows wird nicht unterstützt und die Datenbankserveroptionen -cw und -cm werden nicht mehr unterstützt. Wenn Sie einen großen Cache benötigen, sollten Sie die 64-Bit-Version des SQL Anywhere-Datenbankservers unter einem 64-Bit-Betriebssystem verwenden.

Neue MobiLink-Funktionen

Im Folgenden finden Sie eine Liste der neuen Funktionen von MobiLink, die in Version 16.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **MobiLink-Profiler ersetzt MobiLink-Monitor** Der MobiLink-Monitor wurde entfernt. Mit dem MobiLink-Profiler können Sie während der Entwicklungs- und Testphase detaillierte Performanceinformationen zu Synchronisationen abrufen. Verwenden Sie in der Produktion den SQL Anywhere-Monitor für die Überwachung eines MobiLink-Deployments. Siehe „MobiLink-Profiler“ [*MobiLink - Serveradministration*].
- **Java- und .NET-Datenskripten können keinen SQL-Code mehr zurückgeben** Java- und .NET-Synchronisationsskripten können keine Zeichenfolgen mehr zurückgeben, die vom MobiLink-Server als SQL-Anweisungen interpretiert werden. Java- und .NET-Skripten, die SQL-Code zurückgeben, erzeugen nun Fehler, was dazu führt, dass die Synchronisation fehlschlägt.

Es wird empfohlen, dass Sie die direkte Zeilenbehandlung für Java- oder .NET-Skripten verwenden, die Daten in die konsolidierte Datenbank hochladen bzw. aus dieser herunterladen müssen. Siehe „Direkte Zeilenbehandlung“ [*MobiLink - Serveradministration*].

- **Vorwärts- und Rückwärtskompatibilität des MobiLink-Servers** Jeder Build eines MobiLink-Clients der Version 16 kann mit jedem Build des MobiLink-Servers der Version 16 synchronisiert werden. Bisher musste der Build des MobiLink-Servers mindestens genauso hoch sein wie der des MobiLink-Clients.
- **Unterstützung für den XMLTYPE-Datentyp** Synchronisation mit Spalten im Oracle-Datentyp XMLTYPE wird nun unterstützt. Siehe *XMLTYPE-Datentyp in Oracle* [*MobiLink - Serveradministration*].
- **MobiLink-Server unterstützt integrierte LDAP-Authentifizierung** Die MobiLink-Benutzerauthentifizierung über LDAP-Server ist nun in MobiLink-Server verfügbar. Die LDAP-Authentifizierung wird unter Verwendung der folgenden neuen gespeicherten MobiLink-Prozeduren implementiert:
 - **ml_add_ldap_server-Systemprozedur** Mit dieser Systemprozedur können Sie LDAP-Server hinzufügen, löschen und aktualisieren. Siehe „ml_add_ldap_server-Systemprozedur“ [*MobiLink - Serveradministration*].
 - **ml_add_certificates_file-Systemprozedur** Verwenden Sie diese Systemprozedur, um die vertrauenswürdigen Zertifikate einzurichten, die für die verschlüsselte TLS-Kommunikation zwischen dem MobiLink-Server und einem LDAP-Server benötigt werden. Siehe „ml_add_certificates_file-Systemprozedur“ [*MobiLink - Serveradministration*].

- **ml_add_user_auth_policy-Systemprozedur** Verwenden Sie diese Systemprozedur, um Richtlinien für die MobiLink-Benutzerauthentifizierung einzurichten. Siehe „ml_add_user_auth_policy-Systemprozedur“ [*MobiLink - Serveradministration*].
- **Neue ml_model_drop-Systemprozedur** Verwenden Sie diese Systemprozedur, um ein Synchronisationsmodell und das zugeordnete Schema zu löschen. Nur Synchronisationsmodelle, die mithilfe des MobiLink 16.0 Plug-Ins für Sybase Central hinzugefügt wurden, können gelöscht werden. Siehe „ml_model_drop-Systemprozedur“ [*MobiLink - Serveradministration*].
- **Neue ml_model_check_all_schema-Systemprozedur** Verwenden Sie diese Prozedur, um den Status der einzelnen Schemaobjekte für die einzelnen bereitgestellten Synchronisationsmodelle zu überprüfen. Diese gespeicherte Prozedur gibt Informationen für alle bereitgestellten Synchronisationsmodelle zurück. Siehe „ml_model_check_all_schema-Systemprozedur“ [*MobiLink - Serveradministration*].
- **Neue ml_model_check_version_schema-Systemprozedur** Verwenden Sie diese Prozedur, um den Status der einzelnen Schemaobjekte für die einzelnen bereitgestellten Synchronisationsmodelle zu überprüfen. Diese gespeicherte Prozedur gibt Informationen für die angegebene Skriptversion zurück. Siehe „ml_model_check_all_schema-Systemprozedur“ [*MobiLink - Serveradministration*].
- **MobiLink-Server und -Clients unterstützen nun TLS-Version 1.1** MobiLink-Server und MobiLink-Clients unterstützen nun TLS-Version 1.1. Dies gilt sowohl für die direkte TLS-Synchronisation als auch für die HTTPS-Synchronisation. TLS-Version 1.1 wird unter Mac OS X nicht unterstützt.
- **Informationen zu Sperren und Blockierungen melden** Standardmäßig meldet der MobiLink-Server nun Informationen zu Sperren und Blockierungen für länger laufende benutzerdefinierte Skripten an MobiLink-Profiler, zusätzlich zum Protokollieren der Informationen in der Logdatei des MobiLink-Servers.

Verwenden Sie die gespeicherte Prozedur ml_add_property, um die Eigenschaften für diese Funktion festzulegen. Siehe „ml_add_property-Systemprozedur“ [*MobiLink - Serveradministration*].

- **Option -ts für die Ereignisprotokollierung** Verwenden Sie die neue Option -ts mit den Dienstprogrammen mlsrv16, dbmlsync und dblsn, um eine Protokollierungssitzung einzurichten.

Siehe:

- „mlsrv16-Option -ts“ [*MobiLink - Serveradministration*]
- „dbmlsync-Option -ts“ [*MobiLink - Clientadministration*]
- „dblbn-Option -ts“ [*MobiLink - Serverinitiierte Synchronisation*]

Konsolidierte Datenbanken

Die folgende Liste enthält die Erweiterungen zur Unterstützung konsolidierter Datenbanken für MobiLink in SQL Anywhere Version 16.0.

Weitere Hinweise zu empfohlenen Treibern finden Sie unter <http://www.sybase.com/detail?id=1011880>.

- **Unterstützung für konsolidierte SQL Anywhere 16-Datenbanken** Der MobiLink-Server unterstützt nun konsolidierte SQL Anywhere 16-Datenbanken. Siehe „[Konsolidierte SQL Anywhere-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für SAP HANA 1.0-Datenbankserver** Der MobiLink-Server unterstützt nun SAP HANA-Datenbankserver, Version 1.00.32 und später. Siehe „[Konsolidierte SAP HANA-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für konsolidierte Datenbanken unter SAP Sybase IQ 15.4** Der MobiLink-Server unterstützt nun konsolidierte Datenbanken, die auf SAP Sybase IQ-Servern der Version 15.4 (ESD#1 und höher) laufen. Siehe „[Konsolidierte SAP Sybase IQ-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für konsolidierte Datenbanken Microsoft SQL Server 2012** Der MobiLink-Server unterstützt nun konsolidierte Datenbanken, die auf Microsoft SQL Server 2012 laufen. Diese Unterstützung ist auf MobiLink-Server unter Microsoft Windows beschränkt. Siehe „[Konsolidierte Microsoft SQL Server-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für konsolidierte Datenbanken unter MySQL Server 5.5.X (wobei X 16 oder höher sein muss)** Der MobiLink-Server unterstützt nun konsolidierte Datenbanken, die auf MySQL-Servern der Version 5.5.x laufen, wobei x 16 oder höher sein muss. Siehe „[Konsolidierte MySQL-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für konsolidierte Datenbanken unter Adaptive Server Enterprise 15.7** Der MobiLink-Server unterstützt nun konsolidierte Datenbanken, die auf einem Server mit Sybase ASE 15.7 laufen. Siehe „[Konsolidierte Adaptive Server Enterprise-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für konsolidierte IBM DB2 LUW-Datenbanken** Der MobiLink-Server unterstützt nun konsolidierte IBM DB2 LUW 10.1-Datenbanken. Siehe „[Konsolidierte IBM DB2 LUW-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung für konsolidierte Oracle 11g-Datenbanken** Der MobiLink-Server unterstützt nun konsolidierte Oracle 11g-Datenbanken. Siehe „[Konsolidierte Oracle-Datenbank](#)“ [[MobiLink - Serveradministration](#)].

MobiLink-Server

Die folgende Liste enthält die in SQL Anywhere Version 16.0 eingeführten Erweiterungen des MobiLink-Servers.

Neue mlsrv16-Funktionen

- **MobiLink-Server unterstützt nun mehrere Netzwerk-Worker-Threads** Der MobiLink-Server unterstützt nun die gleichzeitige Verarbeitung seiner Netzwerkdatenströme durch mehrere Netzwerk-Worker-Threads. Legen Sie Datenstrom-Threads mithilfe der neuen mlsrv16-Option -wn fest. Siehe „[mlsrv16-Option -wn](#)“ [[MobiLink - Serveradministration](#)].
- **header_limit-Option für mlsrv16-Option -x** Mithilfe der neuen header_limit-Option für die mlsrv16-Option -x können Sie die maximale Header-Datenmenge in einer Anforderung festlegen. Sie

kann mit den Protokollen HTTP, HTTPS und OE verwendet werden. Siehe „[mlsrv16-Option -x](#)“ [[MobiLink - Serveradministration](#)].

- **trusted_certificates-Option für mlsrv16-Option -x** Wenn die Verschlüsselung aktiviert ist, ermöglicht die trusted_certificates-Option die Validierung von Clients. Verwenden Sie diese Option mit der NetworkData.ClientCertificates-API, um MobiLink stärker in Ihre unternehmensweite Zertifikateinfrastruktur zu integrieren.

Siehe:

- „[mlsrv16-Option -x](#)“ [[MobiLink - Serveradministration](#)]
- [NetworkData-Schnittstelle](#) [[MobiLink-Server Java](#)] [[MobiLink - Serveradministration](#)]
- [NetworkData-Schnittstelle](#) [[MobiLink-Server-API für .NET](#)] [[MobiLink - Serveradministration](#)]

Neue MobiLink-Skriptfunktionen

- **Neuer benannter Systemparameter authentication_message** Der MobiLink-Server unterstützt nun den neuen benannten Systemparameter authentication_message. Dieser benannte Parameter ist in allen Ereignissen der Benutzerauthentifizierung verfügbar, einschließlich der Skripten authenticate_user, authenticate_user_hashed und authenticate_parameters.

Siehe:

- „[authenticate_user](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)]
- „[authenticate_user_hashed](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)]
- „[authenticate_parameters](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)]
- **Neuer benannter Systemparameter script_version** Der MobiLink-Server unterstützt nun den neuen script_version-Parameter, um festzulegen, dass der MobiLink-Server die Skriptversionszeichenfolge für die aktuelle Synchronisation an diesen Parameter übergibt. Dieser benannte Parameter ist in allen Ereignissen der MobiLink-Serversynchronisation verfügbar, mit Ausnahme der Ereignisse begin_connection, begin_connection_autocommit und end_connection. Siehe [Benannte Skriptparameter](#) [[MobiLink - Serveradministration](#)].

MobiLink-Serverdienstprogramme

Die folgende Liste enthält die in SQL Anywhere Version 16.0 eingeführten Erweiterungen der MobiLink-Serverdienstprogramme.

- **Neue Option -r für Dienstprogramm mluser** Verwenden Sie diese Option mit der mluser-Option -u, um den Synchronisationszustand für den angegebenen Benutzernamen und die angegebene entfernte ID zurückzusetzen. Siehe „[Dienstprogramm für die MobiLink-Benutzerauthentifizierung \(mluser\)](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -n für Dienstprogramm mluser** Verwenden Sie diese Option, um einen MobiLink-Benutzer für die LDAP-Benutzerauthentifizierung zu registrieren. Siehe „[Dienstprogramm für die MobiLink-Benutzerauthentifizierung \(mluser\)](#)“ [[MobiLink - Serveradministration](#)].

- **Neues Dienstprogramm zum Stoppen des MobiLink-Arbiters** Verwenden Sie das neue Dienstprogramm `mlarbstop`, um den MobiLink-Arbiterserver zu stoppen. Siehe „Dienstprogramm zum MobiLink-Arbiters-Stop (mlarbstop)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -sv für mlreplay** Verwenden Sie die neue Option -sv für `mlreplay`, um Skriptversionen in einer Wiedergabesitzung durch benutzerdefinierte Werte zu ersetzen. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)].
- **Neues Skriptversionsfeld der Option -sci für mlreplay** Das neue Skriptversionsfeld ist das letzte Feld in der kommagetrennten Datei für simulierte Clientinformationen, das mit der Option -sci für `mlreplay` angegeben wird. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)].
- **Neue Einstellungen des Verfügbarkeits-Timeouts für mlreplay** Das MobiLink Replay-Dienstprogramm (`mlreplay`) kann das Verfügbarkeits-Timeout mithilfe der Datenstromoptionen ändern, die mit der `mlsrv16`-Option -x in der Befehlszeile angegeben werden. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)].

MobiLink-Profiler

Die MobiLink-Profiler liefert detaillierte Informationen zur Performance Ihrer Synchronisationen. Zusätzlich zu den Zeitangaben auf Phasenebene, die im MobiLink-Monitor verfügbar waren, ermöglicht der MobiLink-Profiler eine detailliertere Performanceanalyse und Engpassidentifizierung über Zeitangaben auf Ereignisebene, Stichproben aus Ereignissen und Phasen sowie Blockierungsinformationen aus der konsolidierten Datenbank, wenn der MobiLink-Server zu lang laufende SQL-Anweisungen erkennt.

Alle Ergebnisse der Profilerstellung werden in einer SQL Anywhere-Datenbank gespeichert, die zurück in den MobiLink-Profiler geladen oder von einem anderen Datenbankclient abgefragt werden kann. Der MobiLink-Profiler übernimmt automatisch das Erstellen, Starten und Stoppen der Datenbank sowie das Herstellen der Verbindung.

Siehe „MobiLink-Profiler“ [[MobiLink - Serveradministration](#)].

MobiLink-Monitor

Der MobiLink-Monitor wurde entfernt. Mit dem MobiLink-Profiler können Sie während der Entwicklungs- und Testphase detaillierte Performanceinformationen zu Synchronisationen abrufen. Verwenden Sie in der Produktion den SQL Anywhere-Monitor für die Überwachung eines MobiLink-Deployments. Siehe „MobiLink-Profiler“ [[MobiLink - Serveradministration](#)] und „SQL Anywhere-Monitor“ [[SQL Anywhere Server - Datenbankadministration](#)].

MobiLink-Clients

Die folgende Liste enthält die in SQL Anywhere Version 16.0 eingeführten Erweiterungen für MobiLink-Clients.

- **RBAC (Role-Based Access Control - rollenbasierte Zugriffssteuerung) und Synchronisation** Im neuen rollen- und privilegbasierten Sicherheitsmodell von SQL Anywhere 16.0 müssen dem Benutzer die Rollen SYS_RUN_REPLICATION_ROLE zum Ausführen der Synchronisation und SYS_REPLICATION_ADMIN_ROLE für administrative Aufgaben im Zusammenhang mit der Synchronisation erteilt werden. Jede dieser Rollen umfasst verschiedene Systemprivilegien. Sie können Rollen und Privilegien erteilen und entziehen, um mehr Kontrolle über die Sicherheit Ihrer Synchronisationsumgebung zu erreichen.

Siehe:

- „Sicherheitshinweise zu rollenbasierter Zugriffssteuerung und Synchronisation“ [[MobiLink - Clientadministration](#)]
- Neues Sicherheitsmodell: RBAC (Role-Based Access Control - rollenbasierte Zugriffssteuerung) auf Seite 1
- „Benutzersicherheit (Rollen und Privilegien)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „GRANT ROLE SYS_REPLICATION_ADMIN_ROLE-Anweisung [[MobiLink](#)] [[SQL Remote](#)]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „GRANT ROLE SYS_RUN_REPLICATION_ROLE-Anweisung [[MobiLink](#)] [[SQL Remote](#)]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „REVOKE ROLE SYS_REPLICATION_ADMIN_ROLE-Anweisung [[MobiLink](#)] [[SQL Remote](#)]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „REVOKE ROLE SYS_RUN_REPLICATION_ROLE-Anweisung [[MobiLink](#)] [[SQL Remote](#)]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Neue trusted_certificate_name-Netzwerkprotokolloption des MobiLink-Clients** Mit der trusted_certificate_name-Protokolloption können Sie vertrauenswürdige Stammzertifikate angeben, die in der entfernten Datenbank gespeichert sind: Diese Option ist nur für SQL Anywhere-Clients verfügbar. Siehe „trusted_certificate_name“ [[MobiLink - Clientadministration](#)].
- **HTTP-Performanceverbesserungen** MobiLink-Clients geben jetzt weniger HTTP-Anforderungen aus als in früheren Versionen.

Änderungen der Dbmlsync-C++-API

- **Neue erforderliche Privilegien für die Connect-Methode** Die Privilegien, die für die in der Connect-Methode angegebene Benutzer-ID erforderlich sind, haben sich geändert. Siehe [DbmlsyncClient.Connect-Methode](#) [[Dbmlsync C++](#)] [[MobiLink - Clientadministration](#)].
- **DBSC_Event-Struktur hat ein neues Ganzzahlfeld "val3"** Die DBSC_Event-Struktur hat ein neues Ganzzahlfeld "val3". Die Bedeutung des Werts im val3-Feld hängt vom Typ des Ereignisses ab. Siehe [DBSC_Event-Struktur](#) [[Dbmlsync C++](#)] [[MobiLink - Clientadministration](#)].
- **DBSC_EVENTTYPE_DOWNLOAD_COMMITTED hat nun zwei Parameter** In der DBSC_Event-Struktur für das Ereignis enthält "val1" die Anzahl der während des Downloads eingefügten oder aktualisierten Zeilen und "val2" die Anzahl der durch den Download gelöschten Zeilen.
- **DBSC_EVENTTYPE_UPLOAD_START-Ereignis wurde hinzugefügt** Das neue DBSC_EVENTTYPE_UPLOAD_START-Ereignis wurde hinzugefügt. Das Ereignis wird generiert, wenn dbmlsync beginnt, den Upload an den MobiLink-Server zu senden.

- **DBSC_EVENTTYPE_UPLOAD_SENT-Ereignis wurde hinzugefügt** Das neue DBSC_EVENTTYPE_UPLOAD_SENT-Ereignis wurde hinzugefügt. Das Ereignis wird generiert, wenn dbmlsync das Senden des Uploads an den MobiLink-Server abschließt.
- **DBSC_EVENTTYPE_DOWNLOAD_START-Ereignis wurde hinzugefügt** Das neue DBSC_EVENTTYPE_DOWNLOAD_START-Ereignis wurde hinzugefügt. Das Ereignis wird generiert, wenn dbmlsync mit der Verarbeitung des Downloads beginnt.

Änderungen der Dbmlsync-.NET-API

- **Neue erforderliche Privilegien für die Connect-Methode** Die Privilegien, die für die in der Connect-Methode angegebene Benutzer-ID erforderlich sind, haben sich geändert. Siehe [DbmlsyncClient.Connect-Methode \[Dbmlsync .NET\] \[MobiLink - Clientadministration\]](#).
- **DBSC_Event-Struktur hat ein neues Ganzzahlfeld "val3"** Die DBSC_Event-Struktur hat ein neues Ganzzahlfeld "val3". Die Bedeutung des Werts im val3-Feld hängt vom Typ des Ereignisses ab. Siehe [DBSC_Event-Struktur \[Dbmlsync .NET\] \[MobiLink - Clientadministration\]](#).
- **DBSC_EVENTTYPE_DOWNLOAD_COMMITTED hat nun zwei Parameter** In der DBSC_Event-Struktur für das Ereignis enthält "val1" die Anzahl der während des Downloads eingefügten oder aktualisierten Zeilen und "val2" die Anzahl der durch den Download gelöschten Zeilen.
- **DBSC_EVENTTYPE_UPLOAD_START-Ereignis wurde hinzugefügt** Das neue DBSC_EVENTTYPE_UPLOAD_START-Ereignis wurde hinzugefügt. Das Ereignis wird generiert, wenn dbmlsync beginnt, den Upload an den MobiLink-Server zu senden.
- **DBSC_EVENTTYPE_UPLOAD_SENT-Ereignis wurde hinzugefügt** Das neue DBSC_EVENTTYPE_UPLOAD_SENT-Ereignis wurde hinzugefügt. Das Ereignis wird generiert, wenn dbmlsync das Senden des Uploads an den MobiLink-Server abschließt.
- **DBSC_EVENTTYPE_DOWNLOAD_START-Ereignis wurde hinzugefügt** Das neue DBSC_EVENTTYPE_DOWNLOAD_START-Ereignis wurde hinzugefügt. Das Ereignis wird generiert, wenn dbmlsync mit der Verarbeitung des Downloads beginnt.

SQL-Anweisungen

Die folgende Liste enthält alle SQL-Erweiterungen für SQL Anywhere-Clients und entfernte MobiLink-Datenbanken in SQL Anywhere Version 16.0.

- **Änderungen der ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung** Die ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung akzeptiert nun NULL in der ADDRESS-Klausel, der TYPE-Klausel und der SET SCRIPT VERSION-Klausel. Siehe „[ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Änderungen der ALTER SYNCHRONIZATION USER-Anweisung** Die ALTER SYNCHRONIZATION USER-Anweisung akzeptiert nun NULL in der ADDRESS-Klausel und der TYPE-Klausel. Siehe „[ALTER SYNCHRONIZATION USER-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Änderungen der START SYNCHRONIZATION SCHEMA CHANGE-Anweisung** Die START SYNCHRONIZATION SCHEMA CHANGE-Anweisung akzeptiert nun NULL in der SET SCRIPT_VERSION-Klausel. Siehe „[START SYNCHRONIZATION SCHEMA CHANGE-Anweisung \[MobiLink\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

MobiLink-Plug-In für Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen und Verhaltensänderungen für das MobiLink-Plug-In für Sybase Central in Version 16.0.

Neue Funktionen des MobiLink-Plug-Ins für Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen für das MobiLink-Plug-In für Sybase Central in Version 16.0.

- **Testtool für Synchronisationsmodelle** Mit dem neuen Testtool auf der Registerkarte **Deployment** des Synchronisationsmodells können Sie das Deployment eines Synchronisationsmodells testen. Siehe „[Synchronisationsmodell vor dem Deployment testen](#)“ [*MobiLink - Erste Orientierung*].

Syntaxbeispiele finden Sie unter „[Praktische Einführung: Einführung in MobiLink](#)“ [*MobiLink - Erste Orientierung*].
- **Assistent zum Erstellen eines Projekts wurde vereinfacht** Der **Assistent zum Erstellen eines Projekts** wurde vereinfacht, um das Erstellen eines Projekts zu erleichtern. Zu den wesentlichen Änderungen an diesem Assistenten gehören folgende:
 - Eine konsolidierte Datenbank muss dem Projekt zugeordnet werden.
 - Die Seite **Synchronisationsmodell** wurde entfernt.
 - Beim Erstellen eines Projekts wird automatisch auch ein Synchronisationsmodell erstellt.
 - Jedes Projekt hat nun einen entfernten Datenbanktyp statt eines Typs für jeden entfernten Schemanamen.
- **Vereinfachter Assistent für das Deployment eines Synchronisationsmodells** Der **Assistent für das Deployment eines Synchronisationsmodells** ist nun einfacher und benutzerfreundlicher. Wenn bereits ein Deployment des betreffenden Synchronisationsmodells durchgeführt wurde, bietet der Assistent jetzt die Möglichkeit, beim Deployment des Synchronisationsmodells die Werte aus dem vorherigen Deployment zu verwenden.
- **Vereinfachter Tabellenzuordnungseditor** Der Tabellenzuordnungseditor wurde neu gestaltet. Zu den wesentlichen Änderungen gehören folgende:
 - Die Registerkarten **Spaltenzuordnungen**, **Downloadtyp**, **Löschungen herunterladen**, **Download-Teilmenge**, **Download-Löschen-Teilmenge**, **Konfliktverarbeitung** und **Status** werden nun im Fensterausschnitt **Details** angezeigt.
 - Spalten für die Optionen **Downloadtyp**, **Löschungen herunterladen**, **Download-Teilmenge**, **Download-Löschen-Teilmenge**, **Konflikterkennung** und **Konfliktlösung** werden nicht mehr

- standardmäßig im Fensterausschnitt **Tabellenzuordnung** angezeigt und sind bei Anzeige schreibgeschützt. Sie können durch Rechtsklicken auf den Tabellentitel angezeigt werden.
- Die Registerkarte **Konfliktverarbeitung** im Fensterausschnitt **Details** wird für Konflikterkennungs- und Konfliktlösungsoptionen verwendet und ersetzt die Registerkarte **Konfliktlösung**.
 - Registerkarten für den Fensterausschnitt **Details** werden nun am oberen Rand des Fensters angezeigt statt am unteren.
- **Neues Fenster Eigenschaften der MobiLink-Server-Befehlszeile** Im neuen Fenster **Eigenschaften der MobiLink-Server-Befehlszeile** können Sie MobiLink-Server-Befehlszeilen speichern, um sie in MobiLink-Projekte einzubeziehen.
 - **Möglichkeit zum Löschen von Synchronisationsmodellen, die mithilfe des MobiLink 16.0-Plug-Ins hinzugefügt wurden** Mit dem neuen Kontextmenüelement **Aus der konsolidierten Datenbank entfernen** für Synchronisationsmodelle können Sie ein Synchronisationsmodell und das diesem zugeordnete Schema löschen. Nur Synchronisationsmodelle, die mithilfe des MobiLink 16.0 Plug-Ins für Sybase Central hinzugefügt wurden, können gelöscht werden.
 - **Unterstützung für mehrere Synchronisationsmodelle in der konsolidierten Datenbank** Beim Deployment eines Synchronisationsmodells wird nun das Schema beibehalten, das mit dem zuvor installierten Schema identisch ist. Vor dem Deployment erhalten Sie eine Warnung, wenn die Versionen nicht kompatibel sind. Außerdem werden bei einem erneuten Deployment von Aktualisierungen derselben Skriptversion automatisch Schemata entfernt, die von der neuen Version nicht mehr benötigt werden.
 - **Verbesserte Konfliktverarbeitung für Synchronisationsmodelle** Die gesamte Verarbeitung der Konflikterkennung und -lösung erfolgt nun im upload_update-Skript und Fetch-Vorgänge sind nur noch erforderlich, wenn ein Konflikt auftritt, statt für jede Zeile. Dies führt zu einer erheblichen Verbesserung der Performance, wenn eine große Anzahl von Aktualisierungen verarbeitet werden muss.
 - **Assistent zum Erstellen eines Synchronisationsmodells wurde vereinfacht** Bisher hatte der Assistent optionale Seiten zum Einstellen verschiedener Synchronisationsoptionen. Diese optionalen Seiten wurden entfernt, da die Synchronisationsoptionen auch auf der Registerkarte **Zuordnungen** und im Fenster **Eigenschaften des Synchronisationsmodells** eingestellt werden können.
 - **LDAP-Unterstützung wurde hinzugefügt** Das MobiLink 16-Plug-In umfasst die folgenden Assistenten und Eigenschaftsfenster, um die LDAP-Authentifizierung mit MobiLink zu unterstützen:
 - Assistent zum Erstellen eines LDAP-Servers
 - Assistent zum Erstellen einer Benutzerauthentifizierungsrichtlinie
 - Fenster **Eigenschaften des LDAP-Servers**
 - **Eigenschaften der Authentifizierungsrichtlinie**
 - **MobiLink-Systemobjekte können aus einer Datenbank entfernt werden** Ein neues Popup-Menüelement ist verfügbar, um die MobiLink-Systemeinrichtung aus einer konsolidierten Datenbank zu entfernen. Wenn Sie dieses Element wählen, werden alle MobiLink-Systemobjekte aus der konsolidierten Datenbank entfernt und die Datenbank wird aus dem MobiLink-Projekt entfernt.

- **Dynamische Objekte und Eigenschaften automatisch aktualisieren** Bei der zentralen Administration von entfernten Datenbanken wird die automatische Aktualisierung für Agenten und bereitgestellte entfernte Aufgaben unterstützt.
- **Entfernte Aufgaben können nun in eine Datei exportiert werden** Entfernte Aufgaben können in eine Datei mit der Erweiterung *.mlt* exportiert werden. Siehe „Entfernte Aufgaben exportieren“ [[MobiLink - Serveradministration](#)].
- **Eingeschränkte Unterstützung für den Oracle-Datentyp XMLTYPE** Das MobiLink-Plug-In für Sybase Central bietet eingeschränkte Unterstützung für den Oracle-Datentyp XMLTYPE. Siehe „Einschränkungen von Synchronisationsmodellen“ [[MobiLink - Erste Orientierung](#)].
- **Warnungen zur Spaltenzuordnung** Beim Erstellen eines Synchronisationsmodells oder Hinzufügen einer Tabellenzuordnung zu einem Synchronisationsmodell wird nun eine Warnung ausgegeben, wenn alle Zeichenspalten mit fester Länge in den konsolidierten Tabellen Zuordnungen zu Zeichenspalten mit variabler Länge in den entfernten Tabellen aufweisen. Dies wird nur überprüft, wenn die konsolidierte Datenbank vom Typ Oracle, IBM DB2 LUW, Microsoft SQL Server oder Adaptive Server Enterprise ist.

Verhaltensänderungen von MobiLink

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink, die in Version 16.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Änderungen des MobiLink-Servers

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink-Servern in Version 16.0.

- **COMMIT wird für erfolgreiche end_synchronization-Skripten immer ausgegeben** Den end_synchronization-Skripten wird ein synchronization_ok-Parameter übergeben, um den Status der Synchronisation anzuzeigen (0 bei fehlgeschlagener Synchronisation, 1 bei erfolgreicher Synchronisation). Vor dieser Änderung wurde ein Rollback der end_synchronization-Skripten durchgeführt, wenn der synchronization_ok-Parameter gleich Null war. Wenn alle end_synchronization-Skripten erfolgreich ausgeführt werden, wird nun ein COMMIT ausgeführt, unabhängig vom Wert des synchronization_ok-Parameters.

Um das alte Verhalten zu erreichen, können Sie den Hauptteil eines vorhandenen end_synchronization-Skripts in eine IF-Anweisung einfügen. Beispiel:

```
IF {ml s.synchronization_ok} = 1 THEN
    -- existing script body here
END IF;
```

- **MobiLink-Konnektivität mit einem Webbrowser testen** Wenn Sie HTTP oder HTTPS mit oder ohne Relay Server verwenden, können Sie mit einem Webbrowser überprüfen, ob der MobiLink-Server auf Anforderungen wartet. Der MobiLink-Server antwortet mit einem **404 Not Found**-Fehler, der auch die Hauptversion des MobiLink-Servers enthält. Siehe [MobiLink-Konnektivität](#) [[MobiLink - Serveradministration](#)].

- **32-Bit-MobiLink-Server wird unter Linux nicht mehr unterstützt** Unter Linux wird nur die 64-Bit-Version des MobiLink-Servers unterstützt.
- **32-Bit-MobiLink-Server für Windows wird bei einigen konsolidierten Datenbanken nicht mehr unterstützt** Der 32-Bit-MobiLink-Server für Windows wird bei den folgenden Typen von konsolidierten Datenbanken nicht unterstützt:
 - Sybase Adaptive Server Enterprise
 - SAP HANA
 - SAP Sybase IQ
 - IBM DB2 LUW
 - MySQL
- **Spaltennamen werden nun immer während der Synchronisation gesendet** In MobiLink werden Spaltennamen nun immer während der Synchronisation gesendet. Siehe [Spaltennamen senden auf Seite 8](#).

Änderungen an MobiLink Notifier und Listener

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink Notifier und Listener in Version 16.0.

- **Eingeschränkte Sichtbarkeit von Adresse, Betreff und Inhalt der Benachrichtigung in der Fehlerbehandlungsroutine** Beim Aufrufen der optionalen Notifier-Fehlerbehandlungsroutine werden alle Zeichen in Adresse, Betreff bzw. Inhalt einer Benachrichtigung, die nicht zu den empfohlenen Zeichen gehören, durch Sternchen ersetzt. Siehe „[error_handler-Ereignis](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].
- **SMS-Listener ist nun standardmäßig deaktiviert** Die dblsn-Option -ns wurde entfernt und der SMS-Listener ist nun standardmäßig deaktiviert. Diese Option wird weiterhin von der Software unterstützt, aber bei Angabe ignoriert. Siehe „[Gesichertes Listener-Deployment](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].

Verwenden Sie die Option -ls, um den SMS-Listener zu aktivieren. Siehe „[dblsn-Option -ls](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].

- **UDP-Listener ist nun standardmäßig deaktiviert** Die dblsn-Option -nu wurde entfernt und der UDP-Listener ist nun standardmäßig deaktiviert. Diese Option wird weiterhin von der Software unterstützt, aber bei Angabe ignoriert. Siehe „[Gesichertes Listener-Deployment](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].

Verwenden Sie die Option -lu, um den UDP-Listener zu aktivieren. Siehe „[dblsn-Option -lu](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].

- **Neue dblsn-Option -ls** Verwenden Sie die dblsn-Option -ls, um den SMS-Listener zu aktivieren. Siehe „[dblsn-Option -ls](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].
- **Neue dblsn-Option -lu** Verwenden Sie die dblsn-Option -lu, um den UDP-Listener zu aktivieren. Siehe „[dblsn-Option -ls](#)“ [[MobiLink - Serverinitiierte Synchronisation](#)].

Änderungen des MobiLink-Clients

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink-Clients in Version 16.0.

- **Beispielzertifikate wurden in ein anderes Verzeichnis verschoben** X.509-Beispielzertifikate für die TLS-Kommunikation wurden aus den Verzeichnissen "bin32" und "bin64" im SQL Anywhere-Installationsverzeichnis in das Verzeichnis *C:\Dokumente und Einstellungen\Alle Benutzer\Dokumente\SQL Anywhere 16\Samples\Certificates* verschoben, das in der Dokumentation als "%SQLANYSAMP16%\Certificates" angegeben wird.

Weitere Hinweise finden Sie in der Datei *C:\Dokumente und Einstellungen\Alle Benutzer\Dokumente\SQL Anywhere 16\Samples\Certificates\readme.txt*.

- **Rollen und Privilegien mit Hooks und gespeicherten Prozeduren** Beim Arbeiten mit Hooks und gespeicherten Prozeduren sind aufgrund des neuen rollen- und privilegbasierten Sicherheitsmodells die folgenden Änderungen erforderlich:
 - Benutzer müssen das MANAGE REPLICATION-Systemprivileg haben, um einen Hook erstellen zu können. Siehe „Ereignis-Hooks für SQL Anywhere-Clients“ [[MobiLink - Clientadministration](#)].
 - Um den Zugriff auf die # hook_dict-Tabelle zu gewährleisten, müssen Hooks entweder Eigentum eines Benutzers mit den Systemprivilegien SELECT ANY TABLE und UPDATE ANY TABLE sein oder mit der SQL SECURITY INVOKER-Klausel der CREATE PROCEDURE-Anweisung definiert werden.
 - Um den Zugriff auf die # hook_dict-Tabelle zu gewährleisten, müssen gespeicherte Prozeduren für den skriptgesteuerten Upload entweder Eigentum eines Benutzers mit den Systemprivilegien SELECT ANY TABLE und UPDATE ANY TABLE sein oder mit der SQL SECURITY INVOKER-Klausel der CREATE PROCEDURE-Anweisung erstellt werden. Siehe „Gespeicherte Prozeduren für skriptgesteuerten Upload“ [[MobiLink - Clientadministration](#)].

Änderungen des MobiLink-Plug-Ins für Sybase Central

Im Folgenden finden Sie eine Liste der Verhaltensänderungen für das MobiLink-Plug-In für Sybase Central in Version 16.0.

- **Benachrichtigungsunterstützung in Synchronisationsmodellen wurde entfernt** Die Unterstützung des Synchronisationsmodells für das Einrichten der serverinitiierten Synchronisation über die Registerkarte **Benachrichtigungen** wurde entfernt. Die serverinitiierte Synchronisation wird nun über entfernte Aufgaben in Synchronisationsprojekten unterstützt. Die Registerkarte "Benachrichtigungen" wird nicht mehr angezeigt und das Synchronisationsmodell-Deployment unterstützt die serverinitiierte Synchronisation nicht mehr. Wenn Sie die serverinitiierte Synchronisation über Notifier und Listener verwenden möchten, können Sie dies unabhängig von Ihrem Synchronisationsmodell tun.
- **Neue entfernte Tabellen haben nicht mehr dbo als Eigentümer** Beim Erstellen eines Synchronisationsmodells mit einer konsolidierten Datenbank vom Typ Microsoft SQL Server oder Sybase Adaptive Server Enterprise, deren Tabelleneigentümer **dbo** heißt, und einer neuen entfernten

SQL Anywhere-Datenbank ist **dbo** nicht Eigentümer der entfernten Tabellen. Stattdessen wird der Benutzer der konsolidierten Datenbank als Eigentümer der neuen entfernten Tabellen verwendet.

Nicht mehr empfohlene MobiLink-Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

- **Unterstützung für konsolidierte SQL Anywhere 11.0.1-Datenbanken** SQL Anywhere Version 11.0.1 und früher wird nicht mehr unterstützt.
- **Unterstützung für konsolidierte Datenbanken unter Adaptive Server Enterprise 15.0** Sybase Adaptive Server Enterprise Version 15.0 und früher wird nicht mehr unterstützt.
- **Unterstützung für konsolidierte Datenbanken unter Microsoft SQL Server 2005** Microsoft SQL Server Version 2005 und früher wird nicht mehr unterstützt.
- **Unterstützung für konsolidierte Datenbanken unter SAP Sybase IQ 15.2** SAP Sybase IQ Version 15.2 und früher wird nicht mehr unterstützt.
- **Unterstützung für konsolidierte IBM DB2 LUW-Datenbanken** IBM DB2 LUW 9.5 und früher wird nicht mehr unterstützt.
- **mlstop-Option -f entfernt** Die Befehlszeilenoption -f für mlstop wurde entfernt. Wenn jedoch die Option -f angegeben ist, wird sie von mlstop als Option -h behandelt. Siehe „Dienstprogramm zum Stoppen von MobiLink (mlstop)“ [[MobiLink - Serveradministration](#)].
- **Callbacks aus mlreplay entfernt** Die folgenden Callbacks wurden aus dem MobiLink Replay-Dienstprogramm (mlreplay) entfernt:
 - FreeAllUploadRows
 - GetNumUploadTables
 - GetUploadTable
 - GetNumRows
 - GetRow
- **script_full_path-Feld entfernt** Das script_full_path-Feld wurde aus der a_sync_db-Struktur entfernt. Siehe [a_sync_db-Struktur](#) [[Datenbanktools](#)] [[SQL Anywhere Server - Programmierung](#)].
- **Konflikterkennung auf der Basis von Anzahlen geänderter Zeilen wurde entfernt** Konflikterkennung, die darauf basiert, dass der MobiLink-Server beurteilt, wie viele Zeilen durch ein Upload-Skript geändert wurden, wurde entfernt. Der MobiLink-Server unterstützt die Verwendung eines upload_fetch-Skripts zum Erkennen von Konflikten oder die Option "Keine Konflikterkennung". Siehe „[Konflikterkennung](#)“ [[MobiLink - Serveradministration](#)].
- **Upload-Statistiken conflicted_inserts und conflicted_deletes wurden entfernt** Die Upload-Statistiken conflicted_inserts und conflicted_deletes wurden entfernt.

- **Modus "Erzwungener Konflikt" wurde entfernt** Der bereits zuvor nicht mehr empfohlene Modus "Erzwungener Konflikt" wurde aus MobiLink entfernt.
- **ml_add_column-Systemprozedur wird nicht mehr empfohlen** Die ml_add_column-Systemprozedur wird nicht mehr empfohlen. Spaltennamen werden immer gesendet.
- **ml_column-Systemtabelle wird nicht mehr empfohlen** Die ml_column-Systemtabelle wird nicht mehr empfohlen. Spaltennamen werden immer gesendet.
- **Die MobiLink-Protokolloptionen tls_type und e2ee_type akzeptieren ECC nicht mehr als Option** Siehe [Elliptische Kurvenverschlüsselung \(ECC\) auf Seite 7](#).

Neue Funktionen von QAnywhere

Die Unterstützung für QAnywhere wird nicht mehr empfohlen und wurde entfernt. Um QAnywhere zu verwalten, müssen Sie das QAnywhere-Plug-In in einer früheren Version von Sybase Central verwenden. Weitere Hinweise zu früheren Versionen von QAnywhere finden Sie in http://dcx.sybase.com/index.html#1201en/sqlanywhere_en12/help_top_index.htm.

Neue Funktionen von Relay Server

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Relay Server in Version 16.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Microsoft IIS 8.0 unterstützt** Relay Server kann nun mit Microsoft IIS 8.0 eingerichtet werden. Siehe „Deployment der Relay Server-Komponenten für Microsoft IIS 7.0 unter Windows Server 7.0 oder 8.0“ [\[Relay Server\]](#).
- **Schnellsetup für Microsoft IIS 7.0, 7.5 und 8.0** Eine interaktive Schnellsetup-Funktion, *rs-setup.bat*, wird als Alternative zur manuellen Prozedur bereitgestellt. Die Datei *rs-setup.bat* befindet sich im Verzeichnis `%SQLANY16%\RelayServer\IIS\quicksetup_iis7or8` und führt die folgenden Aufgaben aus:

1. Installiert und konfiguriert IIS.
2. Erstellt eine Demoanwendung, um zu überprüfen, ob der Relay Server betriebsbereit ist.
3. Generiert eine Kurzreferenz.

Siehe „Deployment der Relay Server-Komponenten für Microsoft IIS 6.0 unter Windows Server 2003“ [\[Relay Server\]](#) und „Deployment der Relay Server-Komponenten für Microsoft IIS 7.0 unter Windows Server 7.0 oder 8.0“ [\[Relay Server\]](#).

- **Apache 2.4 unterstützt** Apache 2.4 wird nun unterstützt.

Siehe „Deployment der Relay Server-Komponenten auf Apache unter Linux“ [\[Relay Server\]](#).

- **Schnellsetup für Apache unter Linux** Die Schnellsetup-Funktion führt die folgenden Aufgaben aus:

1. Konfiguriert den Apache-Webserver für den Relay Server. Für diesen Schritt können Sie das Skript *ap-setup.sh* im Verzeichnis *Installationsverzeichnis/RelayServer/quicksetup_apache* ausführen.
2. Erstellt und startet Relay Server-Testdienste. Für diesen Schritt können Sie das Skript *rs-test-setup.sh* im Verzeichnis *Installationsverzeichnis/RelayServer/quicksetup_apache* ausführen.
3. Generiert eine Kurzreferenz.

Siehe „[Deployment der Relay Server-Komponenten auf Apache unter Linux](#)“ [*Relay Server*].

- **Verbesserte Fehlerbehandlung für Outbound Enabler** Outbound Enabler-Fehlermeldungen wurden mit Fehlercodes und detaillierteren Problemlösungsvorschlägen verbessert, die Ihnen bei der Fehlerbehandlung helfen sollen.
- **Verbesserte Datensteuerung für Serverantworten** Es kann vorkommen, dass Shared Memory-Ressourcen aufgrund von Problemen mit übermäßigem Puffern von Serverantworten erschöpft werden. Dies kann an einer großen Anzahl von Clients, langsamen Lesevorgängen der Clients oder großen HTTP-Antworten liegen. Um die Speichernutzung zu verbessern, stellt Relay Server die neue Konfigurationsoption **max_client_buffer = memory size** bereit, mit der Sie die Speicherpuffergröße für jeden Client begrenzen können. Der Standardwert ist 1 MB. Der Höchstwert ist 4 GB. Siehe „[Backend-Farm-Abschnitt](#)“ [*Relay Server*].
- **Überlapptes Cookie erneuern** Wenn ein Client ein überlappendes Standard-Cookie für mehrere gleichzeitige Verbindungen gemeinsam nutzt, können Timeout-Fehler auftreten. Dieses Problem wird im Relay Server-Log angezeigt. Wenn **renew_overlapped_cookie** aktiviert ist, erkennt Relay Server die Cookieüberlappung für die Farm, bei der diese Eigenschaft explizit aktiviert ist, und erneuert das überlappende Cookie durch Erstellen einer neuen Affinitätsbindung. Siehe „[Backend-Farm-Abschnitt](#)“ [*Relay Server*].
- **Verbesserte Proxy-Unterstützung mit automatischem Durchreichen** Proxys nutzen das Durchreichen über HTTPS, indem sie mit einer CONNECT-Anforderung einen Nicht-SSL-Tunnel einrichten. Diese neue Funktion ermöglicht das Streaming von Daten auf Outbound Enabler-Verbindungen mit dem Relay Server über Puffer-Proxys.
- **up_pad_size-Einstellung (Optionsabschnitt in *rs.config*)** Diese Einstellung fügt eine Auffüllung der maximalen Größe einer Übertragungseinheit ein, um die Latenzzeit zu verbessern, wenn über den RSOE-Kanal geringe Anzahlen von kleinen Anforderungen hochgeladen werden. Die Latenzzeit für den Outbound Enabler wird auch bei Serverantworten verbessert. Siehe „[Optionen-Abschnitt](#)“ [*Relay Server*].

- **Entfernte Verwaltung der Relay Server-Logdatei** Die entfernte Verwaltung der Relay Server-Logdatei wird durch die *AdminChannel*-Klasse in *rstool.jar* unterstützt, die Folgendes ermöglicht:
 - Entfernte Aktualisierung und Abruf der Konfiguration
 - Entfernte Extraktion des Relay Server-Logs
 - Lokale Extraktion des Outbound Enabler-Logs
 - Archivierung des Relay Server-Logs

Siehe „Entfernte Verwaltung der Relay Server-Logdatei“ [\[Relay Server\]](#).

Verhaltensänderungen von Relay Server

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von Relay Server, die in Version 16.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Konfigurationsoption zum Begrenzen der Speichernutzung** Die Option **shared_mem = memory size** im Optionsabschnitt der Konfigurationsdatei gibt nun die maximale Shared Memory-Größe an, die der Relay Server für Statusverfolgung und Wachstum nutzen darf. Siehe „Optionen-Abschnitt“ [\[Relay Server\]](#).
- **Relay Server-Statusseite zeigt mehr Details** Die Relay Server-Statusseite wurde erweitert, sodass neben dem systemweiten Status auch der Status für bestimmte Serverfarmen und Backend-Server angezeigt werden kann. Außerdem können Sie festlegen, wie häufig die Statusseite aktualisiert werden soll. Siehe „Relay Server-Statusseite“ [\[Relay Server\]](#).
- **Relay Server-Änderungen protokollieren** Die Relay Server-Protokollierung unterstützt nun Folgendes:
 - Die Logs von Relay Server und Outbound Enabler melden nun Zeitstempel in Millisekundauflösung, wobei das RFC 822-Format mit Ortszeitdifferenz (+/- hhmm) verwendet wird.
 - Beim Verarbeiten von HTTP-Anforderungen mit SAP Passport-Header erhöhen Relay Server und Outbound Enabler die Ausführlichkeitsstufe der Anforderungsverarbeitung entsprechend der erforderlichen Protokollierungsstufe, die im Passport enthalten ist, und fügen außerdem den zugeordneten Logzeilen Suffixe hinzu, um die Schlüsselinformationen des Passports aufzulisten. Siehe „Relay Server-Protokollierung und SAP Passports“ [\[Relay Server\]](#).

Relay Server-Plug-In für Sybase Central

Im Folgenden finden Sie eine Liste der Verhaltensänderungen für das Relay Server-Plug-In für Sybase Central in Version 16.0.

- **Neue Registerkarte Netzwerk im Fenster Eigenschaften der Relay Server-Farm** Das Fenster **Eigenschaften der Relay Server-Farm** enthält die neue Registerkarte **Netzwerk**. Auf dieser Registerkarte können Sie das Senden von Leerzeichenauffüllungspaketen aktivieren, um die Latenzzeit des Netzwerks zu verkürzen.

- **Neue Registerkarte Erweitert im Fenster Eigenschaften der Backend-Serverfarm** Das Fenster **Eigenschaften der Backend-Serverfarm** enthält die neue Registerkarte **Erweitert**. Auf dieser Registerkarte können Sie Folgendes festlegen:
 - **Client-Server-Affinität** Entspricht der bisherigen Option **Affinität** auf der Registerkarte **Allgemein**.
 - **Maximale Client-Puffergröße** Hier können Sie die neue Option **max_client_buffer** setzen.
 - **Überlapptes Cookie erneuern** Wenn ein Clientgerät ein Standard-Cookie für mehrere gleichzeitige überlappende Verbindungen gemeinsam nutzt, können Timeout-Fehler auftreten. Dieses Problem wird im Relay Server-Log angezeigt. Wenn **Überlapptes Cookie erneuern** aktiviert ist, erkennt Relay Server die Cookieüberlappung für die Farm, bei der diese Eigenschaft explizit aktiviert ist, und erneuert das überlappende Cookie durch Erstellen einer neuen Affinitätsbindung.

Siehe auch

- „Optionen-Abschnitt“ [[Relay Server](#)]
- „Affinität“ [[Relay Server](#)]
- „Backend-Farm-Abschnitt“ [[Relay Server](#)]

Nicht mehr empfohlene und nicht mehr weiterentwickelte Relay Server-Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

- **32-Bit-Relay Server für Windows (Microsoft IIS) nicht mehr weiterentwickelt** Die 32-Bit-Version von Relay Server für Windows wird nicht mehr weiterentwickelt und nicht mehr installiert.
- **32-Bit-Relay Server für Apache unter Linux nicht mehr weiterentwickelt** Die 32-Bit-Version von Relay Server für Windows wird nicht mehr weiterentwickelt und nicht mehr installiert.
- **Automatischer Start nicht mehr unterstützt** Die automatische Startoption wird nicht mehr als Eigenschaft im Optionsabschnitt der Relay Server-Konfigurationsdatei unterstützt.
- **ECC** Im Outbound Enabler kann für die `tls_type`-Option nicht mehr "ECC" gewählt werden. Die `tls_type`-Protokolloption wurde aus der Dokumentation entfernt.

Neue Funktionen von UltraLite

Nachstehend finden Sie eine Liste der neuen Funktionen von UltraLite, die in Version 16.0 hinzugefügt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Allgemeine Funktionen

- **Neue Download-Synchronisationsstatistiken sind verfügbar** UltraLite unterstützt nun drei neue Synchronisationsstatistiken für Downloads in der syssyncresult-Systemtabelle: ignored_updates, ignored_deletes und truncate_deletes.

Der ignored_updates-Parameter zeigt die Anzahl der mehrfach vorkommenden Zeilen an, die beim Download empfangen wurden. Der Parameter ignored_deletes zeigt die Anzahl der gelöschten Zeilen an, die beim Download empfangen, aber bereits gelöscht wurden. Der truncate_deletes-Parameter zeigt die Anzahl der Zeilen, die im Download durch einen Kürzungsvorgang gelöscht wurden. Dazu werden im download_delete_cursor-Skript NULL-Spaltenwerte zurückgegeben.

Siehe:

- „syssyncresult-Systemtabelle“ [[UltraLite - Datenbankverwaltung](#)]
 - ul_sync_stats_download-Struktur [[UltraLite C und Embedded SQL-Datentypen](#)] [[UltraLite - C- und C++-Programmierung](#)]
 - ul_sync_stats_upload-Struktur [[UltraLite C- und Embedded SQL-Datentypen](#)] [[UltraLite - C- und C++-Programmierung](#)]
 - „download_delete_cursor-Skripten“ [[MobiLink - Serveradministration](#)]
 - ULSyncProgressData-Klasse [[UltraLite.NET](#)] [[UltraLite - .NET-Programmierung](#)]
- **ZLIB-Komprimierung wird auf BlackBerry-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun die ZLIB-Datenkomprimierung für BlackBerry-Smartphones bei der Synchronisation mit einem MobiLink-Server.

Siehe:

- „Netzwerkprotokolloptionen des MobiLink-Clients“ [[MobiLink - Clientadministration](#)]
 - StreamHTTPParams.setZlibCompression-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
 - StreamHTTPParams.setZlibDownloadWindowSize-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
 - StreamHTTPParams.setZlibUploadWindowSize-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
 - StreamHTTPParams.zlibCompressionEnabled-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- **Ende-zu-Ende-Verschlüsselung (E2EE) wird auf BlackBerry-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun die Ende-zu-Ende-Verschlüsselung für BlackBerry-Smartphones bei der Synchronisation mit einem MobiLink-Server.

Siehe:

- „Netzwerkprotokolloptionen des MobiLink-Clients“ [*MobiLink - Clientadministration*]
- [StreamHTTPParams.getE2eePublicKey-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.setE2eePublicKey-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.E2EE_RSA-Variable \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.getE2eeType-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.setE2eeType-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)

- **HTTP-Authentifizierung wird auf BlackBerry-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun die HTTP-Authentifizierung für BlackBerry-Smartphones.

Kennwörter werden innerhalb von HTTP-Headern angezeigt, aber das HTTPS-Protokoll kann verwendet werden, um die Header zu verschlüsseln und das Kennwort zu schützen.

Siehe:

- [StreamHTTPParams.setHTTPUserIdAndPassword-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.getHTTPUserId-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.getHTTPPassword-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)

- **Benutzerdefinierte HTTP-Header werden auf BlackBerry-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun benutzerdefinierte HTTP-Header für BlackBerry-Smartphones.

Siehe:

- [StreamHTTPParams.addCustomHTTPHeader-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.getCustomHTTPHeaders-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)

- **Index-Hashing wird auf BlackBerry-Smartphones unterstützt** Verwenden Sie die WITH MAX HASH SIZE-Klausel, um die Index-Hash-Größen in einer UltraLite Java Edition-Datenbank anzupassen.

Siehe:

- „Optimale Hash-Größengrenze“ [[UltraLite - Datenbankverwaltung](#)]
- `Connection.OPTION_MAX_HASH_SIZE`-Variable [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- „CREATE TABLE-Anweisung [[UltraLite](#)]“ [[UltraLite - Datenbankverwaltung](#)]
- „ALTER TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE INDEX-Anweisung [[UltraLite](#)]“ [[UltraLite - Datenbankverwaltung](#)]
- „ALTER INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „sysindex-Systemtabelle“ [[UltraLite - Datenbankverwaltung](#)]

- **Performanceverbesserungen der Indizierung für BlackBerry-Smartphones** Performanceverbesserungen wurden für die Indizierung vorgenommen, um die Anzahl der Seiten zu verringern, die beim Festschreiben von Einfügungen, Aktualisierungen und Löschungen geschrieben werden.

Bei einzelnen Einfügungen sind weniger Seitenlese- und -schreibvorgänge erforderlich. Beim Verarbeiten eines Blocks von Einfügungen wird die CPU-Belastung verringert.

Sortieren Sie beim Synchronisieren mit BlackBerry-Smartphones die `download_cursor`- und `download_delete_cursor`-Synchronisationsserverskripten nach Primärschlüssel. Siehe „[Skripten für den Download von Zeilen](#)“ [[MobiLink - Serveradministration](#)].

- **Verbesserungen der Fehlermeldungen für UltraLite auf BlackBerry-Smartphones, wenn sich mehrere Anwendungen gleichzeitig mit einer Datenbank verbinden** Ein `SQLE_FILE_IN_USE`-Fehler wird nun gemeldet, wenn mehr als eine Anwendung versucht, gleichzeitig eine Verbindung mit einer UltraLite Java Edition-Datenbank herzustellen. Siehe „[Einschränkungen für UltraLite- und UltraLite Java Edition-Datenbanken](#)“ [[UltraLite - Datenbankverwaltung](#)].
- **256-Bit-AES-Verschlüsselung und Verschleierung für BlackBerry-Smartphones ist nun integriert** Sie können die Verschlüsselung oder Verschleierung in Ihrer BlackBerry-Anwendung aktivieren, indem Sie die Methode `ConfigPersistent.enableAesDBEncryption` bzw. `ConfigPersistent.enableObfuscation` in der UltraLiteJ-API verwenden.

Die Option `-ek` wurde zu mehreren UltraLite Java Edition-Dienstprogrammen hinzugefügt, damit Sie den Chiffrierschlüssel angeben können, der für den Zugriff auf verschlüsselte Datenbanken benötigt wird.

Siehe:

- „Datenbanksicherheit“ [[UltraLite - Datenbankverwaltung](#)]
- „Dienstprogramme der UltraLite Java Edition“ [[UltraLite - Datenbankverwaltung](#)]
- `ConfigPersistent.enableAesDBEncryption`-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- `ConfigPersistent.enableObfuscation`-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- `ConfigPersistent.setEncryptionKey`-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- `ConfigPersistent.getEncryptionKey`-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- `Connection.changeEncryptionKey`-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]

- **Meldungen zum angepassten Benutzerauthentifizierungsverfahren können nun abgerufen werden** Synchronisationsskripten für das angepasste Benutzerauthentifizierungsverfahren auf dem MobiLink-Server liefern nun zusätzliche Details in Bezug auf die Ergebnisse der Authentifizierung einer Synchronisation.

Eine neue Systemtabelle, `syssyncresult`, wurde zum UltraLite-Verwaltungssystem hinzugefügt, um diese Funktion zu unterstützen. Diese Systemtabelle ist eine Darstellung der `ul_sync_result`-Struktur.

In UltraLite Java Edition kann die `SyncResult.getAuthMessage`-Methode verwendet werden, um die Informationen zum angepassten Benutzerauthentifizierungsverfahren abzurufen.

Siehe:

- „`syssyncresult`-Systemtabelle“ [[UltraLite - Datenbankverwaltung](#)]
 - `ul_sync_result`-Struktur [[UltraLite C- und Embedded SQL-Datentypen](#)] [[UltraLite - C- und C++-Programmierung](#)]
 - `SyncResult.getAuthMessage`-Methode [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- **UltraLite legt nun Zeilen im Cache ab** UltraLite stellt nun eine zusätzliche Caching-Schicht bereit, um die große Speicherkapazität moderner Geräte besser nutzen zu können. Zusätzlich zum Caching von Datenbankseiten im Speicher wird ein Prozentsatz des zugeordneten Speichers für das Caching von einzelnen Zeilen zugewiesen, um einen noch schnelleren Zugriff zu ermöglichen. Die an die Verbindungsparameter für die Cache-Steuerung übergebenen Werte beziehen sich nun auf den für das Caching zugewiesenen Gesamtspeicher, einschließlich Seiten-Cache und Zeilen-Cache.

Siehe:

- „UltraLite-Verbindungsparameter `CACHE_SIZE`“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Verbindungsparameter `CACHE_MIN_SIZE`“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Verbindungsparameter `CACHE_MAX_SIZE`“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Option `cache_allocation`“ [[UltraLite - Datenbankverwaltung](#)]

Plattformen und Geräte

- **RSA TLS- und HTTPS-Synchronisation wird für Pocket PC 2003 unterstützt** Pocket PC 2003 unterstützt nun RSA TLS- und HTTPS-Synchronisation. Die RSA FIPS 140-2-Synchronisation wird nicht unterstützt.

Programmierschnittstellen

UltraLite C/C++

- **Für Upload- und Downloadstatistiken von Synchronisationsdatenströmen werden separate Berichte erstellt** Die `ul_sync_stats`-Struktur wurde in zwei separate Strukturen aufgeteilt, um die Statistikberichte zu verbessern: `ul_sync_stats_download` und `ul_sync_stats_upload`.

Siehe:

- [ul_sync_stats_download-Struktur](#) [UltraLite C und Embedded SQL-Datentypen] [*UltraLite - C- und C++-Programmierung*]
- [ul_sync_stats_upload-Struktur](#) [UltraLite C- und Embedded SQL-Datentypen] [*UltraLite - C- und C++-Programmierung*]
- **Datenstreaming-Unterstützung für Breitzichen (UTF-16/UCS-2) unter iOS** Breitzichenunterstützung unter iOS wurde zu den Streaming-Methoden für Zeichendaten hinzugefügt.

Siehe:

- [ULPreparedStatement.AppendParameterStringChunk-Methode](#) [UltraLite C++] [*UltraLite - C- und C++-Programmierung*]
- [ULResultSet.GetStringChunk-Methode](#) [UltraLite C++] [*UltraLite - C- und C++-Programmierung*]
- [ULResultSet.AppendStringChunk-Methode](#) [UltraLite C++] [*UltraLite - C- und C++-Programmierung*]
- **Anwendungen können den Download-Fortschritt überwachen** Zwei neue Felder, `current_download_row_count` und `total_download_row_count`, wurden zur `ul_sync_status`-Struktur hinzugefügt, damit Anwendungen den Fortschritt eines Downloads überwachen können.

Siehe:

- „Informationen zum Synchronisationsstatus“ [*UltraLite - C- und C++-Programmierung*]
- [ul_sync_status-Struktur](#) [UltraLite C- und Embedded SQL-Datentypen] [*UltraLite - C- und C++-Programmierung*]

UltraLite.NET

- **Anwendungen können den Download-Fortschritt überwachen** Zwei neue Eigenschaften, `CurrentDownloadRowCount` und `TotalDownloadRowCount`, wurden zur `ULSyncProgressData`-Klasse hinzugefügt, damit Anwendungen den Fortschritt eines Downloads überwachen können.

Siehe:

- [ULSyncProgressData.CurrentDownloadRowCount-Eigenschaft](#) [UltraLite.NET] [*UltraLite - .NET-Programmierung*]
- [ULSyncProgressData.TotalDownloadRowCount-Eigenschaft](#) [UltraLite.NET] [*UltraLite - .NET-Programmierung*]

UltraLiteJ

- **Die Berichterstellung für Zeilenstatistiken während der Synchronisation wurde verbessert** Zusätzliche Informationen bezüglich Anzahl und Typen von Vorgängen, die während einer Synchronisation gesendet und empfangen wurden, werden in Berichte aufgenommen.

Sobald die Synchronisation den `SyncObserver.State.RECEIVING_TABLE`-Status für die erste Tabelle erreicht, gibt die `SyncResult.getTotalDownloadRowCount`-Methode die Gesamtzahl der in den Download erwarteten Zeilen zurück.

Die `getSentRowCount`-Methode wurde durch die Methoden `getSentDeletes`, `getSentInserts` und `getSentUpdates` ersetzt.

Die `getReceivedRowCount`-Methode wurde um die Methoden `getReceivedDeletes`, `getReceivedIgnoredDeletes`, `getReceivedIgnoredUpdates`, `getReceivedInserts`, `getReceivedTruncateDeletes` und `getReceivedUpdates` erweitert. Diese neuen Methoden zeigen an, wie der Download in die lokale UltraLite Java Edition-Datenbank übernommen wurde. Die `getReceivedRowCount`-Methode gibt während des Downloads die aktuelle Zeilenanzahl an.

Die `getReceivedRowCount`-Methode gibt nun Langzahlwerte zurück.

Siehe:

- [SyncResult.getTotalDownloadRowCount-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getSentDeletes-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getSentInserts-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getSentUpdates-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedDeletes-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedIgnoredDeletes-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedIgnoredUpdates-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedInserts-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedTruncateDeletes-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedUpdates-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SyncResult.getReceivedRowCount-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Neue Methoden zum Verwalten der Eingabeparameter einer vorbereiteten Anweisung auf Android-Smartphones** Die Methoden `PreparedStatement.getParameterCount` und `PreparedStatement.getParameterType` wurden hinzugefügt, um die Anzahl der Parameter in einer vorbereiteten Anweisung zu ermitteln bzw. den Domänentyp eines Parameters abzurufen.

Siehe:

- [PreparedStatement.getParameterCount-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [PreparedStatement.getParameterType-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Neue Methoden zum Verwalten von Fehlerparametern auf Android-Smartphones** Die Methoden `getParameterCount` und `getParameter` wurden hinzugefügt, um die Anzahl der Fehlerparameter zu ermitteln bzw. einen bestimmten Fehlerparameter zurückzugeben.

Siehe:

- [ULjException.getParameter-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [ULjException.getParameterCount-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SQLInfo.getParameter-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SQLInfo.getParameterCount-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Neue Methode zum Ermitteln der erforderlichen Zeilenanzahl für den Upload auf Android-Smartphones** Die Methode `DatabaseInfo.getNumberOfRowsToUpload(String, int)` wurde hinzugefügt, um die gleichen Funktionen wie die Methode `ULConnection::CountUploadRows(const char * pubList, ul_u_long threshold)` für UltraLite C/C++ bereitzustellen. Siehe [DatabaseInfo.getNumberOfRowsToUpload-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- **Der Status der UltraLite-Datenbankverbindung kann auf Android-Smartphones gemeldet werden** Die `Connection.getState`-Methode wird nun auf Android-Smartphones unterstützt. Siehe [Connection.getState-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- **Warnmeldungen können auf Android-Smartphones abgerufen werden** Verwenden Sie die `Connection.getLastWarning`-Methode zum Abrufen eines `SQLInfo`-Objekts, das wiederum den SQL-Code und die Meldung für jede Warnung abrufen kann, die während der Ausführung der letzten SQL-Anweisung aufgetreten ist.

Siehe:

- [Connection.getLastWarning-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [SQLInfo-Schnittstelle \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Datenbankvalidierung wird auf Android-Smartphones unterstützt** Verwenden Sie die `Connection.validateDatabase`-Methode zum Validieren einer UltraLite-Datenbank auf einem Android-Smartphone. Die neuen Schnittstellen `ValidateDatabaseProgressData` und `ValidateDatabaseProgressListener` ermöglichen es Android-Anwendungen, den Fortschritt des Validierungsvorgangs zu verfolgen.

Siehe:

- [„Eine UltraLite-Datenbank validieren“ \[UltraLite - Datenbankverwaltung\]](#)
- [Connection.validateDatabase-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [ValidateDatabaseProgressData-Schnittstelle \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [ValidateDatabaseProgressData.StatusId-Schnittstelle \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [ValidateDatabaseProgressListener-Schnittstelle \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Chiffrierschlüssel können auf Android-Smartphones geändert werden** Verwenden Sie die `Connection.changeEncryptionKey`-Methode zum Ändern des Chiffrierschlüssels für eine UltraLite-Datenbank auf einem Android-Smartphone.

Siehe:

- „Datenbanksicherheit“ [[UltraLite - Datenbankverwaltung](#)]
- [Connection.changeEncryptionKey-Methode](#) [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]

- **Neu startbare Downloads werden auf Android-Smartphones unterstützt** UltraLite auf Android-Smartphones hat nun die Möglichkeit, Teil-Downloads wieder aufzunehmen, die aufgrund von Kommunikationsfehlern oder bei Abbruch durch das SyncObserver-Objekt fehlschlagen.

Siehe:

- [Connection.rollbackPartialDownload-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncParms.setAdditionalParms-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncParms.getAdditionalParms-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncParms.setKeepPartialDownload-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncParms.getKeepPartialDownload-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncParms.setResumePartialDownload-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncParms.getResumePartialDownload-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [SyncResult.getPartialDownloadRetained-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]

- **UltraLite-Ereignisse werden auf Android-Smartphones unterstützt** UltraLite auf Android-Smartphones unterstützt nun Ereignisse für Tabellenänderungen, Festschreibungen und den Abschluss der Synchronisation. Die ULjEvent-Schnittstelle wurde hinzugefügt, um diese Funktion zu unterstützen.

Eine Android-Anwendung registriert ein Connection-Objekt und wartet auf entsprechende Ereignisse. Ein Connection-Objekt kann Benachrichtigungen über Ereignisse empfangen, die durch Connection-Objekte auf anderen Threads ausgelöst werden.

Siehe:

- [ULjEvent-Schnittstelle](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [Connection.registerForEvent-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [Connection.unregisterForEvent-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [Connection.waitForEvent-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]
- [Connection.cancelWaitForEvent-Methode](#) [[Android](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)]

- **Optionen für "lazy loading" Indizes wurden auf BlackBerry-Smartphones verbessert** In früheren Versionen hatte das Deaktivieren von "lazy loading" (bei Bedarf geladenen) Indizes keine

Auswirkung, wenn die am Zeilen-Punktwert gemessene Bereinigungsgröße größer war als Null. Das "lazy loading" wurde nur dann deaktiviert, wenn auch die entsprechenden Indizes deaktiviert wurden und die am Zeilen-Punktwert gemessene Bereinigungsgröße höchstens gleich Null war.

Wenn Sie die `setRowScoreFlushSize`-Methode verwenden, um die am Zeilen-Punktwert gemessene Bereinigungsgröße auf einen größeren Wert als Null zu setzen, wird nun "lazy loading" als aktiviert markiert. Die Einstellung `FALSE` für die `setLazyLoadIndexes`-Methode setzt nun die am Zeilen-Punktwert gemessene Bereinigungsgröße auf Null.

Siehe:

- [ConfigPersistent.setRowScoreFlushSize-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [ConfigPersistent.setLazyLoadIndexes-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)

Verhaltensänderungen und nicht mehr empfohlene Funktionen von UltraLite

Nachstehend finden Sie eine Liste der nicht mehr empfohlenen Funktionen und Verhaltensänderungen von UltraLite, die in Version 16.0 eingeführt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

Verhaltensänderungen

- **ECC-Unterstützung wurde entfernt** API-Methoden und -Eigenschaften für UltraLite C/C++ und UltraLiteJ im Zusammenhang mit der ECC-Konfiguration wurden entfernt. Siehe [Elliptische Kurvenverschlüsselung \(ECC\) auf Seite 7](#).

- **Visual Studio 2005-Projektdateien für das CustDB-Beispiel wurden entfernt** Die Visual Studio 2005-Projektdateien für das CustDB-Beispiel wurden aus der Software entfernt.

Die Visual Studio 2008-Projektdateien befinden sich nun im Verzeichnis `%SQLANYSAMPI6%\UltraLite\CustDB`.

Sie können ein Upgrade des Visual Studio 2008-Projektdateien für das CustDB-Beispiel durchführen, indem Sie sie in Visual Studio 2010 öffnen.

- **Verschlüsselungssteuerung für BlackBerry-Smartphones ist jetzt integriert** Die `ConfigPersistent.setEncryption`-Methode wurde entfernt. Die Verschlüsselungssteuerung, bei der in der UltraLiteJ-API ein benutzerdefinierter Verschlüsselungscode zum Verschlüsseln einer UltraLite Java Edition-Datenbank angegeben werden musste, wird nicht mehr unterstützt.

- **In UltraLite Java Edition-Datenbanken werden beim Festschreiben von Anweisungen automatisch Checkpoints gesetzt** Die folgenden Methoden wurden aus der UltraLiteJ-API entfernt:
 - ConfigPersistent.getAutoCheckpoint
 - ConfigPersistent.setAutoCheckpoint
 - Connection.checkpoint
- **Unterstützung für nicht beständige Indizes und Speicher ohne Shadow Paging wurde für BlackBerry-Smartphones entfernt** UltraLite Java Edition-Datenbanken können nicht mehr mit nicht beständigen Indizes oder deaktiviertem Shadow Paging erstellt werden.

Die folgenden Methoden wurden aus der UltraLiteJ-API entfernt:

- ConfigPersistent.hasPersistentIndexes
 - ConfigPersistent.setIndexPersistence
 - ConfigPersistent.setShadowPaging
- **Unterstützung für Write-at-End-beständige Speicher wurde für BlackBerry-Smartphones entfernt** UltraLite Java Edition-Datenbanken können nicht mehr als Write-at-End-Datenbanken erstellt werden.

Die folgenden Methoden wurden aus der UltraLiteJ-API entfernt:

- ConfigPersistent.setWriteAtEnd
 - ConfigPersistent.writeAtEnd
- **Die UltraLiteJ-Methoden getEncryptionKey und setEncryptionKey sind synonym mit den Methoden getDatabaseKey bzw. setDatabaseKey** Die neuen Methoden ConfigPersistent.getEncryptionKey und ConfigPersistent.setEncryptionKey wurden zur UltraLiteJ API hinzugefügt.

Siehe:

- [ConfigPersistent.getEncryptionKey-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
 - [ConfigPersistent.setEncryptionKey-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **UltraLite Java Edition-Dienstprogramme wurden umbenannt** Namen von UltraLite Java Edition-Dienstprogrammen bestehen nun nur noch aus Kleinbuchstaben, sodass sie mit den entsprechenden UltraLite-Namen konsistent sind. Siehe „Dienstprogramme der UltraLite Java Edition“ [[UltraLite - Datenbankverwaltung](#)].
 - **Spaltennamen werden nun immer während der Synchronisation gesendet** Sie können das Senden von Spaltennamen nicht mehr steuern. Siehe [Spaltennamen senden auf Seite 8](#).

Entfernte Plattformen

J2ME-Smartphones werden von der UltraLiteJ-API nicht mehr unterstützt.

Entfernte APIs

- **UltraLite für M-Business Anywhere wurde entfernt** Die UltraLite für M-Business Anywhere-API wird nicht mehr unterstützt.
- **Vorherige Implementierung von UltraLite C/C++ wurde entfernt** Die alte Implementierung der UltraLite C/C++-API, die bisher in den Dateien *ulcpp11.cpp* und *uliface.h* definiert war, wird nicht mehr unterstützt.

Neue Funktionen der Administrationstools

Im Folgenden finden Sie eine Liste der Erweiterungen von Administrationstools, die in Version 16.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

JRE 7 als Basis

Die SQL Anywhere-Administrationstools, z.B. Interactive SQL, Sybase Central, das SQL Anywhere-Konsolendienstprogramm (dbconsole) und der Monitor, verwenden und erfordern nun JRE 7 (Java Runtime Environment 1.7.0).

64-Bit-Unterstützung für Linux/Solaris

SQL Anywhere umfasst nun 64-Bit-Administrationstools für Linux und Solaris (64 Bit). Beachten Sie, dass die 32-Bit-Versionen der Administrationstools für Linux- und Solaris-32-Bit-Plattformen sowie die 64-Bit-Version der Administrationstools für Mac OS X weiterhin verfügbar sind.

Änderungen des Fensters Verbinden

Tool "Verbindung testen" für das Fenster Verbinden Mit dem Tool **Verbindung testen** im Fenster **Verbinden** können Sie nun testen, ob die für generische ODBC- und UltraLite-Datenbanken angegebenen Informationen zu einer einwandfreien Verbindung führen.

Performancedaten regelmäßig an das Software-Entwicklungsteam senden

Als Beitrag zur Verbesserung des Produkts bietet die Software die Möglichkeit, Performancedaten automatisch dem Software-Entwicklungsteam zu melden. Bisher wurden Performancedaten nur nach einem Absturz gesendet. Während der Installation wird die Option zum Senden von Performancedaten an das Software-Entwicklungsteam standardmäßig aktiviert. Sie können diese Funktion ebenfalls aktivieren:

- In Interactive SQL durch Klicken auf **Extras » Optionen » Support**
- In Sybase Central durch Klicken auf **Hilfe » SQL Anywhere 16 » Suchfunktion für Updates konfigurieren**
- Mit dem Dienstprogramm für Supportanfragen (dbsupport) durch Angeben der Option `-cc autosubmit`

OEM-Benutzer Standardmäßig ist die Funktion, die automatisch Performancedaten an das Software-Entwicklungsteam sendet, beim Deployment der Administrationstools deaktiviert. Ihre Benutzer können diese Funktion jedoch in den Administrationstools aktivieren bzw. deaktivieren. Um zu verhindern, dass Ihre Benutzer die Einstellung ändern, ob ihre Daten eingesendet werden oder nicht, setzen Sie in der Datei *OEM.ini* die `showPerformanceDataUI`-Option auf `FALSE`.

Damit Ihre Benutzer Performancedaten an SAP senden können, müssen Sie das Dienstprogramm für Supportanfragen (dbsupport) mit der Option -cc autosubmit verwenden.

Siehe „Regelmäßig Performancedaten senden“ [[SQL Anywhere Server - Datenbankadministration](#)].

Neue Funktionen in Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen von Sybase Central in Version 16.0.

Neue Funktionen der Sybase Central-Plug-Ins

Im Folgenden finden Sie eine Liste der neuen Funktionen der Sybase Central-Plug-Ins in Version 16.0.

Neue Funktionen des SQL Anywhere-Plug-Ins

Layouts von Entity-Relationship-Diagrammen importieren und exportieren Sie können Layouts von ER-Diagrammen über Sybase Central importieren und exportieren. Siehe „[Entity-Relationship-Diagramme anzeigen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Login-Richtlinien mehreren Benutzern zuordnen Sie können eine Login-Richtlinie mehreren Benutzern zuordnen, indem Sie im Fensterausschnitt **Benutzer** die Benutzer auswählen und dann auf **Datei » Login-Richtlinie festlegen** klicken. Außerdem können Sie alle Benutzer anzeigen, denen eine bestimmten Login-Richtlinie zugeordnet ist, indem Sie für die Login-Richtlinie das Fenster **Eigenschaften** öffnen und auf die Registerkarte **Benutzer** klicken.

Unterstützung für Textindizes in materialisierten Ansichten Sie können Textindizes in materialisierten Ansichten mit dem SQL Anywhere-Plug-In erstellen und verwalten.

LDAP-Server Sie können LDAP-Server mit dem SQL Anywhere-Plug-In verwalten.

Unterstützung für Rollen und Systemprivilegien Sie können Rollen und Privilegien mit dem SQL Anywhere-Plug-In verwalten.

Zertifikate in der Datenbank speichern Sie können Zertifikate mit dem SQL Anywhere Plug-In hinzufügen und verwalten.

Neue Funktionen des MobiLink-Plug-Ins

Eine Zusammenfassung der Funktionen des MobiLink-Plug-Ins finden Sie unter „[MobiLink-Plug-In für Sybase Central](#)“.

Verhaltensänderungen von Sybase Central und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Sybase Central in Version 16.0.

- **Sybase Central-Version jetzt 16.0** SQL Anywhere 16 enthält die Version 16.0 von Sybase Central.

- Die Dateien *jh.jar* und *binXX/scvw[LL]1600.jar* sind nicht mehr für das Deployment von **Sybase Central** erforderlich. Sybase Central enthält JavaHelp nicht mehr, sodass die folgenden Dateien weder in der Installation enthalten sind noch für das Deployment von Sybase Central benötigt werden: *jh.jar* und *binXX/scvw [LL]1600.jar*.

Verhaltensänderungen des SQL Anywhere-Plug-Ins

Automatische Eingabeaufforderung für `secure_feature_key`, wenn eine Ausnahme durch eine gesicherte Funktion entsteht Beim Ausführen von SQL-Anweisungen, die eine gesicherte Funktion verwenden, werden Sie aufgefordert, den Schlüssel für gesicherte Funktionen anzugeben, bevor der SQL-Code ausgeführt wird. Der Server muss mit einem Schlüssel für gesicherte Funktionen (-sk) gestartet werden, um diese Funktion nutzen zu können.

Mit dem Assistenten zum Erstellen einer Datenbank erstellte Dienstprogrammbefehle anzeigen Der **Assistent zum Erstellen einer Datenbank** enthält eine neue Seite am Ende des Assistenten. Auf dieser Seite wird der Befehl für das Dienstprogramm dbinit angezeigt, der der durch Klicken auf **Fertig stellen** ausgeführten Anweisung entspricht. Klicken Sie auf **Fertig stellen**, um den Dienstprogrammbefehl auszuführen.

Alternativ können Sie den Befehl in die Zwischenablage kopieren, auf "Abbrechen" klicken, um den Assistenten zu beenden, und dann den Befehl an einer Eingabeaufforderung ausführen. Siehe [Von Assistenten generierte SQL-Anweisungen und Dienstprogrammbefehle anzeigen \[SQL Anywhere Server - Datenbankadministration\]](#).

Verhaltensänderungen des MobiLink-Plug-Ins

Eine Zusammenfassung der Änderungen am MobiLink-Plug-In finden Sie unter „[MobiLink-Plug-In für Sybase Central](#)“ auf Seite 47.

Neue Funktionen von Interactive SQL

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Interactive SQL in Version 16.0 hinzugefügt wurden.

Unterstützung für SAP HANA-Datenbanken Verwenden Sie die Befehlszeilenoption **-hana** für das Interactive SQL-Dienstprogramm (dbisql), um eine Verbindung mit einer SAP HANA-Datenbank herzustellen. Alternativ können Sie in Interactive SQL im Fenster **Verbinden** auf **Datenbanktyp ändern** und anschließend auf **SAP HANA** klicken.

Textvervollständigung im SQL-Editor, das Durchsuchen von Tabellen- und Spaltennamen, das Durchsuchen von Prozedurnamen und Quellen sowie Importieren und Exportieren werden unterstützt.

Die folgenden Interactive SQL-Anweisungen werden nicht unterstützt: CONNECT-Anweisung, SET CONNECTION-Anweisung und START SERVER-Anweisung.

Die Plananzeige für SAP HANA ist neu, funktioniert aber ähnlich wie die anderen Plananzeigen. Der **Spatial Viewer** wird nicht unterstützt.

OEMs können den Interactive SQL-Schnelllader deaktivieren Sie können die Schnelllader-Funktion für Interactive SQL in der Datei *OEM.ini* deaktivieren und verhindern, dass Benutzer sie wieder aktivieren. Siehe „[Konfiguration der Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].

OEMs können festlegen, wie Ergebnisse in Interactive SQL angezeigt werden sollen In der Datei *OEM.ini* können Sie mit den folgenden Optionen festlegen, wie Ergebnisse angezeigt werden sollen: `defaultShowResultsForAllStatements` und `defaultShowMultipleResultSets`. Siehe „[Konfiguration der Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].

Numerische Werte werden in als Text formulierten Ergebnismengen rechtsbündig angezeigt Ergebnismengen werden beim Ausführen als Konsolenanwendung immer als Text angezeigt sowie beim Ausführen als Anwendung mit Fenstern, wenn die Option **Ergebnisse als Text anzeigen** ausgewählt ist. Bisher wurden alle Werte linksbündig angezeigt. Nun werden numerische Werte rechtsbündig angezeigt. Dadurch ist es einfacher, die Werte zu vergleichen.

Im Fehlerfenster werden Zeilennummern angezeigt Wenn das Ausführen einer Anweisung einen Fehler verursacht, wird im eingeblendeten Fehlerfenster nun die Zeilennummer des Fehlers angezeigt.

Die Textvervollständigungsfunktion zeigt Parameterlisten Die Option **Tool-Tipps anzeigen** im Fenster **Optionen** in Interactive SQL zeigt nun während der Eingabe für Prozeduren, Funktionen und räumliche Methoden Parameterlisten. Schließen Sie die Parameterliste, indem Sie Esc drücken.

Die Textvervollständigungsfunktion fügt automatisch schließende Klammern, Apostrophe und Anführungszeichen ein Die Textvervollständigungsfunktion fügt das schließende Satzzeichen ein, wenn Sie eine öffnende Klammer, einen Apostroph oder ein Anführungszeichen eingeben. Um das Einfügen der schließenden Satzzeichen zu steuern, klicken Sie auf **Extras » Optionen » Editor » Textvervollständigung**.

Der Import-Assistent kann Spaltennamen aus importierten Textdateien lesen Der **Import-Assistent** kann eine Liste von Spaltennamen aus der ersten Zeile der Datei lesen. Diese Option ist nur beim Importieren von begrenzten Textdateien verfügbar.

Verhaltensänderungen von Interactive SQL und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Interactive SQL in Version 16.0.

ON-Klausel der START DATABASE-Anweisung nicht mehr empfohlen

Mit der START DATABASE-Anweisung können nur Datenbanken auf dem aktuellen Server gestartet werden. Die ON-Klausel der START DATABASE-Anweisung wird nicht mehr empfohlen. Bisher konnten Sie diese Klausel in Interactive SQL verwenden, um einen zu verwendenden Datenbankserver anzugeben. Siehe „[START DATABASE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Neue Funktionen des SQL Anywhere-Monitors

Im Folgenden finden Sie eine Liste der neuen Funktionen, die dem SQL Anywhere-Monitor in Version 16.0 hinzugefügt wurden.

- **JRE 7 als Basis** Der SQL Anywhere-Monitor verwendet und erfordert nun JRE 7 (Java Runtime Environment 1.7.0).
- **Einstellungen für die E-Mail-Alarmbenachrichtigung** Administratoren können nun mit der Option **Einstellungen für die E-Mail-Alarmbenachrichtigung** im Fenster **Administration** Benutzern Ressourcen zuordnen.

Verhaltensänderungen des SQL Anywhere-Monitors

Im Folgenden finden Sie eine Liste der Änderungen am SQL Anywhere-Monitor in Version 16.0.

Änderungen beim Löschen von Metriken aus dem Monitor

Metriken müssen regelmäßig aus dem Monitor gelöscht werden, um den belegten Speicherplatz in Grenzen zu halten. Standardmäßig werden Metriken, die älter sind als 5 Tage, im Rahmen des Wartungsplans täglich gelöscht.

Bisher konnten Sie die Metriken für unbegrenzte Zeit speichern und Sie konnten steuern, wie viel Speicherplatz die Metriken belegen. Nun beträgt der Zeitraum, für den Metriken gespeichert werden können, 2 Wochen, und Sie können nicht festlegen, wie viel Speicherplatz die Metriken belegen. Außerdem können Metrikdaten nicht in eine neue Version des SQL Anywhere-Monitors migriert werden. Die Option -c für das Migrator-Dienstprogramm wurde entfernt. Siehe „[Sichern des Monitors](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Änderung bei den Anmeldeinformationen (Datenbankbenutzer-ID und Kennwort)

Beim Hinzufügen einer Datenbankressource zum Monitor werden Sie aufgefordert, die Anmeldeinformationen (Benutzer-ID und Kennwort) für die Datenbank anzugeben. Beim Hinzufügen einer Datenbank der Version 16 müssen Sie die Anmeldeinformationen für einen Benutzer angeben, der die SYS_SAMONITOR_ADMIN_ROLE-Rolle hat. Ebenso müssen Sie bei der Fehlerbehebung in einer Datenbankressource der Version 16 die Anmeldeinformationen für einen Datenbankbenutzer mit denselben Rollen angeben.

Wenn Sie aufgefordert werden, die Benutzer-ID und das Kennwort einzugeben, um eine Verbindung zu einer Datenbank zu schließen, müssen Sie die Anmeldeinformationen für einen Benutzer mit DROP CONNECTION-Systemprivileg angeben.

Bei Ressourcen von Datenbanken vor Version 16.0 können Sie weiterhin die Anmeldeinformationen für einen Datenbankbenutzer mit DBA-Berechtigung angeben. Siehe „[Datenbankressourcen hinzufügen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Neue Funktionen der SQL Anywhere-Konsole

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Version 16.0 zum SQL Anywhere-Konsolendienstprogramm hinzugefügt wurden.

- **JRE 7 als Basis** Das SQL Anywhere-Konsolendienstprogramm (dbconsole) verwendet und erfordert nun JRE 7 (Java Runtime Environment 1.7.0).

- **Optionen für Protokollierung von Anforderungen aktivieren** Auf der Registerkarte **Konsole** des Fensters **Optionen** unterstützt die Option **Protokollierung von Anforderungen aktivieren** nun die folgenden Optionen:

Option	Beschreibung
Verbindungsblockierungen	Aktiviert die Protokollierung von Details, die zeigen, wenn eine Verbindung an einer anderen Verbindung blockiert bzw. entblockiert ist.
Hostvariablen	Aktiviert die Protokollierung von Hostvariablenwerten. Die für SQL aufgelisteten Informationen werden ebenfalls protokolliert.
Pläne	Aktiviert die Protokollierung von Ausführungsplänen (Kurzform). Ausführungspläne für Prozeduren werden ebenfalls erfasst, wenn die Option Prozeduren aktiviert ist.
Prozeduren	Aktiviert die Protokollierung von Anweisungen, die innerhalb von Prozeduren und benutzerdefinierten Funktionen ausgeführt werden.
SQL	Aktiviert die Protokollierung der Anweisungen START DATABASE, STOP DATABASE, STOP SERVER, EXECUTE IMMEDIATE, COMMIT, ROLL-BACK und DROP STATEMENT, der Vorbereitung und Ausführung von Anweisungen, Optionseinstellungen, PREPARE TO COMMIT-Vorgänge, Verbindungen und Trennungen, beginnenden Transaktionen, Anweisungen, Cursor-Erläuterungen und Fehlern sowie des Öffnens, Schließens und Wiederaufnehmens von Cursorn.
Trigger	Aktiviert die Protokollierung von Anweisungen, die innerhalb von Triggern ausgeführt werden.
Andere	Aktiviert die Protokollierung von zusätzlichen, nicht in SQL enthaltenen Anforderungstypen, z.B. FETCH und PREFETCH. Durch Angeben von Andere wird automatisch SQL aktiviert. Das Angeben von Andere kann zu einem schnellen Anwachsen der Logdatei führen und sich negativ auf die Server-Performance auswirken.
Logdatei ersetzen, wenn die Protokollierung beginnt	Ersetzt bei Beginn der Protokollierung das vorhandene Anforderungslog durch ein neues (leeres) mit demselben Namen. Ansonsten wird das vorhandene Anforderungslog geöffnet und die neuen Einträge werden an das Ende der Datei angehängt.

Neue Funktionen in Version 12.0.1

Informationen über neue Funktionen und Verhaltensänderungen in den Versionen von SQL Anywhere vor der Version 10 finden Sie unter <http://dcx.sybase.com/html/dbwnde10/dbwnde10.html>.

12.0.1 Ergänzung: Neue Funktionen und Verhaltensänderungen

SQL Anywhere 12.0.1 Dokumentation aktualisiert: Diese Version der Dokumentation von SQL Anywhere 12.0.1 ist neuer als die mit der ersten Freigabe von SQL Anywhere 12.0.1 am 15. Februar 2011 zur Verfügung gestellte Dokumentation. Sie enthält die neuen Funktionen und die Verhaltensänderungen bis einschließlich EBF 3554 (Windows). Insbesondere enthält sie die erforderlichen Änderungen zur Unterstützung der neuen Cloud-Funktionen in der SQL Anywhere OnDemand Edition.

Neue Unterstützung für SQL Anywhere OnDemand Edition

SQL Anywhere OnDemand Edition ist eine Datenverwaltungslösung, die es unabhängigen Softwareanbietern (ISVs) erlaubt, Geschäftsanwendungen in der Cloud abzulegen und Software-as-a-Service (SaaS) anzubieten. ISVs können große Cloud-Anwendungen erstellen, bereitstellen und verwalten.

Einige Erweiterungen und Verhaltensänderungen wurden an SQL Anywhere vorgenommen, um SQL Anywhere OnDemand Edition zu unterstützen. Nachstehend finden Sie eine Liste der wichtigsten Änderungen. Andere Änderungen sind detaillierter in der Dokumentation von SQL Anywhere OnDemand Edition unter <http://dcx.sybase.com/cloud100> beschrieben.

- **Neuer Login-Modus-Wert für Datenbanken in der Cloud** Ein neuer Wert, CloudAdmin, wurde für die Datenbankoption login_mode hinzugefügt. Dieser Login-Modus ist für die interne Verwendung durch die Cloud vorgesehen. Siehe „login_mode-Option“ [*SQL Anywhere Server - Datenbankadministration*].
- **Fenster Verbinden, Erweiterung in Sybase Central und Interactive SQL** Das Fenster **Verbinden** in Sybase Central und Interactive SQL enthält nun die Option **Verbindung mit einer laufenden Datenbank in einer Cloud herstellen** für eine Verbindung zu einer Tenant-Datenbank in der Cloud.
- **Erweiterungen des Fensters ODBC-Konfiguration für SQL Anywhere** Das Fenster **ODBC-Konfiguration für SQL Anywhere** im ODBC-Datenquellen-Administrator enthält nun die Option **Verbindung mit einer laufenden Datenbank in einer Cloud herstellen** für eine Verbindung mit einer Tenant-Datenbank in der Cloud.
- **Erweiterung des NodeType-Verbindungsparameters (NODE)** Der Verbindungsparameter NodeType unterstützt nun MIRROR- und READONLY-Werte:

MIRROR Leitet die Anwendung zur Tenant-Spiegeldatenbank um.

READONLY Leitet die Anwendung zu einer schreibgeschützten Datenbankkopie um: entweder einen Kopieknoten oder die Spiegelkopie. Wenn es keine schreibgeschützten Kopieknoten gibt, ist

READONLY gleichwertig mit der Einstellung MIRROR. Wenn es nur schreibgeschützte Kopieknoten gibt, ist READONLY gleichwertig mit der Einstellung COPY.

Siehe „[Verbindungsparameter NodeType \(NODE\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Verhaltensänderung des Verbindungsparameters NodeType (NODE)** Wenn Sie eine Verbindung zu einer Datenbank in der Cloud herstellen, ist der Standardwert für den NodeType-Verbindungsparameter PRIMARY und Sie werden automatisch zur Mandatenprimärdatenbank weitergeleitet. Wenn NodeType auf DIRECT eingestellt ist, wird keine Umleitung verwendet und die Verbindung ist nur erfolgreich, wenn die Datenbank läuft auf dem angegebenen Host läuft.

Siehe „[Verbindungsparameter NodeType \(NODE\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **DatabaseName-Verbindungsparameter beim Herstellen einer Verbindung zu Cloud-Datenbanken erforderlich** Wenn Sie eine Verbindung zu einer Datenbank in der Cloud herstellen, müssen Sie den DatabaseName-Verbindungsparameter angeben. Es gibt keinen Standarddatenbanknamen für Datenbanken in der Cloud. Siehe „[Verbindungsparameter DatabaseName \(DBN\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Einschränkung für ServerName-Verbindungsparameter in der Cloud** Wenn Sie eine Verbindung zu einer Datenbank in der Cloud herstellen, können Sie den ServerName-Verbindungsparameter nur verwenden, wenn der Verbindungsparameter-NodeType auf DIRECT eingestellt ist. In den meisten Fällen sollten Sie nicht ServerName oder NodeType oder DIRECT festlegen, wenn Sie eine Verbindung zu einem Cloud-Server herstellen. Siehe „[Verbindungsparameter ServerName \(Server\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen von SQL-Anweisungen** Folgende Anweisungen wurden erweitert, um ihre Verwendung in Tenant-Datenbanken in der Cloud zu vereinfachen:
 - „[ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[ALTER SYNCHRONIZATION USER-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[START SYNCHRONIZATION SCHEMA CHANGE-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Wenn eine Datenbank in der Cloud läuft, können einige SQL Anywhere-Anweisungen nicht angewendet werden oder sie haben Einschränkungen. Entsprechende Informationen sind in der Beschreibung der einzelnen Funktionen und Systemprozeduren zu finden.

- **Einschränkungen bezüglich der Ergebnisse von Systemprozeduren und Funktionen bei Ausführung in einer Datenbank in der Cloud** Wenn eine Datenbank in der Cloud läuft, liefern einige der SQL Anywhere-Funktionen und -Systemprozeduren nur Informationen über die aktuelle Tenant-Datenbank. Sie zeigen keine Informationen über andere Tenant-Datenbanken, die auf demselben Cloud-Server laufen. Entsprechende Informationen sind in der Beschreibung der einzelnen Funktionen und Systemprozeduren zu finden.

- **Einige Datenbankoptionen, Serveroptionen und Verbindungsparameter sind nicht für die Verwendung in der Cloud verfügbar** Einige Datenbankoptionen, Server, Optionen und Verbindungsparameter sind nicht für die Verwendung in der Cloud verfügbar. Weitere Hinweise finden Sie in der Beschreibung zu den einzelnen Optionen oder Parameter.

Andere neue Funktionen und Verhaltensänderungen in SQL Anywhere

- **Angabe von IPv6-Adressen** Bei IPv6-Adressen, die eine Portnummer enthalten, müssen Sie die Adresse entweder in eckige Klammern oder in Klammern setzen. Siehe „IPv6-Unterstützung in SQL Anywhere“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen zum Laden von Tabellen** In früheren Versionen konnten Sie keine Tabellenladevorgänge für temporäre Tabellen in einer schreibgeschützten Datenbank durchführen. Diese Einschränkung wurde aufgehoben.

Bislang war der Standardwert für LOAD TABLE für eine Datenbank, die auf einem Primär-, Spiegel- oder Stammknotenserver ausgeführt wurde, WITH FILE NAME LOGGING. Nun ist der Protokollierungsstandardwert für LOAD TABLE für diese Typen von Datenbanken WITH ROW LOGGING.

- **Neue Unterstützung für den Vergleich und die Anpassung von Datenbankschemas** Sie können Sybase Central verwenden, um zwei Datenbanken zu vergleichen. Der Vergleich generiert SQL-Anweisungen, die Sie prüfen können, um die Unterschiede zwischen zwei Datenbanken zu ermitteln. Sie können die SQL-Anweisungen ausführen, um eine Datenbank an die andere anzugleichen. Siehe „Vergleich von Datenbankschemata“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Konvertieren eines Datenbankschemas, um es an ein anderes anzugleichen“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Kontosperre für DBA-Benutzer** Um sicherzustellen, dass DBA-Benutzer nicht unbegrenzt aus der Datenbank ausgesperrt werden, wenn sie ein falsches Kennwort eingeben, werden ihre Konten für eine Minute gesperrt, wenn sie die in ihrer Login-Richtlinie angegebene Anzahl von fehlgeschlagenen Verbindungsversuchen überschreiten.
- **Neue Verzeichnisooption für Datei OEM.ini** Die Option **directory** des Abschnitts **[preferences]** Abschnitt in der Datei OEM.ini gibt das Verzeichnis an, das von den Administrationstools verwendet wird, um benutzerspezifische Konfigurationsdateien zu speichern. Diese Dateien enthalten Informationen zu den Einstellungen und zum Verlauf der Administrationstools. Sie müssen einen voll qualifizierten Verzeichnisnamen angeben, der nicht mit einem Trennzeichen endet. Siehe „Konfiguration der Administrationstools“ [[SQL Anywhere Server - Programmierung](#)].
- **Neue Grenzwerte für TCP-Protokolloptionen Timeout, SendBufferSize und ReceiveBufferSize** Bei der Verwendung von TCP/IP haben die folgenden Protokolloptionen nun Obergrenzen.
 - **ReceiveBufferSize-Protokolloption (RCVBUFSZ)** Die maximale Größe, die Sie für einen Puffer angeben können, der vom TCP/IP-Protokoll-Stack verwendet wird, ist 1 MB.
 - **SendBufferSize-Protokolloption (SNDBUFSZ)** Die maximale Größe, die Sie für einen Puffer angeben können, der vom TCP/IP-Protokoll-Stack verwendet wird, ist 1 MB.

- **Timeout-Protokolloption (TO)** Die maximale Anzahl von Sekunden, die Sie dafür angeben können, wie lange beim Herstellen von Verbindungen auf eine Antwort gewartet wird (TCP/IP), ist 3600 Sekunden.

Wenn Sie Werte oberhalb dieser Grenzwerte angeben, wird ein Fehler zurückgegeben. Siehe „[Verbindungsfehler: %1](#)“ [[Fehlermeldungen](#)].

- **Erweiterungen der Datenbankspiegelung** Die Serveroption -xa unterstützt nun die folgenden Werte:
 - Das Angeben von **DBN=*** bedeutet, dass jede beliebige Datenbank den Server als Arbitr verwenden kann.
 - Sie können die Authentifizierungszeichenfolge weglassen, sodass keine Validierung der von einem Spiegelserver angegebenen Authentifizierungszeichenfolge erfolgt.
 - Wenn nur eine Authentifizierungszeichenfolge angegeben wird, müssen alle Datenbanken diese Authentifizierungszeichenfolge verwenden.

Siehe „[Datenbankserveroption -xa](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

In einem Scale-Out-System mit hoher Verfügbarkeit und Schreibschutz kann ein Kopieknoten als Arbiterserver agieren. Sie müssen ein Upgrade aller Server in Ihrem Datenbankspiegelungssystem vornehmen, um diese Funktion zu verwenden. Siehe „[Verwenden eines Kopieknotens als Arbitr](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Performanceverbesserungen für Anweisungen** Die Performance von Anweisungen, die gespeicherte Prozeduren oder benutzerdefinierte Funktionen verwenden, wurde verbessert.
- **Protokolloption LogFormat (LF) weist auf unbekannte HTTP-Methode für URL hin** Wenn eine Webanforderung fehlschlägt, weil die HTTP-Anforderungsmethode nicht unterstützt wird bzw. weil der URI entweder nicht richtig geformt ist oder darin ein erforderlicher Datenbankname fehlt, liefert die HTTP-Methode (@M) und die HTTP-Version (@V) die Zeichenfolge ??? und der URI (@U) die angegebene Anforderung mit vorangestelltem >>>. Siehe „[LogFormat-Protokolloption \(LF\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue Eigenschaften** Die folgenden Verschlüsselungseigenschaften wurden hinzugefügt:
 - LastCommitRedoPos

Die folgenden Datenbankeigenschaften wurden hinzugefügt:

- LastCommitRedoPos
- LastWrittenRedoPos
- LastSyncedRedoPos

Die folgenden Datenbankserveigenschaften wurden hinzugefügt:

- ProcessId

- **Verbesserte Performance von Anweisungen, die gespeicherte Prozeduren oder benutzerdefinierte Funktionen verwenden** Zur Verbesserung der Performance wurde die Art, wie der Datenbankserver benutzerdefinierte Prozeduren und Funktionen verarbeitet, verbessert. Diese Verbesserung der Performance hängt von der Anzahl der Aufrufe für Prozeduren und benutzerdefinierte Funktionen ab.
- **Verhaltensänderung der ALTER SERVER-Anweisung** In früheren Versionen bewirkten alle Klauseln der ALTER SERVER-Anweisung ein automatisches Festschreiben. Die CONNECTION CLOSE-Klausel bewirkt keine automatische Festschreibung mehr. Siehe „[ALTER SERVER-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der OPENSTRING-Klausel** Die OPENSTRING-Klausel der FROM-Klausel unterstützt nun die Optionen [NOT | AUTO] COMPRESSED und ENCRYPTED. Die Option ENCRYPTED kann verwendet werden, um Dateien zu lesen, die von der UNLOAD-Anweisung erstellt wurden, wenn die ENCRYPTED-Klausel angegeben wurde. Siehe „[FROM-Klausel](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen des Dienstprogramms zur Zertifikatserstellung (createcert)** Eine Reihe von Optionen wurden dem Dienstprogramm zur Zertifikatserstellung hinzugefügt, um die Erstellung von Zertifikaten zu erleichtern. Siehe „[Dienstprogramm zum Erstellen von Zertifikaten \(createcert\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen von Abfrageplan-Zeichenfolgen** Pläne enthalten jetzt bekannte Werte für Ausdrücke und ausführliche Pläne enthalten auch Informationen über den im Cache abgelegten Plan für eine Anweisung.

Wenn Sie die Anwendungsprofilerstellung verwenden, enthalten ausführliche Pläne nun zusätzliche Informationen darüber, wie eine Abfrage optimiert wurde, und über die Prädikate, die in einem partiellen Index-Scan verwendet wurden.

- **Abfrageoptimierung** Es wurde eine Optimierung hinzugefügt, um IN-Prädikate anzunehmen, die für partielle Index-Scans von OR-Prädikaten verwendet werden können, die nicht in AND-Prädikate umgewandelt werden können. Siehe „[Optimierungen während der Abfrageverarbeitung](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Neue Unterstützung für ADO.NET Entity Framework 4.2** SQL Anywhere bietet nun Unterstützung für die Funktion Code First von ADO.NET Entity Framework 4.2. Siehe „[Unterstützung für Entity Framework](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Neue salocation-Option für SetupVSPackage .NET-Assemblies-Installationsprogramm** Die SetupVSPackage-Anwendung gestattet es Ihnen nun, den Speicherort der Installation von SQL Anywhere mit der salocation-Option festzulegen. Siehe „[Deployment eines .NET Clients](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Neue Unterstützung für JDBC 4.0-Treiber für OSGi-Bundles** Der SQL Anywhere JDBC 4.0-Treiber (sajdbc4.jar) enthält nun die richtigen Ladelisten-Informationen, mit denen er als OSGi-Bundle (Open Services Gateway-Initiative) geladen werden kann. Siehe „[JDBC-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].

Verhaltensänderungen von SQL Remote

SQL Remote unterstützt HTTP/HTTPS als ein Nachrichtensystem für SQL Remote-Nachrichten für entfernte Datenbanken. Verwenden Sie den SET REMOTE OPTION-Befehl, um die Datenbank zu konfigurieren.

Siehe:

- „Das HTTP-Nachrichtensystem“ [[SQL Remote](#)]
- „SET REMOTE OPTION-Anweisung [SQL Remote]“ [[SQL Remote](#)]
- „Praktische Einführung: Einrichten eines Replikationssystems unter Verwendung des HTTP-Nachrichtensystems“ [[SQL Remote](#)]
- „Praktische Einführung: Einrichten eines Replikationssystems unter Verwendung des HTTP-Nachrichtensystems mit der konsolidierten Datenbank als Nachrichtenserver“ [[SQL Remote](#)]
- „Praktische Einführung: Einrichten eines Replikationssystems unter Verwendung des HTTP-Nachrichtensystems und mit der konsolidierten Datenbank als Nachrichtenserver über Relay Server“ [[SQL Remote](#)]

Verhaltensänderungen von UltraLite

- **ALTER TABLE...ADD CONSTRAINT erfordert beim Hinzufügen einer Eindeutigkeits-Integritätsregel die Angabe nicht nullwertfähiger Spalten** Verbesserungen der ALTER TABLE-Anweisung erfordern, dass alle Integritätsregel-Spalten nicht nullwertfähig sind, wenn eine neue Eindeutigkeits-Integritätsregel hinzugefügt wird.

Siehe „[ALTER TABLE-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)].

- **MobiLink-Dateiübertragungen werden nun durch Android-Smartphones unterstützt** Die DatabaseManager.createFileTransfer-Methode kann nun dazu verwendet werden, um Dateien zwischen einem Android Client und einem MobiLink-Server zu übertragen.

Siehe:

- [DatabaseManager.createFileTransfer-Methode \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)]
- [FileTransfer-Schnittstelle \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)]
- [FileTransferProgressListener-Schnittstelle \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)]
- [FileTransferProgressData-Schnittstelle \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)]
- **ZLIB-Kompression wird nun von Android-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun die ZLIB-Datenkomprimierung für Android-Smartphones bei der Synchronisation mit einem MobiLink-Server über ein HTTPS-Protokoll.

Die folgenden Methoden stehen für Android-Smartphones zur Verfügung, um die ZLIB-Kompression zu erlauben:

- `StreamHTTTParms.setZlibCompression`
- `StreamHTTTParms.setZlibDownloadWindowSize`
- `StreamHTTTParms.setZlibUploadWindowSize`

- `StreamHTTPParams.zlibCompressionEnabled`

Siehe:

- „Netzwerkprotokolloptionen des MobiLink-Clients“ [[MobiLink - Clientadministration](#)]
- `StreamHTTPParams-Schnittstelle` [UltraLiteJ] [[UltraLite® – Java-Programmierung](#)]

- **Ende-zu-Ende-Verschlüsselung (E2EE) wird nun von Android-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun die Ende-zu-Ende-Verschlüsselung für Android-Smartphones bei der Synchronisation mit einem MobiLink-Server über ein HTTPS-Protokoll.

Die folgenden Methoden stehen für Android-Smartphones zur Verfügung, um E2EE zu unterstützen:

- `StreamHTTPParams.getE2eePublicKey`
- `StreamHTTPParams.setE2eePublicKey`

Siehe:

- „Netzwerkprotokolloptionen des MobiLink-Clients“ [[MobiLink - Clientadministration](#)]
- `StreamHTTPParams-Schnittstelle` [UltraLiteJ] [[UltraLite® – Java-Programmierung](#)]

- **Vertrauenswürdige Zertifikate aus dem Standardspeicher für vertrauenswürdige Zertifikate für die Synchronisation über das HTTPS-Protokoll werden nun von Android-Smartphones unterstützt** Zertifikate werden gemäß den folgenden Prioritätsregeln verwendet:

1. Wenn die `StreamHTTPParams.setTrustedCertificates`-Methode aufgerufen wird, werden die Zertifikate aus der angegebenen Datei verwendet.
2. Wenn die `StreamHTTPParams.setTrustedCertificates`-Methode nicht aufgerufen wird und Zertifikate in der Datenbank mit den Dienstprogrammen ulinit oder ulload gesetzt wurden, werden diese Zertifikate verwendet.
3. Wenn Zertifikate nicht mit der `StreamHTTPParams.setTrustedCertificates`-Methode oder dem Dienstprogramm ulinit oder ulload angegeben werden und Sie Android verwenden, werden Zertifikate vom vertrauenswürdigen Zertifikatspeicher des Betriebssystems gelesen. Der Zertifikatspeicher wird von Webbrowsern bei der Verbindung mit sicheren Webservern über HTTPS verwendet.

Siehe:

- `StreamHTTPParams.setTrustedCertificates-Methode` [UltraLiteJ] [[UltraLite® – Java-Programmierung](#)]

- **Methoden zur Cursor-Bewegung werden nun von Android-Smartphones unterstützt** Die UltraLiteJ-API unterstützt nun alle Navigationsmethoden für SQL-Ergebnismengen, die in anderen UltraLite-APIs verfügbar sind.

Die folgenden Navigationsmethoden sind jetzt für Android-Smartphones verfügbar:

- `ResultSet.afterLast`
- `ResultSet.beforeFirst`
- `ResultSet.first`
- `ResultSet.getRowCount`
- `ResultSet.last`
- `ResultSet.relative`

Siehe:

- „Zeilendaten abrufen“ [[UltraLite® – Java-Programmierung](#)]
- [ResultSet-Schnittstelle \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)]
- **Einige BlackBerry-spezifische ResultSet-Methoden werden jetzt auch von Android-Smartphones unterstützt** Die get-Methoden, die einen Zeichenfolgenparameter erfordern und auf BlackBerry-Smartphones unterstützt werden, sind jetzt für Android-Smartphones verfügbar.

Die folgenden Methoden werden nun von Android unterstützt:

- `ResultSet.getBlobInputStream(Zeichenfolgenname)`
- `ResultSet.getBoolean(Zeichenfolgenname)`
- `ResultSet.getClobReader(Zeichenfolgenname)`
- `ResultSet.getBytes(Zeichenfolgenname)`
- `ResultSet.getDate(Zeichenfolgenname)`
- `ResultSet.getDecimalNumber(Zeichenfolgenname)`
- `ResultSet.getDouble(Zeichenfolgenname)`
- `ResultSet.getFloat(Zeichenfolgenname)`
- `ResultSet.getLong(Zeichenfolgenname)`
- `ResultSet.getSize(Zeichenfolgenname)`
- `ResultSet.getString(Zeichenfolgenname)`
- `ResultSet.getUUIDValue(Zeichenfolgenname)`
- `ResultSet.isNull(Zeichenfolgenname)`

Siehe [ResultSet-Schnittstelle \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)].

- **ResultSetMetaData-Methoden werden jetzt von Android-Smartphones unterstützt** Die folgenden Methoden werden nun von Android-Smartphones unterstützt und liefern Werte, die den für BlackBerry-Smartphones gelieferten Werten stark angenähert sind.

- `ResultSetMetaData.getAliasName(int column_no)`
- `ResultSetMetaData.getDomainName(int column_no)`
- `ResultSetMetaData.getCorrelationName(int column_no)`
- `ResultSetMetaData.getQualifiedName(int column_no)`
- `ResultSetMetaData.getTableColumnName(int column_no)`
- `ResultSetMetaData.getTableName(int column_no)`
- `ResultSetMetaData.getWrittenName(int column_no)`

Siehe [ResultSetMetadate-Schnittstelle \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **Die DatabaseInfo.getPageSize-Methode wird nun von Android-Smartphones unterstützt** Die DatabaseInfo.getPageSize-Methode fragt nun die UltraLite-Datenbank nach der Seitengröße ab.

Siehe [DatabaseInfo.getPageSize-Methode \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **Die DatabaseInfo.getNumberOfRowsToUpload-Methode wird nun von Android-Smartphones unterstützt** Die DatabaseInfo.getNumberOfRowsToUpload-Methode gibt nun die Anzahl der Zeilen zurück, die auf den Upload warten.

Siehe [DatabaseInfo.getNumberOfRowsToUpload-Methode \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **DatabaseInfo.getRelease-Werte, die auf Android-Smartphones zurückgegeben werden, enthalten jetzt die Build-Nummer** Die DatabaseInfo.getRelease-Methode gibt die vollständige Software-Versionsnummer zurück.

Siehe [DatabaseInfo.getRelease-Methode \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **Zusätzliche Optionseinstellungen für das MobiLink-Client-Netzwerkprotokoll werden nun von Android-Smartphones unterstützt** Die Methoden StreamHTTPParms.getExtraParameters und StreamHTTPParms.setExtraParameters zum Angeben und Abrufen einer durch Semikolons getrennten Liste von Netzwerkprotokolloptionen für den MobiLink-Client wurden hinzugefügt. Sie unterstützen auch zusätzliche Optionen, die früher nicht von Android-Smartphones unterstützt wurden.

Siehe:

- [StreamHTTPParams.getExtraParameters-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.setExtraParameters-Methode \[Android\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Fehler-Berichterstellung für die DatabaseManager-Klasse der UltraLiteJ-API wurde verbessert** Ein ULjException-Fehler wird nun mit dem SQLE_NOT_CONNECTED-Fehlercode generiert, wenn die connect- oder createDatabase-Methode aufgerufen wird, nachdem die release-Methode aufgerufen wurde.

Siehe [DatabaseManager.release-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

- **Neu startbare HTTP-Verbindung unterstützt die UltraLiteJ-API** Ist die neu startbare HTTP-Verbindung aktiviert, kann UltraLiteJ Netzwerkunterbrechungen tolerieren, sodass Synchronisationen in unzuverlässigen Netzwerken nicht so häufig fehlschlagen.

Siehe:

- [StreamHTTPParams.setRestartable-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [StreamHTTPParams.isRestartable-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Behandlung von Null ResultSet wurde für die UltraLiteJ-API verbessert** Von der ResultSet.getString-Methode abgerufene Nullwerte werden nun als Null und nicht als leere Zeichenfolge zurückgegeben.

Siehe [ResultSet.getString-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

- **Die Zeilenbegrenzung auf einem BlackBerry-Smartphone aktiviert immer langsame Lade-Indizes** Datenbanken auf BlackBerry-Smartphones, auf die mit aktivierter Zeilenbeschränkung zugegriffen wird, aktivieren nun immer langsame Lade-Indizes.

Siehe:

- [ConfigPersistent.setRowScoreFlushSize-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- [ConfigPersistent.setRowScoreMaximum-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#)
- **Sichere Synchronisation von einem BlackBerry-Smartphone über den BlackBerry Enterprise Server wurde verbessert** Einige BlackBerry-Smartphones, die über das HTTPS-Protokoll mit einem BlackBerry Enterprise Server (BES) synchronisieren, erfordern möglicherweise, dass dem URL-Suffix der Begriff ";EndToEndRequired" hinzugefügt wird, um zu vermeiden, dass eine Synchronisation fehlschlägt.

Siehe [StreamHTTPParams.setURLSuffix-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

- **Die Datenbankseitengrößen können mit dem UltraLite Java Edition-Dienstprogramm zum Laden von Datenbanken mit der Option -z angepasst werden** Die neue Option -z kann verwendet werden, um die Seitengröße (in Byte) einer UltraLite Java Edition-Datenbank festzulegen.

Siehe „Lade-Dienstprogramm für UltraLite Java Edition-Datenbank (uljload)“ [[UltraLite - Datenbankverwaltung](#)].

- **Unterstützung für die Methoden `Connection.getNewUUID` und `UUID.toString` unter UltraLite für M-Business Anywhere** Die Methoden `Connection.getNewUUID` und `UUID.toString` werden wieder eingesetzt.
- **Der qualifizierte Name für eine Tabellenspalte kann mit der UltraLite C++-API erhalten werden** Wenn der `ul_name_type_qualified`-Bezeichner der `ul_column_name_type`-Enumeration gemeinsam mit der `ULResultSetSchema.GetColumnName`-Methode verwendet wird, kann er den zugeordneten qualifizierten Namen für eine Spalte im `ResultSet`-Objekt abfragen.

Siehe:
 - [ul_column_name_type-Enumeration \[UltraLite C++\]](#) [[UltraLite - C- und C++-Programmierung](#)]
 - [ULResultSetSchema.GetColumnName-Methode \[UltraLite C++\]](#) [[UltraLite - C- und C++-Programmierung](#)]
- **Die Pocket PC 2003-Plattform wird nun von der UltraLite C++-API unterstützt** UltraLite wird nun auf der Pocket PC 2003-Plattform unterstützt. Unterstützung für TLS- und HTTPS-Synchronisationsverschlüsselung ist nicht verfügbar.
- **Xcode 4.2-Unterstützung für Beispiele CustDB und Names** Die Beispiele `CustDB` und `Names` werden nun von Xcode 3.2 und 4.2 unterstützt.

Neue MobiLink-Funktionen

- **NetworkData-Klasse zu MobiLink-Server-APIs hinzugefügt** Eine neue `NetworkData`-Klasse wurde den Java- und .NET-Skripterstellungs-APIs hinzugefügt, um Zugriff auf HTTP-Header und clientseitige Zertifikate in einer Anforderung bereitzustellen.

Siehe:
 - [NetworkData-Schnittstelle \[MobiLink-Server Java\]](#) [[MobiLink - Serveradministration](#)]
 - [NetworkData-Schnittstelle \[MobiLink-Server-API für .NET\]](#) [[MobiLink - Serveradministration](#)]
- **collect_network_data-Protokoll wurde der mlsrv12-Option -x hinzugefügt** Verwenden Sie die `collect_network_data`-Protokolloption mit der `mlsrv12`-Option `-x`, um Synchronisationsskripten zu aktivieren, um Informationen von jeder Synchronisation zu lesen. Siehe „[mlsrv16-Option -x](#)“ [[MobiLink - Serveradministration](#)].

Verhaltensänderungen von MobiLink

- **Aktualisierungen nicht synchronisierter Spalten** Aktualisierungen nicht synchronisierter Spalten (außer Zeitstempelspalte oder logische Löschspalten im Synchronisationsmodell) aktualisieren die Zeitstempelspalte nicht mehr, wenn ein Trigger im Synchronisationsmodell für die Verwaltung des Zeitstempels verwendet wird.
- **Menüs für die Tabellenzuordnungsrichtung** Die Menüs für die Tabellenzuordnungsrichtung bleiben aktiviert, nachdem die Zuordnungsrichtung auf "Nicht synchronisiert" geändert wurde. Siehe „[Tabellen- und Spaltenzuordnungen](#)“ [[MobiLink - Erste Orientierung](#)].

- **Option -qi für mlsis und dbmlsync** Die Option -qi ist nun in den Syntaxinformationen für mlsis und dbmlsync enthalten. Siehe „dblsn-Option -qi“ [[MobiLink - Serverinitiierte Synchronisation](#)] und „dbmlsync-Option -qi“ [[MobiLink - Clientadministration](#)].
- **Neue Fehlermeldungen für mlreplay** Die folgenden Fehlermeldungen wurden für das MobiLink-Wiedergabe-Dienstprogramm hinzugefügt:
 - **MLGENREPLAYAPI_FAILED_TO_DETERMINE_CLIENT_TYPE (-5062)** Es konnte nicht ermittelt werden, ob das aufgezeichnete Protokoll von einem dbmlsync-Client stammt.
 - **MLREPLAY_INCOMPATIBLE_RECORDED_PROTOCOL_VERSION (-5075)** %1 ist eine inkompatible aufgezeichneten Protokollversion dieser Version von mlreplay.
 - **MLGENREPLAYAPI_INCOMPATIBLE_RECORDED_PROTOCOL_VERSION (-5076)** %1 ist eine inkompatible aufgezeichneten Protokollversion dieser Version des Replay API-Generators.
- **Erweiterte dbmlsync-Protokollierung** Dbmlsync gibt nun Informationen über das Betriebssystem und die Architektur des Rechners, auf dem es läuft, in das Log aus. Es gibt außerdem die Plattform an, für die das Programm kompiliert wurde. Siehe „SQL Anywhere-Clientlogs“ [[MobiLink - Clientadministration](#)].
- **Unterstützung für SHA2-Zertifikate** MobiLink-Server und MobiLink-Clients unterstützen nun mit SHA2 unterzeichnete Zertifikate. Siehe „trusted_certificates“ [[MobiLink - Clientadministration](#)] und „mlsrv16-Option -x“ [[MobiLink - Serveradministration](#)].

Neue Funktionen von Relay Server

- **Schnelles Deployment von Relay Server unter Windows IIS6, IIS7 und Apache** Relay Server verfügt über eine Reihe von Dienstprogrammen, die Sie Schritt für Schritt durch den Deployment-Prozess für Windows IIS 6.0, Windows IIS 7.0 oder 7.5 und Apache führen. Weitere Hinweise finden Sie unter .
 - „Deployment der Relay Server-Komponenten für Microsoft IIS 6.0 unter Windows Server 2003“ [[Relay Server](#)]
 - „Deployment der Relay Server-Komponenten für Microsoft IIS 7.0 unter Windows Server 7.0 oder 8.0“ [[Relay Server](#)]
 - „Deployment der Relay Server-Komponenten auf Apache unter Linux“ [[Relay Server](#)]
- **Relay Server unterstützt die Identitäts-Weiterleitung** Das SAP Gateway bietet mehrere Methoden für die Authentifizierung von Clients, einschließlich Weiterleitung des X.509-Zertifikats durch vertrauenswürdige Vermittler. Relay Server können Identitätsinformationen vom entfernten Client an das SAP NetWeaver Gateway oder den Web Dispatcher mit HTTP-Headern weiterleiten. Siehe [Eigenschaften im Backend-Farm-Abschnitt](#) [[Relay Server](#)].

Neue Funktionen in SQL Anywhere

Im Folgenden finden Sie eine Liste neuer Funktionen in SQL Anywhere Version 12.0.1.

Weitere Hinweise zu den Änderungen der unterstützten Plattformen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Erweiterungen zur Unterstützung räumlicher Daten** Die folgenden Erweiterungen wurden zur Unterstützung räumlicher Daten in SQL Anywhere vorgenommen:

Verbesserte Performance Für folgende räumliche Vorgänge wurden erhebliche Performance-Verbesserungen erzielt:

- Ladedauer für Formdateien
- Laden von Polygonen und Multipolygonen und Sammlungen mit Polygonen, die zu einem räumlichen Bezugssystem mit gewölbter Erddarstellung gehören
- Laden von komplexen Polygonen und Multipolygonen (z.B. Polygone, die durch viele Ringe oder Ringe mit vielen Punkten definiert sind)
- Auf komplexe Geometrien angewendete räumliche Set-Operationen, wie etwa ST_Union und ST_Intersection
- Räumliche Prädikate, wie z.B. ST_Contains und ST_Intersects
- Räumliche Prädikate, bei denen eine der Geometrien ein Punkt ist
- Räumliche Prädikate, die ST_WithinDistance oder ST_Distance auf indizierte Spalten in einer Geometrie mit gewölbter Erddarstellung anwenden

Interactive SQL: Import-Assistent unterstützt nun Formdateien Der **Import-Assistent** verfügt nun über eine Option für das Importieren von ESRI-Formdateien. Siehe „[Importieren von Daten mit dem Import-Assistenten](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Interactive SQL: Neue FORMAT SHAPEFILE- und SRID-Klausel für die INPUT-Anweisung Die INPUT-Anweisung unterstützt nun das Laden von ESRI-Formdateien mit der FORMAT SHAPEFILE-Klausel. Die SRID-Klausel wurde ebenfalls zur INPUT-Anweisung hinzugefügt, um die zu verwendende SRID beim Laden der Formdatei anzugeben. Siehe „[INPUT-Anweisung \[Interactive SQL\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Neue st_geometry_load_shapefile-Systemprozedur Mit der st_geometry_load_shapefile-Systemprozedur können Sie eine ESRI-Formdatei laden, indem Sie den Namen der Datei, die beim Laden der Daten zu verwendende SRID und den Namen der zu erstellenden Tabelle, in die die Daten geladen werden sollen, angeben. Die Spalten für die Tabelle werden aus den in der Formdatei angegebenen Spaltennamen übernommen. Siehe „[st_geometry_load_shapefile-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Hinweis

Sie müssen ein Upgrade Ihrer Datenbank ausführen, um auf diese neue gespeicherte Prozedur zugreifen zu können.

Neue Datenbankoption zur Angabe der ST_CircularString-Interpolationstoleranz Die st_geometry_interpolation-Option wurde zur Steuerung der Interpolation von ST_CircularString-Geometrien hinzugefügt. Siehe „[st_geometry_interpolation-Option](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Erweiterungen von ST_WithinDistanceFilter Das räumliche Prädikat ST_WithinDistanceFilter wird nun für Geometrien in einem räumlichen Bezugssystem mit gewölbter Erddarstellung unterstützt. Siehe [ST_WithinDistanceFilter-Methode für den ST_Geometry-Datentyp \[SQL Anywhere Server - Unterstützung für räumliche Daten\]](#).

- **Erweiterung der java_class_path-Option** Sie können nun mit der java_class_path-Datenbankoption die Klassen und JAR-Dateien festlegen, die der Systemklassenlader zum Klassenpfad hinzufügen soll, bevor die Java-VM gestartet wird. Diese Option ist nützlich, wenn Ihre Anwendung die Verzeichnisse oder JAR-Dateien angeben muss. Siehe „[java_class_path-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neuer ODBC-Verbindungsparameter MATVIEW** Mit dem MATVIEW-Verbindungsparameter können Sie die Zeichenfolge angeben, die für den Tabellentyp von materialisierten Ansichten zurückgegeben werden soll, wenn die ODBC-Funktion SQLTables ausgeführt wird. Standardmäßig ist der von der SQLTables-Funktion zurückgegebene Wert VIEW. Siehe „[Verbindungsparameter MatView \(MATVIEW\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue Datenbankserveroption -kp** Sie können nun die Option -kp verwenden, um den Serverprinzipal im Standardformat *server-name/hostname@REALM* anzugeben. Die Option -kp aktiviert von Kerberos authentifizierte Verbindungen zum Datenbankserver. Siehe „[Datenbankserveroption -kp](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen der PartnerState-Eigenschaft** Die PartnerState-Eigenschaft gibt nun einen der folgenden Werte zurück, wenn sie mit der DB_PROPERTY-Funktion verwendet wird:
 - **connected** Es gibt eine Verbindung vom aktuellen Server zum angegebenen Server und eine Verbindung vom angegebenen Server zum aktuellen Server.
 - **incoming only** Es gibt eine Verbindung vom angegebenen Server zu diesem Server.
 - **outgoing only** Es gibt eine Verbindung von diesem Server zum angegebenen Server.
 - **disconnected** Es gibt keine Verbindungen zwischen diesen Server und dem angegebenen Server.
 - **NULL** Die Datenbank wird nicht gespiegelt.

Siehe [PartnerState-Datenbankeigenschaft \[SQL Anywhere Server - Datenbankadministration\]](#).

- **Erweiterungen der MirrorServerState-Eigenschaft** Die MirrorServerState-Eigenschaft gibt nun einen der folgenden Werte zurück, wenn sie mit der DB_EXTENDED_PROPERTY-Funktion verwendet wird:
 - **connected** Es gibt eine Verbindung vom aktuellen Server zum angegebenen Server und eine Verbindung vom angegebenen Server zum aktuellen Server.
 - **incoming only** Es gibt eine Verbindung vom angegebenen Server zu diesem Server.
 - **outgoing only** Es gibt eine Verbindung von diesem Server zum angegebenen Server.

- **disconnected** Es gibt keine Verbindungen zwischen diesen Server und dem angegebenen Server.
- **NULL** Die Datenbank wird nicht gespiegelt.

Siehe [MirrorServerState-Datenbankeigenschaft \[SQL Anywhere Server - Datenbankadministration\]](#), [„DB_PROPERTY-Funktion \[System\]“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und [„DB_EXTENDED_PROPERTY-Funktion \[System\]“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

- **Neue sp_forward_to_remote_server-Prozedur** Sie können mit der gespeicherten Prozedur `sp_forward_to_remote_server` einer Anwendung erlauben, eine SQL-Anweisung auf einem Fremdserver auszuführen und durch die Anweisung generierte Ergebnismengen abzufragen. Siehe [„sp_forward_to_remote_server-Systemprozedur“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

Hinweis

Sie müssen ein Upgrade Ihrer Datenbank ausführen, um auf diese gespeicherte Prozedur zugreifen zu können.

- **Neue sa_user_defined_counter_add-Systemprozedur** Mit der Systemprozedur `sa_user_defined_counter_add` können Sie den Wert einer benutzerdefinierten Eigenschaft ändern. Siehe der Systemprozedur `sa_user_defined_counter_add`. Siehe [„sa_user_defined_counter_add-Systemprozedur“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

Hinweis

Sie müssen ein Upgrade Ihrer Datenbank ausführen, um auf diese gespeicherte Prozedur zugreifen zu können.

- **Neue sa_user_defined_counter_set-Systemprozedur** Mit der Systemprozedur `sa_user_defined_counter_set` können Sie den Wert einer benutzerdefinierten Eigenschaft festlegen. Siehe [„sa_user_defined_counter_set-Systemprozedur“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

Hinweis

Sie müssen ein Upgrade Ihrer Datenbank ausführen, um auf diese gespeicherte Prozedur zugreifen zu können.

- **SQLANYAMP12-Umgebungsvariable** Das Installationsprogramm für Unix und Mac OS X setzt nun die SQLANYAMP12-Umgebungsvariable in den Skripten `sa_config` und `sample_config`. Unter Unix kann das Skript `sample_config` verwendet werden, um eine Kopie der Beispiele pro Benutzer zu erstellen. Diese Option ist bei einer Mehrbenutzer-Installation sinnvoll. Bei Einbenutzer-Installationen setzt das Skript `sa_config` SQLANYAMP12 auf `$SQLANY12/samples`. Siehe [„SQLANYAMP16-Umgebungsvariable“ \[SQL Anywhere Server - Datenbankadministration\]](#) und [„sa_config.sh und sample_config32.sh/sample_config64.sh sourcen \[Unix und Mac OS X\]“ \[SQL Anywhere Server - Datenbankadministration\]](#).
- **IN-Parameter vom Datentyp LONG VARCHAR, LONG BINARY und LONG NVARCHAR sind nun in entfernten Prozeduraufrufen zulässig** Ein entfernter Prozeduraufruf kann nun IN-

Parameter vom Datentyp LONG VARCHAR, LONG NVARCHAR und LONG BINARY enthalten. Außerdem sind Parameter vom Datentyp VARCHAR, NVARCHAR oder BINARY nicht mehr auf 255 Byte beschränkt. Siehe „Erstellen von entfernten Prozeduren (Sybase Central)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

- **USING-Klausel der CREATE SERVER-Anweisung kann nun Variablen enthalten** Die USING-Klausel der CREATE SERVER-Anweisung kann jetzt Variablen enthalten. Diese Funktion gestattet es dem Benutzer, dynamische Ferndatenzugriffsserver zu erstellen. Siehe „CREATE SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Ferndatenzugriff kann den SQL Anywhere ODBC-Treiber direkt laden** Sie können einen Fremdserver definieren, damit der Ferndatenzugriff den SQL Anywhere-Treiber direkt lädt und den ODBC-Treibermanager auf Windows- und Unix-Plattformen umgeht. Verwenden Sie beim Definieren des Fremdservers die folgende Syntax, gefolgt vom Rest der Verbindungszeichenfolge:

```
CREATE SERVER remote-server CLASS 'SAODBC' USING 'DRIVER=SQL Anywhere Native;...';
```

Wenn mehrere SQL Anywhere-Fremdserver ohne 'DRIVER=SQL Anywhere Native' definiert sind, dann verwendet der Ferndatenzugriff immer noch einen Treibermanager für die anderen Fremdserver.

- **AT-Klausel der Anweisungen CREATE EXISTING TABLE und CREATE PROCEDURE kann nun Variablen enthalten** Die AT-Klausel für die Anweisungen CREATE EXISTING TABLE und CREATE PROCEDURE können nun Variablen enthalten. Diese Funktion gestattet es dem Benutzer, eine Proxy-Tabelle oder Proxy-Prozedur mehreren entfernten Tabellen und Prozeduren zuzuordnen. Siehe „CREATE EXISTING TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „CREATE PROCEDURE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der RAISERROR-Klausel der Anweisungen MERGE und RAISERROR** Mit der RAISERROR-Anweisung erlaubt es der SQL Anywhere-Datenbankserver einer Anwendung, einen benutzerdefinierten Fehler auszulösen. Der Datenbankserver stellt auch die integrierte, globale Variable SQLCODE bereit, deren Wert geprüft werden kann, um zu ermitteln, welcher Fehler während der Ausführung der letzten Anweisung auf der aktuellen Verbindung ausgelöst wurde. Der Datenbankserver meldet nun die benutzerdefinierte Fehlernummer für SQLCODE anstelle einer festen Fehlermeldung von -631. Siehe „MERGE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „RAISERROR-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **VALIDATE TEXT INDEX-Anweisung** Sie können mit der VALIDATE TEXT INDEX-Anweisung überprüfen, ob die Positionsdaten für die Begriffe im Textindex intakt sind. Falls die Positionsdaten beschädigt sind, wird ein Fehler generiert. Siehe „VALIDATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterte Syntax für die Klauseln TOP und LIMIT** Die Klauseln TOP { ALL | limit } START AT startat und LIMIT limit [OFFSET offset] unterstützen nun einfache arithmetische Ausdrücke für die Argumente limit, offset und startat. TOP unterstützt die Grenze ALL, die anzeigt, dass alle Zeilen nach dem unter startat angegebenen Wert zurückgegeben werden sollen.

Der Höchstwert für (*limit* + *offset*) und (*limit* + *startat* -1) wurde auf 9223372036854775807 = 2⁶⁴-1 erhöht.

Siehe:

- „Zeilenbeschränkungsklauseln in SELECT-, UPDATE- und DELETE-Abfrageblöcken“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*]
- „SELECT-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „DELETE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „UPDATE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]

- **SQL Anywhere OLE DB-Provider unterstützt nun den DBTYPE_DBTIMESTAMPOFFSET-Datentyp** Der SQL Anywhere OLE DB-Provider unterstützt nun den DBTYPE_DBTIMESTAMPOFFSET-Datentyp. DBTYPE_DBTIMESTAMPOFFSET (146) ist ein OLE DB-Typ, der den Datentyp `TIMESTAMP WITH TIME ZONE` (oder `DATETIMEOFFSET`) unterstützt. Die Unterstützung für diesen Datentyp erleichtert die Übertragung von Datentabellen zwischen SQL Anywhere-Datenbanken und anderen Datenbank-Managementsystemen (einschließlich SQL Anywhere).
- **Webdienste unterstützt nun eine verbesserte Kontrolle des HTTP-Umleitungsvorgangs** Die neue `SET REDIR`-Klausel der Anweisungen `CREATE PROCEDURE` und `CREATE FUNCTION` ermöglicht die Kontrolle der maximalen Anzahl von zulässigen Umleitungen und legt fest, welche HTTP-Statusangaben automatisch umgeleitet werden sollen.

Eine Webdienstprozedur, die eine `POST` HTTP-Methode festlegt und den Status 303 empfängt, gibt mit der `GET` HTTP-Methode eine Umleitungsanforderung aus.

HTTP-Clientprozeduren verarbeiten nun Umleitungen mit relativen Pfaden. Zuvor waren Umleitungen nur erfolgreich, wenn der Server eine absolute URL bereitstellte.

Eine `GET`-Methode, die eine Umleitung empfängt, stellt nur die Abfragekomponente bereit, die von der URL des `Location-Header` der Umleitungsantwort angegeben wurde. Eine `POST`-Methode, die eine Umleitung empfängt, gibt eine Anforderungs-URL mit dem Pfad und den Abfragekomponenten aus, die vom `Location-Header` der Umleitungsantwort angegeben wurde. Der Hauptteil enthält die von der Prozedur generierte Abfragekomponente.

Abfrageparameter können nun sowohl in der URL-Klausel als auch in Parametern (automatisch generiert) angegeben werden, die einer Prozedur übergeben werden. Dies gilt nur für Prozeduren, die eine `GET` HTTP-Methode angeben.

- **Sichere Webdienste nun unter Windows Mobile unterstützt** Webdienst-Prozeduren, die `HTTPS` und `HTTPS_FIPS` verwenden, werden nun unter Windows Mobile unterstützt.
- **SQL Anywhere-JDBC-Treiber unterstützt nun `PreparedStatement.setClob()`** Der SQL Anywhere-JDBC-Treiber unterstützt nun `PreparedStatement.setClob()`.

In früheren Versionen stellte der SQL Anywhere-JDBC-Treiber Unterstützung für `PreparedStatement.setBlob`, `ResultSet.getBlob` und `ResultSet.getClob` bereit. Der SQL Anywhere-JDBC-Treiber unterstützt jetzt zwei der drei `PreparedStatement.setClob`-Methoden. Diese werden nachfolgend aufgelistet.

```
PreparedStatement.setClob( int parameterIndex, Clob x )  
PreparedStatement.setClob( int parameterIndex, Reader reader, long  
length )
```

Die folgende Variante wird nicht vom SQL Anywhere-JDBC-Treiber unterstützt.

```
PreparedStatement.setClob( int parameterIndex, Reader reader )
```

Bei der Verwendung der `PreparedStatement.setClob(int parameterIndex, Clob x)`-Überladung muss die benutzerdefinierte Clob-Implementierung die `Clob.length`- und `Clob.getCharacterStream`-Methode unterstützen. Auch bei großen Zeichenfolgen und umfangreichen Zeichendatenströmen sind die neuen `PreparedStatement.setClob`-Methoden innerhalb des SQL Anywhere-JDBC-Treibers den `PreparedStatement.setString`- und `PreparedStatement.setCharacterStream`-Methoden sowohl in Bezug auf Performance als auch auf Speichernutzung vorzuziehen. Siehe „JDBC-Unterstützung“ [[SQL Anywhere Server - Programmierung](#)].

- **Java VM ClassLoader von SQL Anywhere unterstützt nun Shutdown-Hooks** Der Java VM ClassLoader von SQL Anywhere, der zur Bereitstellung von Unterstützung für Java in der Datenbank dient, ermöglicht es Anwendungen, Shutdown-Hooks zu installieren. Siehe „Shutdown-Hooks in der Java VM“ [[SQL Anywhere Server - Programmierung](#)].
- **.NET SetupVSPackage-Installationsprogramm von SQL Anywhere** Die SetupVSPackage-Anwendung führt nun mehrere Installationsprogrammfunktionen aus, wie z.B. die Aktualisierung des Global Assembly Cache und der Microsoft Windows .NET-Datei *machine.config*. Wenn SQL Server 2008 oder höher auf dem System installiert ist, installiert SetupVSPackage auch zwei Zuordnungsdateien namens *MSSqlToSA.xml* und *SAToMSSql10.xml* im SQL Server-Ordner *DTS\MappingFiles*. Siehe „Deployment eines .NET Clients“ [[SQL Anywhere Server - Programmierung](#)].

Performanceverbesserungen

Die folgende Liste enthält die Performanceverbesserungen in Version 12.0.1, die nicht mit sichtbaren Änderungen der Benutzeroberfläche einhergehen.

- **Sofort-Textindizes** Die Performance der Sofort-Textindexwartung wurde verbessert. Die festgestellte Performanceverbesserung hängt von der jeweiligen Arbeitslast, dem Dokument und dem Textindexinhalt ab. Die Erstellung von Textindizes langer Dokumente profitiert am meisten von dieser Verbesserung.
- **Physische Löschungen werden bis zum Aufräumvorgang verschoben** Zu Verbesserung der Transaktions-Performance werden Zeilen zum Zeitpunkt der Festschreibung nicht physisch gelöscht. Stattdessen werden sie als logisch gelöscht markiert und die tatsächliche Löschung wird bis zum Aufräumvorgang verschoben. Um ein übermäßiges Anwachsen der Datenbank zu verhindern, wurde der Bereinigungszeitplan intensiver gestaltet. Siehe „sa_clean_database-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Für umgehbare Abfragen können jetzt berechnete Spalten für die Tabelle in der FROM-Klausel definiert werden** Eine der Bedingungen dafür, dass eine Abfrage zur Umgehung des Optimierers in Betracht gezogen wird, war, dass keine berechneten Spalten für die Basistabelle

definiert waren, die in der FROM-Klausel referenziert wurden. Diese Einschränkung wurde in SA 12.0.1 beseitigt. Der allgemeine Eliminierungsprozess für Unterausdrücke mithilfe von berechneten Spalten wird für umgangene Abfragen durchgeführt, ähnlich Abfragen mithilfe des SQL Anywhere-Optimierers.

- **Umgehen von Abfragen ist nun möglich, wenn die Basistabelle berechnete Spalten enthält** Früher konnte einer Abfrage nicht als Bypass-Abfrage bestimmt werden, wenn die in der FROM-Klausel referenzierte Basistabelle berechneten Spalten enthielt. Diese Einschränkung wurde aufgehoben. Siehe „[Eignung für ein Überspringen der Abfrageverarbeitungsphase](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
- **Erweiterungen für die Verwendung von Indizes durch den Optimierer** Die Indizierungsfunktionen von SQL Anywhere wurden in mehrfacher Hinsicht erweitert, einschließlich Verbesserungen zur Optimierung von Abfragen, die partielle Index-Scans und mehrfache Index-Scans mit demselben Index verwenden.
- **IS NOT NULL-Suchbedingung** Der Optimierer kann nun einen Index zur Prüfung der IS NOT NULL-Suchbedingung verwenden. Siehe „[Abfrageprädikate](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Eigenschaften und Systemmonitorstatistiken

In der folgenden Liste sind die Erweiterungen von Eigenschaften und Systemmonitorstatistiken in SQL Anywhere Version 12.0.1 aufgeführt.

- **Verbindungseigenschaften** Folgende Verbindungseigenschaften wurden in dieser Version hinzugefügt:
 - ConnectedTime
 - UserDefinedCounterRate01
 - UserDefinedCounterRate02
 - UserDefinedCounterRate03
 - UserDefinedCounterRate04
 - UserDefinedCounterRate05
 - UserDefinedCounterRaw01
 - UserDefinedCounterRaw02
 - UserDefinedCounterRaw03
 - UserDefinedCounterRaw04
 - UserDefinedCounterRaw05

Weitere Hinweise zu diesen Eigenschaften finden Sie unter „[Liste der Verbindungseigenschaften](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Datenbankeigenschaften** Folgende Datenbankeigenschaften wurden in dieser Version hinzugefügt:

- ApproximateCPUTime
- BytesReceived
- BytesReceivedUncomp
- BytesSent
- BytesSentUncomp
- CarverHeapPages
- ClientStmtCacheHits
- ClientStmtCacheMisses
- ConnectedTime
- HeapsCarver
- HeapsLocked
- HeapsQuery
- HeapsRelocatable
- PacketsReceived
- PacketsReceivedUncomp
- PacketsSent
- PacketsSentUncomp
- PrepStmt
- QueryHeapPages
- QueryMemActiveCurr
- QueryMemGrantFailed
- QueryMemGrantGranted
- QueryMemGrantRequested
- QueryMemGrantWaited
- QueryMemGrantWaiting
- ReqCountActive
- ReqCountBlockContention
- ReqCountBlockIO
- ReqCountBlockLock
- ReqCountUnscheduled
- ReqTimeActive
- ReqTimeBlockContention
- ReqTimeBlockIO
- ReqTimeBlockLock
- ReqTimeUnscheduled
- RequestsReceived
- Rlbk
- UserDefinedCounterRate01
- UserDefinedCounterRate02
- UserDefinedCounterRate03
- UserDefinedCounterRate04
- UserDefinedCounterRate05
- UserDefinedCounterRaw01
- UserDefinedCounterRaw02
- UserDefinedCounterRaw03

- UserDefinedCounterRaw04
- UserDefinedCounterRaw05

Weitere Hinweise zu diesen Eigenschaften finden Sie unter „[Liste der Datenbankseigenschaften](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Datenbankservereigenschaften** Folgende Datenbankservereigenschaften wurden in dieser Version hinzugefügt:

- ApproximateCPUTime
- Commit
- ConnectedTime
- Cursor
- CursorOpen
- PrepStmt
- Rlbk
- UserDefinedCounterRate01
- UserDefinedCounterRate02
- UserDefinedCounterRate03
- UserDefinedCounterRate04
- UserDefinedCounterRate05
- UserDefinedCounterRaw01
- UserDefinedCounterRaw02
- UserDefinedCounterRaw03
- UserDefinedCounterRaw04
- UserDefinedCounterRaw05

Weitere Hinweise zu diesen Eigenschaften finden Sie unter „[Liste der Datenbankservereigenschaften](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Systemmonitor-Statistiken** In dieser Version wurden die folgenden Systemmonitorstatistiken hinzugefügt:

- Benutzerdefinierte Rate: Counter1
- Benutzerdefinierte Rate: Counter2
- Benutzerdefinierte Rate: Counter3
- Benutzerdefinierte Rate: Counter4
- Benutzerdefinierte Rate: Counter5
- Benutzerdefinierte Rohdaten: Counter1
- Benutzerdefinierte Rohdaten: Counter2
- Benutzerdefinierte Rohdaten: Counter3
- Benutzerdefinierte Rohdaten: Counter4
- Benutzerdefinierte Rohdaten: Counter5

Verhaltensänderungen von SQL Anywhere

Nachstehend finden Sie eine Liste der Verhaltensänderungen von SQL Anywhere, die in Version 12.0.1 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Von der LOAD und UNLOAD TABLE-Anweisung verwendete Escapezeichen** In früheren Versionen durfte das Escapezeichen für diese Anweisungen nicht länger als 1 Byte sein. Nun wird empfohlen, dass die für das Escapezeichen angegebene Zeichenfolge nicht länger als 255 Byte ist. Es wird aber empfohlen, dass die Zeichenfolge nicht mehr als ein Zeichen umfasst.
- **Minimale und anfängliche Cachegröße kann sich abhängig von der maximalen Cachegröße erhöhen** Wenn Sie versuchen, Ihre anfängliche oder minimale Cachegröße auf einen Wert, der kleiner ist als ein Achtel der maximalen Cachegröße zu setzen, werden die anfängliche bzw. die minimale Cachegröße automatisch relativ zur maximalen Cachegröße erhöht. Daher sind die minimale und die anfängliche Cachegröße häufig größer als in früheren Versionen.
- **Änderungen des Bereichs der reserved_keywords-Option** Früher konnten Sie die reserved_keywords-Option für einzelne Benutzer festlegen oder angeben, dass die Einstellung temporär war. Wenn die temporären Einstellungen oder die Einstellungen auf Benutzerebene jedoch bei der Ausführung bestimmter DDL-Anweisungen von den entsprechenden PUBLIC-Einstellungen abwichen, gab es möglicherweise Probleme bei der Wiederherstellung oder dem Neuaufbau der Datenbank.

Die folgende Verhaltensänderung für die reserved_keywords-Option gilt für neue Datenbanken der Version 12.0.1:

- Temporäre und Nicht-PUBLIC-Einstellungen sind *nicht* zulässig.

Die folgenden Verhaltensänderungen der reserved_keywords-Option gelten für vorhandene Datenbanken der Version 12:

- Vorhandene Nicht-PUBLIC-Einstellungen werden während der Ausführung ignoriert.
- Das Entfernen von vorhandenen Nicht-PUBLIC-Einstellungen ist zulässig.
- Dbunload ignoriert vorhandene Nicht-PUBLIC-Einstellungen.

- **Optionen der Multiprogramming-Stufe** Früher wurden Datenbankserver-Optionen zur Multiprogramming-Stufe, die sich ausschließlich auf den Netzwerk-Datenbankserver (dbsrv12) bezogen, vom Personal Datenbankserver (dbeng12) ohne Generierung einer Fehlermeldung ignoriert. Nun generieren die folgenden Datenbankserveroptionen eine Warnung, wenn sie auf dem Personal Datenbankserver verwendet werden:
 - dbsrv12-Serveroption **-gna**
 - dbsrv12-Serveroption **-gnh**
 - dbsrv12-Serveroption **-gnl**
 - dbsrv12-Serveroption **-gns**

- **Änderung der Datenbankserveroption -zt** Die Datenbankserveroption -zt macht die Eigenschaften ReqCountBlockIO und ReqTimeBlockIO nicht mehr verfügbar. Diese beiden Eigenschaften sind nun immer verfügbar, unabhängig davon, ob die RequestTiming-Verbindungseigenschaft aktiviert ist.
- **Änderung des ODBC-Verbindungsparameters für die SQLTables-Funktion** Wenn Sie die ODBC-Funktion SQLTables ausführen, gibt die TABLE_TYPE-Spalte nun für materialisierte Ansichten standardmäßig den Wert VIEW zurück. In früheren Versionen gab die Funktion den Wert MATERIALIZED VIEW zurück. Sie können dieses Standardverhalten mit dem MATVIEW-Verbindungsparameter ändern. Siehe „[Verbindungsparameter MatView \(MATVIEW\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **CONVERT-Funktion** Die syntaktische Analyse (Parsing) von formatierten Zeit-Zeichenfolgen wurde so erweitert, dass der Zeitteil einer Zeichenfolge akzeptiert wird, vorausgesetzt, er hat das Format hh:nn:ss.ssssssAA. Die Zeit-Zeichenfolge muss die Stundenzahl angeben. Alle anderen Teile der Uhrzeit sind optional. Der AM/PM-Indikator wird immer akzeptiert, unabhängig davon, ob Teile der Uhrzeit weggelassen werden. Dies erlaubt bis zu sechs Stellen nach der Sekundenangabe, um Mikrosekunden anzugeben.

Diese Änderung betrifft die Konvertierung von Zeichenfolgen in TIME und TIMESTAMP. Siehe „[CONVERT-Funktion \[Datentypkonvertierung\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SQL Anywhere 10 und spätere Versionen konnten Konvertierungen einer Zeichenfolge in einen Zeitwert mithilfe der CONVERT-Funktion und einem angegebenen *format-style* zurückweisen, die in früheren Versionen waren. Beispiel: Die folgende Anweisung wird in Version 9 akzeptiert, jedoch in Version 10 und höher zurückgewiesen:

```
SELECT CONVERT( TIME, '11:45am', 14 ) tm_conv
```

Das Verhalten bei der Konvertierung von Zeichenfolgen in TIME wurde von Version 9 von SQL Anywhere zu Version 10 und höher geändert. Version 10 und später wenden dieselben Regeln an, die bei Konvertierungen von Zeichenfolgen in TIMESTAMP gelten. Beispiel: Die Zeichenfolge 11:45am stimmt nicht exakt mit dem Formatstil 14 (hh:nn:ss.sss) überein, da sie einen amIndikator enthält, der nicht im Stil enthalten ist.

Die folgende Anweisung wird in Version 9 akzeptiert, jedoch in Version 10 und höher zurückgewiesen, weil die Zeichenfolge nicht mit dem Formatstil 101 (mm/tt/jjjj) übereinstimmt:

```
SELECT CONVERT( TIME, '1991-02-03 11:45', 101 )
```

- **Änderung des lizenzierten Benutzers und Firmennamens mit dem Serverlizen-Dienstprogramm (dblic)** Sie können nun den lizenzierten Benutzer und die lizenzierte Firma ändern, ohne die Benutzerzahl und den Lizenztyp anzugeben. Siehe „[Dienstprogramm für die Serverlizenzierung \(dblic\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Nicht mehr empfohlene SQL Anywhere-Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

Im Folgenden finden Sie eine Liste nicht mehr empfohlener Funktionen in Version 12.0.1:

- **Option -kr** Bislang mussten Kerberos-Prinzipale das Format *server-name@default-realm* haben.

Der Kerberos-Serverprinzipal, einschließlich des Realm, kann mit der Serveroption -kp angegeben werden. Der mit -kp angegebene Serverprinzipal muss in die Kerberos-Keytab-Datei auf dem Computer extrahiert worden sein, auf dem der Datenbankserver läuft. Die Option -kr kann nicht angegeben werden, wenn die Option -kp angegeben ist. Siehe „[Datenbankserveroption -kp](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **SQL Anywhere JDBC 3.0-Treiber** Der SQL Anywhere JDBC 3.0-Treiber wird in dieser Softwareversion nicht mehr empfohlen. Es wird empfohlen, in Anwendungen anstelle von *sajdbc.jar* nun *sajdbc4.jar* zu verwenden. Siehe „[JDBC-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].

Neue MobiLink-Funktionen

Nachstehend finden Sie eine Liste der neuen Funktionen von MobiLink, die in Version 12.0.1 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Konsolidierte Datenbanken

Die folgende Liste enthält die Erweiterungen zur Unterstützung konsolidierter Datenbanken für MobiLink in SQL Anywhere Version 12.0.1.

- **SAP Sybase IQ wird nun unterstützt** Der MobiLink-Server unterstützt nun SAP Sybase IQ als konsolidierte Datenbank. Siehe „[Konsolidierte SAP Sybase IQ-Datenbank](#)“ [[MobiLink - Serveradministration](#)]. Weitere Hinweise zu empfohlenen Treibern finden Sie unter <http://www.sybase.com/detail?id=1011880>.
- **Unterstützung für ASE 15.5 hinzugefügt** Der MobiLink-Server unter Windows und Linux unterstützt nun konsolidierte Datenbanken, die auf einem ASE 15.5 Server laufen. Weitere Hinweise zu empfohlenen Treibern finden Sie unter <http://www.sybase.com/detail?id=1011880>.

MobiLink-Server

Die folgende Liste enthält die in SQL Anywhere Version 12.0.1 eingeführten Erweiterungen des MobiLink-Servers.

- **Automatische Anpassung der Datenbank-Worker-Threads** Der MobiLink-Server kann nun die Anzahl der Datenbank-Worker-Threads zur Maximierung des Durchsatzes automatisch anpassen. Siehe „[mlsrv16-Option -wm](#)“ [[MobiLink - Serveradministration](#)].
 - **Neue benannte Systemparameter** Die folgenden neuen benannten Systemparameter wurden MobiLink hinzugefügt, damit Skripten eine neue entfernte ID oder einen neuen entfernten Benutzernamen identifizieren können:
 - **new_remote_id** Dieser neue Systemparameter gibt eine neue entfernte ID an und ist in den Verbindungsskripten `authenticate_user` und `begin_synchronization` sowie im `begin_synchronization`-Tabellenskript verfügbar.
 - **new_username** Dieser neue Systemparameter gibt einen neuen Benutzernamen an und ist in den Verbindungsskripten `authenticate_user` und `begin_synchronization` sowie im `begin_synchronization`-Tabellenskript verfügbar.
- Weitere Hinweise finden Sie unter:
- „[authenticate_user](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)]
 - „[begin_synchronization](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)]
 - „[begin_synchronization](#) (Tabellenereignis)“ [[MobiLink - Serveradministration](#)]
- **Datentypen bigint und bigdatetime zur Unterstützung von ASE 15.5 hinzugefügt** Diese Datentypen sollten in SQL Anywhere und UltraLite TIME bzw. TIMESTAMP zugeordnet werden. Siehe „[Zuordnung der Datentypen von Adaptive Server Enterprise](#)“ [[MobiLink - Serveradministration](#)].
 - **Maximale Länge für Authentifizierungsparameter wurde erweitert** Jeder MobiLink-Authentifizierungsparameter kann eine Länge von bis zu 4000 Byte haben. Die frühere Obergrenze war 128 Byte. Siehe „[Authentifizierungsparameter](#)“ [[MobiLink - Serveradministration](#)].

Neue mlsrv12-Funktionen

- **Neue Option -wm zum Aktivieren der automatischen Anpassung von Datenbank-Worker-Threads** Die mlsrv12-Option -wm wurde hinzugefügt. Verwenden Sie die Option -wm, um die maximale Anzahl von Datenbank-Worker-Threads festzulegen, sodass der MobiLink-Server automatisch die optimale Anzahl von Datenbank-Worker-Threads wählen kann. Siehe „[mlsrv16-Option -wm](#)“ [[MobiLink - Serveradministration](#)].
- **Die Optionen -cmax, -cmin und -cinit unterstützen nun einen Prozentwert für den Größenparameter** Der Größenparameter für die Serveroptionen -cmax, -cmin und -cinit kann nun als Prozentwert angegeben werden. Siehe „[mlsrv16-Option -cmax](#)“ [[MobiLink - Serveradministration](#)], „[mlsrv16-Option -cmin](#)“ [[MobiLink - Serveradministration](#)] und „[mlsrv16-Option -cinit](#)“ [[MobiLink - Serveradministration](#)].
- **Neuer log_bad_request-Parameter für mlsrv12-Option -x** Die mlsrv12-Option -x unterstützt nun die `log_bad_request`-Option für HTTP, HTTPS und OE. Bei der Einstellung "yes" gibt der MobiLink-Server einen Fehler aus, wenn er eine unvollständige oder unerwartete HTTP-Anforderung empfängt. Siehe „[mlsrv16-Option -x](#)“ [[MobiLink - Serveradministration](#)].

- **DER-kodierte Schlüssel werden nun für die Ende-zu-Ende-Verschlüsselung unterstützt** Die `e2ee_private_key`-Option für die `mlsrv12-Serveroption -x` unterstützt nun sowohl PEM- als auch DER-kodierte Schlüssel.

Neue zentrale MobiLink-Administration von entfernten Datenbankfunktionen

Im Folgenden finden Sie eine Liste der Erweiterungen der zentralen MobiLink-Administrationsfunktion für entfernte Datenbanken, die in SQL Anywhere Version 12.0.1 eingeführt wurde.

- **Neue Variablen für Bedingungen für Aufgaben** Der MobiLink-Agent für die zentrale Administration von entfernten Datenbanken unterstützt nun zwei neue Variablen für Bedingungen von Aufgaben: `{is_online}` und `{network_conn_name}`. Die Variable `{is_online}` kann verwendet werden, um zu überprüfen, ob das Clientgerät mit einem Netzwerk verbunden ist, auf dem der MobiLink-Server des Agenten zu erreichen ist. Die Variable `{network_conn_name}` gibt den Namen der Netzwerkverbindung an, die vom Agenten für die Kommunikation mit dem MobiLink-Server verwendet wird. Siehe „[Variablen in Parametern](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Unterstützung für zufällige Verzögerung der Ausführung einer Aufgabe** Eine entfernte Aufgabe kann jetzt ein zufälliges Verzögerungsintervall (in Sekunden) haben. Jeder Agent verwendet das Intervall zum Generieren einer zufälligen Anzahl von Sekunden, um die die Ausführung jeder Aufgabe verzögert wird. Siehe „[Entfernte Aufgaben](#)“ [[MobiLink - Serveradministration](#)].
- **Der MobiLink-Agent für die zentrale Administration von entfernten Datenbanken kann nun interaktiv konfiguriert werden** Der MobiLink-Agent kann nun interaktiv auf einem entfernten Gerät über eine Reihe von Konfigurationsfenstern konfiguriert werden.

Serverdienstprogramme

Die folgende Liste enthält die in SQL Anywhere Version 12.0.1 eingeführten Erweiterungen der MobiLink-Serverdienstprogramme.

MobiLink Replay-Dienstprogramm (mlreplay)

- **Das mlreplay-Dienstprogramm kann nun mithilfe der Befehlszeile mehrere simulierte Clients ausführen, um ein aufgezeichnetes Protokoll wiederzugeben** Sie steuern die Anzahl der simulierten Clients mit der `mlreplay`-Option `-n`. Siehe „[Dienstprogramm MobiLink Replay \(mlreplay\)](#)“ [[MobiLink - Serveradministration](#)].
- **Neue mlreplay-Option -rep hinzugefügt** Die `mlreplay`-Option `-rep` legt fest, wie oft simulierte Clients das aufgezeichnete Protokoll wiedergeben sollen. Siehe „[Dienstprogramm MobiLink Replay \(mlreplay\)](#)“ [[MobiLink - Serveradministration](#)].
- **Neue mlreplay-Option -rnt hinzugefügt** Die `mlreplay`-Option `-rnt` weist simulierte Clients an, neue Wiederholungen der Protokollwiedergaben zu starten, bis die angegebene Laufzeit erreicht ist. Siehe „[Dienstprogramm MobiLink Replay \(mlreplay\)](#)“ [[MobiLink - Serveradministration](#)].
- **Neue mlreplay- und mlgenreplayapi-Option @data zur Unterstützung der Konfigurationsdateien hinzugefügt** Mit der `mlreplay`-Option `@data` können Sie `mlreplay` und `mlgenreplayapi` mit Befehlszeilenspezifikationen ausführen, die in einer Konfigurationsdatei

gespeichert sind. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)], „MobiLink-generiertes Replay-API-Dienstprogramm (mlgenreplayapi)“ [[MobiLink - Serveradministration](#)] und „Konfigurationsdateien“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Neue mlreplay-Option -ls hinzugefügt** Die mlreplay-Option -ls protokolliert Statistiken für jeden simulierten Client. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)].
- **Neue mlreplay- und mlgenreplayapi-Option -os hinzugefügt** Die Option -os begrenzt die maximale Größe der Logdatei. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)] und „MobiLink-generiertes Replay-API-Dienstprogramm (mlgenreplayapi)“ [[MobiLink - Serveradministration](#)].
- **In der Befehlszeile können nun mehrere simulierte Clients ausgeführt werden** Sie können nun in der Befehlszeile in den Parametern -u, -p und -r das Sternchensymbol verwenden, um mehrere Benutzernamen, Kennwörter und entfernte IDs anzugeben. Das Sternchen wird bei jedem Aufruf eines simulierten Clients durch die Nummer des simulierten Clients ersetzt.

Mit der neuen Option -rp mlreplay können Sie das Sternchen als Platzhaltersymbol durch ein anderes Symbol ersetzen. Sie können dann das neue Symbol verwenden, um mehrere Benutzernamen, Kennwörter und entfernte IDs anzugeben. Siehe „Dienstprogramm MobiLink Replay (mlreplay)“ [[MobiLink - Serveradministration](#)].

MobiLink-Plug-In für Sybase Central

Die folgende Liste enthält die in SQL Anywhere Version 12.0.1 eingeführten Erweiterungen des MobiLink-Plug-Ins für Sybase Central.

- **Synchronisationsmodell-Deployment erstellt nun Synchronisationsprofile** Beim Deployment eines Synchronisationsmodells in einer entfernten Datenbank wird nun automatisch ein Synchronisationsprofil erstellt.
- **Unterstützung des Synchronisationsmodells für SAP Sybase IQ** Synchronisationsmodelle können nun mit SAP Sybase IQ-Datenbanken als konsolidierten Datenbanken erstellt werden. Siehe „[Konsolidierte SAP Sybase IQ-Datenbank](#)“ [[MobiLink - Serveradministration](#)].
- **Zeitstempelbasierte Downloads mithilfe von Spaltenstandardwerten anstelle von Triggern** In Synchronisationsmodellen, die konsolidierte SQL Anywhere-, SAP Sybase IQ-, IBM DB2 LUW- und MySQL-Datenbanken verwenden, können jetzt zeitstempelbasierte Downloads verwendet werden, ohne dass Trigger zum Verwalten von TIMESTAMP-Spalten erforderlich sind. Anstelle von Triggern können Spaltenstandardwerte verwendet werden, um automatisch die TIMESTAMP-Spalte in einer Zeile zu aktualisieren, wenn die Zeile eingefügt oder aktualisiert wird. Zur Vereinfachung wurde die neue Option **Spaltenstandardwert anstelle von Trigger verwenden** auf der Seite **Zeitstempel-Downloadoptionen** des **Assistenten zum Erstellen eines Synchronisationsmodells** und auf der Registerkarte **Downloadtyp** des Tabellenzuordnungseditors hinzugefügt.

- **Namen für entfernte Schemata können aus einer konsolidierten Datenbank importiert werden** Wenn Sie einem Projekt entweder mit dem **Assistenten zum Erstellen eines Projekts** oder dem **Assistenten zum Hinzufügen einer konsolidierten Datenbank** eine konsolidierte Datenbank hinzufügen, prüft der Assistent automatisch, ob Namen für entfernte Schemata in der konsolidierten Datenbank definiert sind, die noch nicht im Projekt enthalten sind. Wenn dies der Fall ist, werden Sie gefragt, ob Sie sie importieren wollen.
- **Gruppen können aus einer Datei erstellt werden** Sie können mit dem **Assistenten zum Erstellen von Gruppen** die Agenten für eine Gruppe auswählen, indem Sie auf eine Textdatei zeigen, die den Namen der Agenten enthält. Sie können Mitglieder auch hinzufügen, indem Sie die in einer konsolidierten Datenbank definierten Agenten durchsuchen.
- **Neue Hilfeschaftfläche für Befehle für entfernte Aufgaben** Es gibt nun rechts von der Dropdown-Liste **Befehlstyp** ein Fragezeichensymbol, das Sie verwenden können, um Hilfe zu den Befehlen zur Erstellung von entfernten Aufgaben anzuzeigen.
- **Neues Feld Beschreibung für Agenten** Die Fenster **Eigenschaften des Agenten** und **Assistent zum Erstellen von MobiLink-Agenten** haben jetzt ein Feld mit dem Namen **Beschreibung**, in das Sie eine Beschreibung des Agenten eingeben können.
- **Neue CustDB-Beispielprojektdatei hinzugefügt** Eine CustDB-Beispielprojektdatei, `%SQLANY12%\MobiLink\CustDB\project.mlp`, ist nun verfügbar, damit Sie einfach mit CustDB-Projekten und Skripten zur Anzeige von Datenbanken arbeiten können. Weitere Hinweise finden Sie unter „MobiLink-Beispiel CustDB“ [[MobiLink - Erste Orientierung](#)] und „Praktische Einführung: Erstellen der Beispielanwendung CustDB“ [[UltraLite - Datenbankverwaltung](#)].

MobiLink-Clients

Die folgende Liste enthält die in SQL Anywhere Version 12.0.1 eingeführten Erweiterungen für MobiLink-Clients.

- **Automatische Wiederaufnahme von HTTP-Sitzungen** Wenn eine HTTP-Verbindung getrennt wird, versuchen MobiLink-Clients automatisch, die Verbindung mit dem MobiLink-Server wiederherzustellen und die Synchronisation fortzusetzen. Die Synchronisation schlägt nur fehl, wenn der Client die Verbindung mit dem MobiLink-Server mehrfach nicht wiederherstellen kann.
- **DER-kodierte Schlüssel werden nun für die Ende-zu-Ende-Verschlüsselung unterstützt** Die Client-Protokolloption `e2ee_public_key` unterstützt nun sowohl PEM- als auch DER-kodierte Schlüssel.
- **ECC-Kurvenunterstützung wurde zur Unterstützung von 15 Kurven erweitert** Die unterstützten ECC-Kurven sind: `sect163k1`, `sect163r2`, `sect233k1`, `sect233r1`, `sect283k1`, `sect283r1`, `sect409k1`, `sect409r1`, `sect571k1`, `sect571r1`, `secp192r1`, `secp224r1`, `secp256r1`, `secp384r1` und `secp521r1`.
- **Unterstützung öffentlicher Schlüssel aus x.509-Zertifikaten** Die Ende-zu-Ende-Verschlüsselung unterstützt nun öffentliche Schlüssel aus x.509-Zertifikaten. Siehe „`e2ee_public_key`“ [[MobiLink - Clientadministration](#)].

Relay Server

Folgende Relay Server-Funktionen wurden in dieser Version hinzugefügt:

- **Der Relay Server Outbound Enabler (RSOE) unterstützt nun HTTP-Authentifizierung (Basic und Digest) bei Webservern und HTTP-Proxy-Servern** Die folgenden neuen (optionalen) Netzwerkverbindungsoptionen wurden dem Outbound Enabler hinzugefügt:
 - **http_userid** Benutzer-ID für die Authentifizierung.
 - **http_password** Kennwort für die Authentifizierung.
 - **http_proxy_userid** Benutzer-ID für die Proxy-Authentifizierung.
 - **http_proxy_password** Kennwort für die Proxy-Authentifizierung.
 - **proxy_host** Gibt den Hostnamen oder die IP-Adresse des Proxy-Servers an.
 - **proxy_port** Legt die Portnummer des Proxy-Servers fest.

Weitere Hinweise finden Sie unter:

- „Outbound Enabler“ [[Relay Server](#)]
- „Netzwerkprotokolloptionen des MobiLink-Clients“ [[MobiLink - Clientadministration](#)]

Verhaltensänderungen von MobiLink

Nachstehend finden Sie eine Liste der Verhaltensänderungen von MobiLink, die in Version 12.0.1 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Änderungen des MobiLink-Servers

- **32-Bit-MobiLink-Server wird auf 64-Bit-Betriebssystemen nicht mehr unterstützt** Auf 64-Bit-Betriebssystemen wird der MobiLink-Server nur als 64-Bit-Anwendung installiert und muss als 64-Bit-Anwendung ausgeführt werden. 32-Bit-Versionen des MobiLink-Servers, die möglicherweise auf Ihrem System aus früheren Installationen der Version 12 verblieben sind, können auf einem 64-Bit-System nicht ausgeführt werden.

Hinweis

Für MobiLink-Server-Versionen vor 12.0.1 wird *kein* Fehler ausgegeben, wenn der MobiLink-Server auf einem 64-Bit-Betriebssystem als 32-Bit-Anwendung ausgeführt wird.

Ab Version 12.0.1, wird der Fehler -10381 zurückgegeben, wenn der MobiLink-Server als 32-Bit-Anwendung auf einem 64-Bit-Betriebssystem läuft. Siehe „[Der MobiLink-Server muss als 64-Bit-Anwendung auf einem 64-Bit-Betriebssystem ausgeführt werden](#)“ [[Fehlermeldungen](#)].

- **Neues Verhalten für mlsrv12-Option -w** Wenn Sie die Option -wm verwenden, um die Anzahl der Datenbank-Worker-Threads automatisch anzupassen, legt nun die Option -w für mlsrv12 die anfängliche Anzahl von Datenbank-Worker-Threads fest. Früher legte diese Option die Anzahl von Datenbank-Worker-Threads fest. Siehe „[mlsrv16-Option -w](#)“ [[MobiLink - Serveradministration](#)] und „[mlsrv16-Option -wm](#)“ [[MobiLink - Serveradministration](#)].
- **Die Option -cm ist nun ein Alias für die Option -cmax** Die Option -cm ist ein Alias für die Serveroption -cmax. Siehe „[mlsrv16-Option -cmax](#)“ [[MobiLink - Serveradministration](#)].

Änderungen der MobiLink-Server-Dienstprogramme

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink-Server-Dienstprogrammen in Version 12.0.1.

MobiLink Replay-Dienstprogramm (mlreplay)

- **Neues Verhalten für mlreplay- und mlgenreplayapi-Optionen -o und -ot** Die mlreplay- und mlgenreplayapi-Optionen -o und -ot protokollieren nun Befehlszeilenoptionen. Siehe „[Dienstprogramm MobiLink Replay \(mlreplay\)](#)“ [[MobiLink - Serveradministration](#)] und „[MobiLink-generiertes Replay-API-Dienstprogramm \(mlgenreplayapi\)](#)“ [[MobiLink - Serveradministration](#)].
- **Die mlreplay-Optionen -n und -sci können nun unter bestimmten Bedingungen gemeinsam verwendet werden** Die mlreplay-Optionen -n und -sci können gemeinsam verwendet werden, wenn die durch die Option -n angegebene Anzahl der simulierten Clients kleiner oder gleich der Anzahl der simulierten Clients in der Informationsdatei für simulierte Clients ist. Wenn die Optionen gemeinsam verwendet werden, gibt -n die Anzahl der ausgeführten simulierten Clients an.

Diese Optionen ermöglichen es, mithilfe einer Informationsdatei, die x als Anzahl der simulierten Clients angibt, ein Protokoll mit 1 bis x simulierten Clients wiederzugeben. Das frühere Verhalten erlaubte nur die Wiedergabe des aufgezeichneten Protokolls mit einer Anzahl von genau x simulierten Clients. Siehe „[Dienstprogramm MobiLink Replay \(mlreplay\)](#)“ [[MobiLink - Serveradministration](#)].

- **Codevorlage ist für Callback-Funktionen IdentifySimulatedClient und FiniIdentifySimulatedClient verfügbar** Wenn sie generiert werden, enthalten diese Callbacks auskommentierten Code, den Sie bei der Verwendung von generischen Benutzernamen, Kennwörtern und entfernten IDs für eine Wiedergabesitzung implementieren können. Siehe „[Callback-Funktion "IdentifySimulatedClient"](#)“ [[MobiLink - Serveradministration](#)] und „[Callback-Funktion "FiniIdentifySimulatedClient"](#)“ [[MobiLink - Serveradministration](#)].

Änderungen des MobiLink-Clients

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink-Clients in Version 12.0.1.

- **Die folgenden Änderungen wurden am Fenster DBMLSync-Setup für SQL Anywhere vorgenommen**
 - Die Optionen **Neuversuch gemäß entferntem Bearbeitungsfortschritt**, **Entfernter liegt zurück**, **Entfernter liegt voran**, **Verbindungen mit Konflikten trennen**, **Standortskript** und **Befehlszeilen-Hilfe** wurden entfernt.

- Die Option **Publikation** wurde durch die Option **Subskription** ersetzt.
- Wenn Sie versuchen, Ihr MobiLink-Kennwort zu ändern, prüft die Software, ob Sie ein nicht-leeres Kennwort eingegeben haben und ob das neue Kennwort mit dem Überprüfungskennwort übereinstimmt, bevor Sie das Fenster schließen können.
- Wenn das Fenster nach einem Synchronisationsversuch geöffnet wird, weil ein falsches MobiLink-Kennwort eingegeben wurde, wird in einer Meldung angezeigt, welches Problem vorliegt. Früher wurde die Fehlermeldung im dbmlsync-Fenster angezeigt.
- **Neuer Standardwert für beständige HTTP- und HTTPS-Synchronisationen** Standardmäßig verwenden MobiLink-Clients (mit Ausnahme von UltraLiteJ) bei der Synchronisation über HTTP oder HTTPS nun beständige Verbindungen. Zur Rückkehr zum früheren Verhalten muss in den Netzwerkprotokolloptionen des Clients die Angabe `persistent=no` gemacht werden.

Änderungen des MobiLink-Plug-Ins für Sybase Central

Im Folgenden finden Sie eine Liste der Verhaltensänderungen für das MobiLink-Plug-In für Sybase Central in Version 12.0.1.

- **Die Seite MobiLink-Benutzer und Subskription des Assistenten für das Deployment eines Synchronisationsmodells wurde geändert** Die Seite **MobiLink-Benutzer und Subskription** des Assistenten für das Deployment eines Synchronisationsmodells wurde in **MobiLink-Benutzer- und Synchronisationsprofil** umbenannt. Die Seite verfügt nun über eine Option zur Angabe des Namens des Synchronisationsprofils. Das Synchronisationsprofil wird nun erstellt, wenn ein Synchronisationsmodell per Deployment bereitgestellt wird.
- **Synchronisationsmodell-Deployment-Option zur Wahl von geeigneten Standardwerten für entfernte Aufgaben** Die erste Seite des **Assistenten für das Deployment eines Synchronisationsmodells** verfügt nun beim Deployment auf einer entfernten Datenbank über eine Option zum Initialisieren des Assistenten mit geeigneten Einstellungen für die Erstellung von entfernten Aufgaben bei der zentralen Administration von entfernten Datenbanken. Diese Option ist standardmäßig ausgewählt.

Nicht mehr empfohlene MobiLink-Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

- **IBM DB2 Mainframe nicht mehr unterstützt** IBM DB2 Mainframe wird nicht mehr als konsolidierte Datenbank unterstützt. MobiLink unterstützt jedoch weiterhin DB2 LUW (Linux, Unix und Windows) als konsolidierte Datenbank.

Neue Funktionen in SQL Remote

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in SQL Remote in Version 12.0.1 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Neue Steuerungsparameter für SET REMOTE OPTION-Anweisung** Wenn SQL Remote im kontinuierlichen Modus ausgeführt wird und beim Zugriff auf das Nachrichtensystem ein Fehler auftritt, können Sie nun Folgendes steuern:
 - Wie oft SQL Remote den Sende- bzw. Empfangsvorgang wiederholen soll, bevor das Programm beendet wird. Weitere Informationen finden Sie unter der allgemeinen max_retries-Option in „SET REMOTE OPTION-Anweisung [SQL Remote]“ [SQL Anywhere Server - SQL-Referenzhandbuch].
 - Wie lange zwischen Neuversuchen gewartet werden soll. Weitere Informationen finden Sie unter der allgemeinen pause_after_failure-Option in „SET REMOTE OPTION-Anweisung [SQL Remote]“ [SQL Anywhere Server - SQL-Referenzhandbuch].

Verhaltensänderungen von SQL Remote

Im Folgenden finden Sie eine Liste der Änderungen, die in SQL Remote in Version 12.0.1 vorgenommen wurden.

- **Verbesserte Performance bei der Verarbeitung fehlender Nachrichten** Wenn in Versionen vor SQL Remote Version 16.0 die konsolidierte Datenbank beendet wurde, bevor alle Nachrichten in einer mehrteiligen Nachricht empfangen wurden, löschte die entfernte Datenbank alle empfangenen Nachrichten und sendete eine Neusendeanforderung an die konsolidierte Datenbank. Die konsolidierte Datenbank durchsuchte dann das anwendbare Transaktionslog, um der Neusendeanforderung zu entsprechen. Bei SQL Remote Version 16.0 generiert SQL Remote beim Start eine Warnmeldung zu den fehlenden Nachrichten, löscht die vorhandenen Nachrichten und sendet nur dann eine Neusendeanforderung, wenn es eine Nachricht mit Festschreibung erkannt hat, die darüber hinaus der letzte Teil einer mehrteiligen Nachricht oder die nächste Nachricht nach der vorherigen mehrteiligen Nachricht ist.

Neue Funktionen von UltraLite

Nachstehend finden Sie eine Liste der neuen Funktionen von UltraLite, die in Version 12.0.1 hinzugefügt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Allgemeine Funktionen

- **UltraLite wird nun auf Android-Smartphones unterstützt** UltraLite unterstützt nun Android-Smartphones. Sie können mit der Android-Implementierung der UltraLiteJ-API Anwendungen erstellen, die UltraLite-Datenbanken verwenden. Die Implementierungssyntax unterscheidet sich von der BlackBerry-Implementierung, die nur UltraLite-Datenbanken unterstützt, die speziell für

BlackBerry-Smartphones eingesetzt werden. Weitere Hinweise finden Sie unter „Vorteile der UltraLite APIs für Windows Mobile“ [[UltraLite - Datenbankverwaltung](#)] und „UltraLiteJ-Anwendungsentwicklung“ [[UltraLite® – Java-Programmierung](#)].

In einer neuen praktischen Einführung wird die Erstellung einer UltraLite-Anwendung auf einem Android-Smartphone oder eines Simulators in der Eclipse-Umgebung gezeigt. Diese praktische Einführung basiert auf einem neuen Beispielcode, der sich im Verzeichnis `%SQLANY12%\UltraLiteJ\Android\CustDB` befindet. Siehe „Praktische Einführung: Eine Android-Anwendung erstellen“ [[UltraLite® – Java-Programmierung](#)].

- **Praktische Einführung zur Erstellung einer BlackBerry-Anwendung unterstützt nun die Eclipse-Umgebung** Die praktische Einführung für BlackBerry wurde aktualisiert und verwendet nun die Eclipse-Umgebung. Siehe „Praktische Einführung: Eine BlackBerry-Anwendung erstellen“ [[UltraLite® – Java-Programmierung](#)].
- **Dynamische Cachedimensionierung** UltraLite lässt nun seinen Datenbankdatei-Cache anwachsen, wenn dies als Reaktion auf Datenbankvorgänge empfehlenswert erscheint und der verfügbare Speicher dies zulässt. Anwendungen können die Cachegröße auch explizit ändern (normalerweise in Fällen, bei denen die Anwendung aufgefordert wird, die Speicherbelegung zu verringern). Siehe:
 - „UltraLite-Verbindungsparameter `CACHE_SIZE`“ [[UltraLite - Datenbankverwaltung](#)]
 - „UltraLite-Verbindungsparameter `CACHE_MIN_SIZE`“ [[UltraLite - Datenbankverwaltung](#)]
 - „UltraLite-Verbindungsparameter `CACHE_MAX_SIZE`“ [[UltraLite - Datenbankverwaltung](#)]
 - „UltraLite-Option `cache_allocation`“ [[UltraLite - Datenbankverwaltung](#)]
- **Unterstützung für reine Download-Tabellen in UltraLite**

_download_only-Suffix Änderungen der Tabelle in der konsolidierten Datenbank werden für UltraLite (nicht für Datenbanken der UltraLite Java Edition) während der Synchronisation heruntergeladen, lokale Änderungen werden jedoch nicht an MobiLink gesendet. Alle Tabellen mit dem Suffix **_download_only** sind als reine Download-Tabellen markiert. Tabellen können auch als reine Download-Tabellen eingerichtet werden, indem für die Integritätsregel-Klausel der Synchronisation der SQL-Anweisungen **CREATE TABLE** und **ALTER TABLE** **SYNCHRONIZE DOWNLOAD** angegeben wird. Siehe „UltraLite-Tabellen für reinen Download“ [[UltraLite - Datenbankverwaltung](#)], „CREATE TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)] und „ALTER TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)].

Neuer Parameter `TABLE_IS_DOWNLOAD_ONLY` Nicht festgeschriebene clientseitige Änderungen können zu Download-Konflikten bei der Synchronisation von Tabellen führen. Für UltraLite und die UltraLite Java Edition kann die `table_flags`-Spalte in der `systable`-Systemtabelle nun den Parameter `TABLE_IS_DOWNLOAD_ONLY` enthalten.

sync=download-Attribut für Tabellen Das XML-Format für entladene UltraLite-Datenbanken umfasst nun das Attribut `sync="download"` für Tabellen. Siehe „systable-Systemtabelle“ [[UltraLite - Datenbankverwaltung](#)].
- **TLS-Identitäten können in der Datenbank gespeichert werden** TLS-Identitäten, bestehend aus einem X.509-Zertifikat, einem privaten Schlüssel und (optional) einer Kette von Zertifikaten der

Zertifizierungsstelle, die das Clientzertifikat signiert hat, können für UltraLite-Datenbanken (nicht für Datenbanken der UltraLite Java Edition) nun zum Erstellungszeitpunkt in der UltraLite-Datenbank gespeichert werden. Den Dienstprogrammen ulinit und ulload wurden neue Optionen hinzugefügt. Siehe „[UltraLite-Dienstprogramm zum Initialisieren einer Datenbank \(ulinit\)](#)“ [[UltraLite - Datenbankverwaltung](#)], „[UltraLite-Dienstprogramm zum Laden von Daten aus XML-Dateien \(ulload\)](#)“ [[UltraLite - Datenbankverwaltung](#)].

- **Speicherorte für UltraLiteJ-Beispielcode sind nun nach Gerät und Plattformunterstützung organisiert** Codebeispiele befinden sich nun in den folgenden Verzeichnissen:

- BlackBerry-Beispiele: %SQLANY12%\UltraLiteJ\BlackBerry
- Java J2ME-Beispiele: %SQLANY12%\UltraLiteJ\J2ME
- Java J2SE-Beispiele: %SQLANY12%\UltraLiteJ\J2SE
- Android-Beispiele: %SQLANY12%\UltraLiteJ\Android

Siehe „[Codebeispiele](#)“ [[UltraLite® – Java-Programmierung](#)].

Plattformen und Geräte

- **Neue Geräte** Android-Smartphones werden nun durch die UltraLiteJ-Programmierschnittstelle unterstützt. Die UltraLiteJ-Programmierschnittstelle gilt für die Plattformen BlackBerry und Android, und Anwendungen müssen für jede der beiden Plattformen in Java geschrieben werden. Auf BlackBerry-Smartphones ist das zugrundeliegende DBMS eine Java-Implementierung von UltraLite, während auf Android das DBMS die C++- Version ist, die auch für iPhone und iPad, Windows Mobile und Windows bereitgestellt wird.

Siehe:

- „[UltraLite-Überblick](#)“ [[UltraLite - Datenbankverwaltung](#)]
- „[UltraLiteJ-Anwendungsentwicklung](#)“ [[UltraLite® – Java-Programmierung](#)]
- „[Praktische Einführung: Eine Android-Anwendung erstellen](#)“ [[UltraLite® – Java-Programmierung](#)]
- „[UltraLiteJ-API-Referenz](#)“ [[UltraLite® – Java-Programmierung](#)]

Programmierschnittstellen

UltraLite.NET

- **Asynchrone Synchronisation** Die folgenden Mitglieder wurden zur Unterstützung der asynchronen Synchronisation hinzugefügt:
 - **public delegate void ULSyncProgressedDlg(IAsyncResult result, ULSyncProgressData data)** Diese Methode wird während der Synchronisation aufgerufen, um die Fortschrittsinformationen bereitzustellen. Siehe [ULSyncProgressedDlg-Delegat](#) [[UltraLite.NET](#)] [[UltraLite - .NET-Programmierung](#)].

- **public IAsyncResult ULConnection.BeginSynchronize()** Diese Methode erstellt einen neuen Thread zur Durchführung einer Synchronisation und kehrt dann sofort zurück. Siehe [ULConnection.BeginSynchronize-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
- **public IAsyncResult ULConnection.BeginSynchronize(Control control, ULSyncProgressedDlg dlg, object state)** Diese Methode erstellt einen neuen Thread zur Durchführung einer Synchronisation und kehrt dann sofort zurück. Siehe [ULConnection.BeginSynchronize-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
- **public void ULConnection.CancelSynchronize(IAsyncResult asyncResult)** Diese Methode weist die Beendigung des Synchronisations-Threads an und kehrt dann sofort zurück. Siehe [ULConnection.CancelSynchronize-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
- **public void ULConnection.EndSynchronize(IAsyncResult asyncResult)** Diese Methode wird blockiert, bis eine asynchron gestartete Synchronisation beendet ist. Wenn die Synchronisation fehlschlägt, wird eine ULException-Ausnahmebedingung ausgelöst. Siehe [ULConnection.EndSynchronize-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
- **public bool ULSyncProgressData.IsFinalSyncProgress** Diese Eigenschaft ist TRUE, wenn dies die letzte Meldung zum Verarbeitungsfortschritt für eine Synchronisation ist. Siehe [ULSyncProgressData.IsFinalSyncProgress-Eigenschaft \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].

UltraLite für M-Business Anywhere

Diese API wird für die UltraLite-Versionen 12.0.0 und 12.0.1 nicht mehr empfohlen.

UltraLiteJ

- **Unterstützung der Datenbankkonfiguration für Android-Geräte** Die neue Methode DatabaseManager.CreateConfigurationFileAndroid richtet das Objekt ConfigFileAndroid für eine beständige Datenbank ein, die in einer Datei auf einem Android-Gerät gespeichert wird. Siehe [ConfigFileAndroid-Schnittstelle \[Android\]](#) [*UltraLiteJ*] [*UltraLite® – Java-Programmierung*].
- **Android-bezogene Aktualisierungen an der Schnittstelle ConfigPersistent** Die folgenden Mitglieder sind von der UltraLiteJ-Unterstützung für Android betroffen:

Mitgliedsname	Status
enableAesDBEncryption-Methode	Neu, nur für Android. Aktiviert die AES-Verschlüsselung der Datenbank. Siehe ConfigPersistent.enableAesDBEncryption-Methode [UltraLiteJ] [<i>UltraLite® – Java-Programmierung</i>].
getAutoCheckpoint-Methode	Nicht mehr empfohlen für BlackBerry; aktiv für Android.

Mitgliedsname	Status
getConnectionString-Methode	Neu, nur für Android. Ruft eine mit SetConnectionString registrierte Verbindungszeichenfolge auf. Siehe ConfigPersistent.getConnectionString-Methode [Android] [UltraLiteJ] [UltraLite® – Java-Programmierung] .
getCreationString-Methode	Neu, nur für Android. Ruft eine mit SetCreationString registrierte Erstellungszeichenfolge auf. Siehe ConfigPersistent.getCreationString-Methode [Android] [UltraLiteJ] [UltraLite® – Java-Programmierung] .
getDatabaseKey-Methode	Neu, nur für Android. Ruft einen mit SetEncryptionKey registrierten Datenbank-Chiffrierschlüssel ab. Siehe ConfigPersistent.getEncryptionKey-Methode [UltraLiteJ] [UltraLite® – Java-Programmierung] .
getUserName-Methode	Neu, nur für Android. Ruft den mit setUsername festgelegten Namen eines Benutzers ab. Siehe ConfigPersistent.getUserName-Methode [Android] [UltraLiteJ] [UltraLite® – Java-Programmierung] .
setAutocheckpoint-Methode	Nicht mehr empfohlen für BlackBerry; aktiv für Android.
setConnectionString-Methode	Neu, nur für Android. Legt die Verbindungszeichenfolge fest, die zum Erstellen einer Datenbank oder zum Verbinden mit einer Datenbank verwendet werden soll. Siehe ConfigPersistent.setConnectionString-Methode [Android] [UltraLiteJ] [UltraLite® – Java-Programmierung] .
setCreationString-Methode	Neu, nur für Android. Legt die Erstellungszeichenfolge fest, die zum Erstellen einer Datenbank verwendet werden soll. Siehe ConfigPersistent.setCreationString-Methode [Android] [UltraLiteJ] [UltraLite® – Java-Programmierung] .
setDatabaseKey-Methode	Neu, nur für Android. Legt den Schlüssel für die Verschlüsselung fest. Siehe ConfigPersistent.setEncryptionKey-Methode [UltraLiteJ] [UltraLite® – Java-Programmierung] .
setEncryption-Methode	Nur BlackBerry.
setUserName-Methode	Neu, nur für Android. Legt den Namen des Benutzers fest. Siehe ConfigPersistent.setUserName-Methode [Android] [UltraLiteJ] [UltraLite® – Java-Programmierung] .

Weitere Hinweise finden Sie unter [ConfigPersistent-Schnittstelle \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

- Android-bezogene Aktualisierungen der Schnittstelle Connection** Die folgenden Mitglieder sind von der UltraLiteJ-Unterstützung für Android betroffen:

Mitgliedsname	Status
checkpoint-Methode	Nicht mehr empfohlen für BlackBerry; aktiv für Android.
emergencyShutdown-Methode	Nicht verfügbar für Android.
getDatabaseId-Methode	Nicht verfügbar für Android.
getDatabaseProperty-Methode	Nicht verfügbar für Android.
getState-Methode	Nicht verfügbar für Android.
isSynchronizationDeleteDisabled-Methode	Nicht verfügbar für Android.
OPTION_BLOB_FILE_BASE_DIR-Variable	Nur BlackBerry und Java SE.

Weitere Hinweise finden Sie unter [Connection-Schnittstelle \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **Android-bezogene Aktualisierungen der DatabaseManager-Klasse** Die folgenden Mitglieder sind für Android nicht verfügbar:

- createConfigurationNonPersistent-Methode
- createFileTransfer-Methode

Weitere Hinweise finden Sie unter [DatabaseManager-Klasse \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **Android-bezogene Aktualisierungen der Schnittstelle ResultSet** Die folgenden Mitglieder sind für Android nicht verfügbar:

- getBoolean-Methode
- getBytes-Methode
- getClobReader-Methode
- getDate-Methode
- getDecimalNumber-Methode
- getDouble-Methode
- getFloat-Methode
- getLong-Methode
- getSize-Methode

- getString-Methode
- getUUIDValue-Methode
- isNull-Methode

Weitere Hinweise finden Sie unter [ResultSet-Schnittstelle \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen

Nachstehend finden Sie eine Liste der nicht mehr empfohlenen Funktionen und Verhaltensänderungen von UltraLite, die in Version 12.0.1 eingeführt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

Verhaltensänderungen

- **Änderung am Verhalten von SET OPTION** UltraLite führt nun eine Festschreibung durch, wenn beständige Optionen festgelegt werden. Siehe „[SET OPTION-Anweisung \[UltraLite\]](#)“ [*UltraLite - Datenbankverwaltung*].
- **Standardeinstellung für die beständige Netzwerkprotokolloption ist nun TRUE** Der Standardwert für die beständige Netzwerkprotokolloption für MobiLink-Clients (mit Ausnahme von UltraLiteJ) ist nun TRUE. Synchronisationen, die beständige Verbindungen verwenden, sind schneller als nichtbeständige Verbindungen, insbesondere bei HTTPS. Wenn jedoch ein zwischengeschalteter Server nichtbeständige Verbindungen anfordert oder der Client erkennt, dass ein zwischengeschalteter Server beständige Verbindungen nicht unterstützt, führt er automatisch ein Downgrade auf nichtbeständige Verbindungen für den Rest der Synchronisationssitzung durch. Siehe „[persistent](#)“ [*MobiLink - Clientadministration*].
- **Fehlermeldung SQLE_COMMUNICATIONS_ERROR (-85) geändert in SQLE_MOBILINK_COMMUNICATIONS_ERROR (-1305)** Alle Informationen über das jeweilige Kommunikationsproblem sind nun in der Fehlermeldung enthalten. Sie brauchen auf diese Werte nicht mehr von dem separaten Datenstrom-Fehlerobjekt aus zuzugreifen. Bestehender Code, der diese Werte referenziert, funktioniert weiterhin. Die Parameter für diesen Fehler sind der Datenstrom-Fehlercode, Parameter und Systemcode. Siehe „[Verbindungsfehler](#)“ [*Fehlermeldungen*].
- **Fehlermeldung SQLE_UNKNOWN_PROPERTY** SQLE_UNKNOWN_PROPERTY wird nun für unbekannte Eigenschaftsnamen signalisiert. Siehe „[%1 ist eine unbekannte Eigenschaft](#)“ [*Fehlermeldungen*].

- **Fehlermeldung**
SQL_E_MOBILINK_AUTHENTICATION_FAILED `SQL_E_MOBILINK_AUTHENTICATION_FAILED` wird während der Synchronisation signalisiert statt `SQL_E_INVALID_LOGON` und umfasst Authentifizierungsstatus und -werte. Siehe „[Synchronisation ist fehlgeschlagen, da MobiLink den Authentifizierungsstatus '%1' mit Wert '%2' zurückgegeben hat](#)“ [[Fehlermeldungen](#)].

Neue Funktionen der Administrationstools

Im Folgenden finden Sie eine Liste der neuen Administrationstools, die in SQL Remote in Version 12.0.1 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **OEMs können das voreingestellte Verzeichnis von DBISQL, Sybase Central, DBConsole und MobiLink-Monitor festlegen** Wenn Sie das Deployment einer *OEM.ini*-Datei zusammen mit dem der Administrationstools durchführen, müssen Sie darauf achten, dass die folgenden Zeilen in der Datei enthalten sind, um anzugeben, welches Verzeichnis zum Speichern von Voreinstellungen für Administrationstools verwendet wird:

```
[preferences]
directory=preferences_file_directory
```

Siehe „[Konfiguration der Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].

Neue Plug-In-Funktionen in Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen der Sybase Central-Plug-Ins in Version 12.0.1.

Neue Funktionen der SQL Anywhere-Plug-Ins

- **Neue Empfehlung zur Anwendungsprofilerstellung** Wenn Sie mit dem **Assistenten für die Anwendungsprofilerstellung** die Gesamt-Performance Ihrer Datenbank basierend auf ihrem Schema prüfen, kontrolliert der Assistent die Reihenfolge der Spalten in Tabellen. Der Assistent gibt Empfehlungen, wenn die Mehrzahl der breiten Spalten vor den schmalen Spalten liegt. Die Anordnung von breiteren Spalten vor schmälere Spalten kann sich negativ auf die Antwortzeit auswirken. Schmale Spalten sollten in der Tabellendeklaration vor breiteren Spalten definiert sein, es sei denn, es wird selten auf sie zugegriffen.

Breite Spalten sind Spalten mit einer Größe von über 15 Byte oder mit LONG-Datentypen (z.B. LONG VARCHAR) oder als XML definierte Spalten. Siehe „[Tipp: Spaltenreihenfolge in Tabellen überprüfen](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

- **Erweiterungen zur Verwaltung und Anzeige von Benutzerberechtigungen** An der Anzeige von Benutzerberechtigungen im Plug-In angezeigt wurden zahlreiche Erweiterungen vorgenommen. Es ist einfacher, die explizit für ein Datenbankobjekt eingestellten Berechtigungen anzuzeigen und zu ändern. Wenn Sie ein Datenbankobjekt auswählen, werden die Berechtigungsinformationen nun auf einer Registerkarte im rechten Fensterausschnitt angezeigt. Spaltenberechtigungen werden mit den zugehörigen Tabellenberechtigungen angezeigt. Siehe „[Privilegien auf Objektebene erteilen \(Sybase Central\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Verhaltensänderungen der Sybase Central-Plug-Ins

Im Folgenden finden Sie eine Liste von Verhaltensänderungen der Sybase Central-Plug-Ins in Version 12.0.1.

Neue Funktionen der SQL Anywhere-Plug-Ins

- **Sybase Central unterstützt nur Datenbanken der SQL Anywhere-Versionen 10x und höher** Die Unterstützung für Datenbankserver der Version 9 und für Datenbanken, die mit der Version 9 erstellt wurden, wurde aus dem SQL Anywhere-Plug-In entfernt. Wenn Sie die Datenbank entladen und in eine Reload-Datei oder in eine neue oder vorhandene Datenbank neu laden, können Sie dennoch eine Verbindung mit einer Datenbank herstellen, die mit Version 5, 6, 7, 8 oder 9 erstellt wurde und auf einem Datenbankserver der Version 9 oder höher ausgeführt wird. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

Neue Funktionen von Interactive SQL

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Interactive SQL in Version 12.0.1 hinzugefügt wurden.

- **Fenster "Verbinden" unterstützt SAP Sybase IQ** Sie können Interactive SQL verwenden, um eine Verbindung mit einer SAP Sybase IQ-Datenbank herzustellen. Klicken Sie im Fenster **Verbinden** auf **Datenbanktyp ändern** und anschließend auf **SAP Sybase IQ**.
- **Textvervollständigung** Die folgende Liste beschreibt die Änderungen der Textvervollständigungsfunktion in Interactive SQL und Sybase Central:
 - Standardmäßig wird das Textvervollständigungsfenster nun automatisch geöffnet, wenn Sie im Fensterausschnitt **SQL-Anweisungen** eine Eingabe machen. Sie können das Textvervollständigungsfenster auch öffnen, indem Sie auf **Bearbeiten** & **Textvervollständigung öffnen** klicken oder Strg+Leertaste drücken.
 - Standardmäßig stehen SQL-Schlüsselwörter und vollständige Datenbank-Objektnamen zwischen Anführungszeichen.
 - SQL-Anweisungen und Schlüsselwörter sind nun Bestandteil der Liste der Vorschläge.
 - Das Textvervollständigungsfenster kann nun für Datenbanken verwendet werden, in denen bei Zeichenfolgen die Groß-/Kleinschreibung berücksichtigt wird, bei Bezeichnern jedoch nicht.
 - Folgende Tastenkürzel wurden geändert:

Altes Tastenkürzel	Neues Tastenkürzel	Beschreibung
Tabulatortaste	Strg+A	Zeigt eine kontextunabhängige Liste der Übereinstimmungen an.

Altes Tastenkürzel	Neues Tastenkürzel	Beschreibung
+	Strg+Pluszeichen (+)	Fügt das Element mit seiner Parameterliste in den Fensterausschnitt SQL-Anweisungen ein.
*	Strg+Stern	Fügt das Elemente mit seiner Parameter- und Typenliste in den Fensterausschnitt SQL-Anweisungen ein.
'	Strg+Anführungszeichen (")	Fügt das Element in den Fensterausschnitt SQL-Anweisungen ein und setzt es zwischen Anführungszeichen.
k.A.	Tabulatortaste	Akzeptiert die Auswahl und schließt das Textvervollständigungsfenster.

Eine vollständige Liste der Tastenkürzel finden Sie unter „[Tastenkürzel bei der Textvervollständigung](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Weitere Hinweise finden Sie unter „[Textvervollständigung](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Neue Import/Export-Option** Wenn Sie auf **Optionen** und **Import/Export** wählen, können Sie im Feld **NULL-Werte exportieren als** festlegen, wie Nullwerte exportiert werden sollen. Siehe „[output_nulls-Option](#) [*Interactive SQL*]“ [*SQL Anywhere Server - Datenbankadministration*].

Verhaltensänderungen von Interactive SQL

Im Folgenden finden Sie eine Liste der Änderungen von Interactive SQL in Version 12.0.1.

- **Kodierungsalgorithmus für READ-Anweisung geändert** Wenn Sie die READ-Anweisung in Interactive SQL ausführen, wird die zum Lesen der Daten verwendete Kodierung nun in der folgenden Reihenfolge ermittelt:
 1. Die durch die ENCODING-Klausel angegebene Kodierung (sofern diese Klausel angegeben ist).
 2. Die durch die BOM (Bytereihenfolge-Markierung) in der Datei angegebene Kodierung (wenn die Bytereihenfolge-Markierung angegeben ist).
 3. Die standardmäßige TEXT-Formatkodierung (wenn die Formatkodierung angegeben ist).
 4. Die Standardkodierung für die Plattform, die Sie verwenden. Auf Computern mit englischem Windows ist die Standardkodierung 1252.

Siehe „[READ-Anweisung](#) [*Interactive SQL*]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Neue Funktionen des SQL Anywhere-Monitors

Nachstehend finden Sie eine Liste der Funktionen, die dem SQL Anywhere-Monitor in Version 12.0.1 hinzugefügt wurden.

- **Der Monitor kann nun E-Mail-Alarme über SMTP-Server versenden** Der SQL Anywhere-Monitor unterstützt nun das Senden von E-Mail-Alarmen über SMTP-Server, wofür TLS-Verbindungen erforderlich sind (z.B. Gmail). Siehe „[Senden von Alarm-E-Mails im Monitor aktivieren](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Monitor-Datenbanken in Systemen mit Scale-Out mit Schreibschutz und Spiegelungssystemen** Der Monitor unterstützt die Überwachung des Datenbankservers im Stammknoten in Scale-Out-Systemen mit Schreibschutz und des primären Datenbankservers in einem Spiegelungssystem. In einem Spiegelungssystem kann der Monitor so konfiguriert werden, dass er immer den primären Datenbankserver überwacht, selbst wenn ein Failover auftritt. Das neue Widget **Topologie des SQL Anywhere-Scale-Out** zeigt Topologie-Informationen über das SQL Anywhere-Spiegelungssystem und das Scale-Out-System an.

Zum Erstellen einer Ressource für einen Stammknoten in einem Scale-Out-System mit Schreibschutz oder der primären Datenbank in einem Spiegelungssystem verwenden Sie das Fenster **Ressourcen hinzufügen** und wählen Sie die Erstellung einer **SQL Anywhere Server**-Ressource. Siehe „[Datenbankressourcen hinzufügen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Es gibt keinen Unterschied in der Ressourcenkonfiguration für einen Stammknoten in einem Scale-Out-System mit Schreibschutz und anderen Datenbankservern. Wenn die Ressource konfiguriert oder migriert wird, erkennt der Monitor, dass die Ressource der Stammknoten in einem Scale-Out-System mit Schreibschutz ist.

Eine Ressource für eine primäre Datenbank in einem Spiegelungssystem erfordert eine besondere Konfiguration. Um sicherzustellen, dass der Monitor den Server mit der Primärdatenbank immer überwacht, auch wenn ein Failover vorkommt, müssen Sie die Ressource mit den folgenden Optionen konfigurieren:

- Geben Sie im Feld **Host** die Hostnamen und Portnummern der Computer, die auf denen Primär- und Spiegelserver ausgeführt werden, als kommagetrennte Liste ein. Beispiel: `my-primary-server:2638,my-mirror-server:49152`.
- Vergewissern Sie sich, dass das Feld **Port** leer ist.
- Geben Sie im Feld **Servername** den alternativen Namen für den Primärserver ein. Damit definieren Sie den Namen, unter dem sich Clients mit dem Datenbankserver verbinden, der im Datenbankspiegelungssystem als Primärserver agiert.

Sie müssen Sie **Topologie des SQL Anywhere-Scale-Out** Ihrem Dashboard hinzufügen, da es nicht eines der Standard-Widgets ist, die bei Erstellung eines neuen Dashboards angezeigt werden. Siehe „[Widgets](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Siehe „[Lektion 9: \(Optional\) Scale-Out-Systeme mit Schreibschutz aus dem SQL Anywhere-Monitor überwachen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[Lektion 8: Datenbankspiegelungssysteme überwachen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Außerdem wurde der Alarm **Alarm, wenn die Anzahl der getrennten Scale-Out-Knoten den angegebenen Schwellenwert überschreitet** hinzugefügt. Siehe „[Angaben von Alarm-Schwellenwerten](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Neue Monitor-Metrik und neuer Alarm für MobiLink-Server** Die Monitor-Metriken für MobiLink-Serverressourcen enthalten nun eine Metrik namens **Ist der konsolidierte Typ verfügbar?** Diese Metrik zeigt **TRUE** an, wenn der MobiLink-Server eine Verbindung zur konsolidierten Datenbank herstellen kann. Wenn die konsolidierte Datenbank ausfällt, wird der Alarm **Konsolidierte Datenbank verfügbar** ausgegeben.
- **Neue Namen für MobiLink-Ressourcen-Metriken** Die folgenden Metriken des SQL Anywhere-Monitors für MobiLink-Serverressourcen wurden geändert:

Alter Name	Neuer Name
Rate gelesener Byte	Rate gelesener TCP-Byte
Rate geschriebener Byte	Rate geschriebener TCP-Byte

- **Fenster Alarmkonfiguration** Das Fenster **Alarmkonfiguration** enthält nun zwei Optionen, mit denen die Anzahl der E-Mail-Nachrichten pro Tag und der Typ der versendeten E-Mails beschränkt werden kann.

E-Mail-Benachrichtigungen nur für Alarme mit hoher Priorität senden Setzen Sie diese Option für den Empfang von E-Mail-Benachrichtigungen nur, wenn Alarme mit hoher Priorität gemeldet werden.

Nicht mehr als die angegebene Anzahl von E-Mail-Benachrichtigungen pro Tag senden Geben Sie diese Option an, um die Menge der an die einzelnen Benutzer pro Tag versendeten E-Mail-Nachrichten zu beschränken.

Siehe „[Senden von Alarm-E-Mails im Monitor aktivieren](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Fenster Alarme löschen** Wenn der Benutzer einen Alarm wählt und auf die Schaltfläche **Löschen** klickt, wird das Fenster **Alarme löschen** angezeigt, und der Benutzer wird aufgefordert, eine der folgenden Optionen zu wählen:

Ausgewählte Alarme löschen Wählen Sie diese Option, um nur die Alarme zu löschen, die im Widget **Alarmliste** ausgewählt sind.

Alarme löschen, die empfangen wurden vor Wählen Sie diese Option, um alle Alarme zu löschen, die vor dem angegebenen Zeitpunkt empfangen wurden.

Alle Alarme für die folgende Ressource löschen Wählen Sie diese Option, um alle Alarme für die ausgewählte Ressource zu löschen.

Siehe „[Löschen von Alarmen](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Verhaltensänderungen des SQL Anywhere-Monitors

Nachstehend finden Sie eine Liste der Änderungen, die an SQL Anywhere-Monitor in Version 12.0.1 vorgenommen wurden.

- **Maximale Menge von Daten, die exportiert werden können** Die maximale Menge der Daten, die in eine Datei exportiert werden kann, beträgt 25 Metriken oder 1 Million Punkte. Siehe „Exportieren von Metriken“ [[SQL Anywhere Server - Datenbankadministration](#)].

Erweiterungen der Dokumentation

Nachstehend finden Sie eine Liste der Änderungen, die an der Dokumentation von SQL Anywhere in der Version 12.0.1 vorgenommen wurden.

- **Liste der praktischen Einführungen in SQL Anywhere** Eine vollständige Liste der praktischen Einführungen in SQL Anywhere ist nun in der Dokumentation enthalten. Siehe „Liste der praktischen Einführungen in SQL Anywhere“ [[SQL Anywhere 16 - Einführung](#)].

Neue Funktionen in Version 12.0.0

Informationen über neue Funktionen und Verhaltensänderungen in den Versionen von SQL Anywhere vor der Version 10 finden Sie unter <http://dcx.sybase.com/html/dbwnde10/dbwnde10.html>.

Produktweite neue Funktionen

Nachstehend finden Sie eine Liste der neuen, produktübergreifenden Funktionen, die in Version 12.0.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Escapezeichen werden für Konfigurationsdateien unterstützt** Die syntaktische Analyse für Konfigurationsdateien wurde verbessert und akzeptiert jetzt \\ als Escapesequenz für einen \ und \" als Escapesequenz für ein ". Siehe [Escapezeichen in der Konfigurationsdatei \[SQL Anywhere Server - Datenbankadministration\]](#).
- **Deployment-Assistent unterstützt nun 64-Bit-Deployments** Frühere Versionen des Deployment-Assistenten unterstützten 32-Bit-Deployments. Jetzt werden auch 64-Bit-Deployments unterstützt. Siehe „[Deployment-Assistent](#)“ [\[SQL Anywhere Server - Programmierung\]](#).

Produktweite Verhaltensänderungen

Nachstehend finden Sie eine Liste der neuen, produktübergreifenden Verhaltensänderungen, die in Version 12.0.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **SQL Anywhere Replication Agent für Sybase Replication Server wird nicht unterstützt** Der SQL Anywhere Replication Agent für Sybase Replication Server wird in Version 12 nicht unterstützt. Sie müssen eine alternative Replikations- oder Synchronisationstechnologie wie MobiLink oder SQL Remote verwenden. Siehe „[MobiLink-Technologie](#)“ [\[MobiLink - Erste Orientierung\]](#) und „[SQL Remote-Systeme](#)“ [\[SQL Remote\]](#).

Die folgenden Änderungen wurden aufgrund dieser Änderung an der Software vorgenommen:

- **a_change_log DBTools-Struktur** Das Mitglied ignore_ltm_trunc wird nicht mehr unterstützt.
- **LTMGeneration-Datenbankeigenschaft** Diese Eigenschaft ist für die Verwendung durch das System reserviert.
- **LMTTrunc-Datenbankeigenschaft** Diese Eigenschaft ist für die Verwendung durch das System reserviert.
- **Log-Transfer-Manager-Dienstprogramm (dbltm)** Dieses Dienstprogramm wurde entfernt.
- **Dienstprogramm zur Transaktionslogkonvertierung (dbtran)** Die Option -is unterstützt den Wert RepServer nicht mehr.

Die Option -rsu wurde entfernt.

- **Dienstprogramm für Dienste (dbsvc)** Sie können keine Dienste mehr für den Replication Agent erstellen. Die Dienstgruppe SQLANYLTM wird nicht mehr unterstützt.

Die Optionen -w und -t unterstützen den Wert **dbltm** nicht mehr.

- **Dienstprogramm für Supportanfragen (dbsupport)** Dieses Dienstprogramm gibt keine Informationen über den SQL Anywhere Replication Agent (dbltm) zurück.
- **Transaktionslog-Dienstprogramm (dblog)** Die Optionen -g und -il werden nicht mehr unterstützt. Siehe „[Transaktionslog-Dienstprogramm \(dblog\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **replicate_all-Datenbankoption** Diese Option wurde entfernt.
- **delete_old_logs-Datenbankoption** Diese Option wird nicht für die Nutzung durch den Replication Agent unterstützt.
- **ALTER PROCEDURE-Anweisung** Die folgende Syntax wird nicht mehr unterstützt:

```
ALTER PROCEDURE [ owner.]procedure-name  
  REPLICATE { ON | OFF }
```

Siehe „[ALTER PROCEDURE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **ALTER TABLE-Anweisung** Die Klausel **REPLICATE { ON | OFF }** wird nicht mehr unterstützt. Siehe „[ALTER TABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Ausführbare Programme halten sich jetzt an die Einstellungen des Benutzers für die umask-Funktion, wenn sie als Daemon ausgeführt werden.** In früheren Versionen galt: Wenn ein ausführbares Programm unter Unix als Daemon ausgeführt wurde (mit der Option -ud gestartet), ignorierte das ausführbare Programm die Einstellungen des Benutzers für die umask-Funktion und rief umask(0) auf. Dadurch wurden neue Dateien mit Gruppen- und anderen Schreib-/Leseberechtigungen erstellt. Wenn Sie eine ausführbare SQL Anywhere 12-Datei als Daemon starten, ruft sie umask(0) nicht auf und respektiert die umask-Einstellungen des Benutzers. Da die umask-Einstellung des aktuellen Benutzers die Berechtigungen für ausführbare Dateien steuert, müssen Sie sicherstellen, dass der umask-Wert des Benutzers auf die gewünschte Ebene eingestellt ist, bevor Sie die ausführbare Datei starten.

Diese Verhaltensänderung gilt für folgende ausführbare Dateien:

- dbeng12
- dbsrv12
- dbltm
- dbmlsync
- dbns12
- dbremote
- mlsrv12
- uleng12

Neue Funktionen in SQL Anywhere

Im Folgenden finden Sie eine Liste neuer Funktionen in SQL Anywhere Version 12.0.0.

Weitere Hinweise zu den Änderungen der unterstützten Plattformen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Wichtigste Funktionen

Im Folgenden finden Sie eine Liste der wichtigsten Funktionen, die in SQL Anywhere Version 12.0.0 eingeführt wurden.

- **Neue Unterstützung räumlicher Daten** Die folgenden Funktionen wurden zur Unterstützung der neuen Funktionalität für räumliche Daten in SQL Anywhere 12.0.0 hinzugefügt. Sie müssen ein Upgrade Ihrer Datenbank durchführen, um diese Funktionen verwenden zu können.

Hinweis

Die Unterstützung räumlicher Daten für 32-Bit-Windows und 32-Bit Linux erfordert eine CPU, die SSE2-Anweisungen verarbeiten kann. Diese Unterstützung ist bei Prozessoren der Serie Intel Pentium 4 oder höher (ab 2001) und AMD Opteron oder höher (ab 2003) gegeben.

SQL-Anweisungen Die folgenden Erweiterungen für SQL-Anweisungen wurden zur Unterstützung der räumlichen Funktionalität hinzugefügt:

- **SHAPEFILE-Klausel** Für die OPENSTRING-Unterklausel der FROM-Klausel steht eine neue SHAPEFILE-Formatoption zur Verfügung. Siehe „FROM-Klausel“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Für die FORMAT-Klausel der LOAD TABLE-Anweisung ist ebenfalls eine neue SHAPEFILE-Formatoption verfügbar. Siehe „LOAD TABLE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

- **CREATE SPATIAL REFERENCE SYSTEM-Anweisung** Erstellt oder ersetzt ein räumliches Bezugssystem. Siehe „CREATE SPATIAL REFERENCE SYSTEM-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

- **ALTER SPATIAL REFERENCE SYSTEM-Anweisung** Ändert die Einstellungen eines vorhandenen räumlichen Bezugssystems. Vor dem Ändern eines räumlichen Bezugssystems lesen Sie bitte die Hinweise im Abschnitt "Bemerkungen". Siehe „[ALTER SPATIAL REFERENCE SYSTEM-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **DROP SPATIAL REFERENCE SYSTEM-Anweisung** Löscht ein räumliches Bezugssystem. Siehe „[DROP SPATIAL REFERENCE SYSTEM-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **CREATE SPATIAL UNIT OF MEASURE-Anweisung** Erstellt oder ersetzt eine räumliche Maßeinheit. Siehe „[CREATE SPATIAL UNIT OF MEASURE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **DROP SPATIAL UNIT OF MEASURE-Anweisung** Löscht eine räumliche Maßeinheit. Siehe „[DROP SPATIAL UNIT OF MEASURE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Änderungen bei Interactive SQL Interactive SQL wurde durch das neue Anzeigetool **Spatial Viewer** ergänzt, mit dem räumliche Geometrien angezeigt werden können. Räumliche Daten lassen sich im oberen Bereich des Anzeigetools abfragen, die Ergebnisse werden als Bild im unteren Bereich angezeigt. Siehe „[Anzeigen von räumlichen Daten als Bilder \(Interactive SQL\)](#)“ [*SQL Anywhere Server - Unterstützung für räumliche Daten*].

Wenn Sie eine Ergebniszeile in Interactive SQL anzeigen, können Sie mithilfe der neuen Registerkarte **Räumliche Vorschau** die Vorschau einer Geometrie in Form einer skalierbaren Vektorgrafik (SVG) anzeigen. Siehe „[Anzeigen von räumlichen Daten als Bilder \(Interactive SQL\)](#)“ [*SQL Anywhere Server - Unterstützung für räumliche Daten*].

Neue Datentypen, Methoden und Konstruktoren Neue Datentypen, Methoden und Konstruktoren ermöglichen den Zugriff auf räumliche Daten sowie ihre Modellierung und Analyse. Siehe „[Zugriff auf räumliche Daten und ihre Verarbeitung](#)“ [*SQL Anywhere Server - Unterstützung für räumliche Daten*].

Dank zahlreicher neuer Kompatibilitätsfunktionen lassen sich beim Aufrufen und Bearbeiten räumlicher Daten reguläre SQL-Funktionen simulieren. Diese Funktionen bieten Kompatibilität mit anderen Produkten und nutzen die Methoden und Konstruktoren für räumliche Daten in SQL Anywhere. Siehe „[Funktionen der räumlichen Kompatibilität](#)“ [*SQL Anywhere Server - Unterstützung für räumliche Daten*].

Neue Funktionen und Systemprozeduren Die folgenden Funktionen und Systemprozeduren wurden zur Unterstützung räumlicher Daten in der Datenbank hinzugefügt:

- **TREAT-Funktion** Damit ändern Sie den deklarierten Typ eines geometrischen Ausdrucks in einen Subtyp. Siehe „[TREAT-Funktion \[Datentypkonvertierung\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **sa_describe_shapefile-Systemprozedur** Beschreibt die Namen und Typen der Spalten in einer ESRI-Formdatei. Diese Systemfunktion wird mit der räumlichen Funktionalität verwendet. Siehe „[sa_describe_shapefile-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

- **sa_install_feature-Systemprozedur** Installiert zusätzliche Funktionen, die bei der Installation von SQL Anywhere nicht in der Datenbank vorhanden waren. Siehe „sa_install_feature-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **st_geometry_dump-Systemprozedur** Gliedert ein Geometrieobjekt in eine Ergebnismenge auf. Dabei stellt jede Zeile ein Geometrieobjekt aus der Eingabe dar. Siehe „st_geometry_dump-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Assistenten In Sybase Central wurden die folgenden Assistenten zur Unterstützung der räumlichen Funktionalität hinzugefügt:

- **Assistent zum Erstellen eines räumlichen Bezugssystems** Der **Assistent zum Erstellen eines räumlichen Bezugssystems** ermöglicht die Erstellung neuer räumlicher Bezugssysteme. Siehe „Räumliche Bezugssysteme erstellen“ [[SQL Anywhere Server - Unterstützung für räumliche Daten](#)].
- **Assistent Maßeinheit erstellen** Der Assistent **Maßeinheit erstellen** ermöglicht es Ihnen, neue Maßeinheiten für räumliche Daten zu erstellen. Siehe „Maßeinheiten erstellen“ [[SQL Anywhere Server - Unterstützung für räumliche Daten](#)].

Katalogänderungen Zur Unterstützung der neuen räumlichen Funktionalität wurden die folgenden Änderungen des Katalogs vorgenommen:

- **SYSSPATIALREFERENCESYSTEM-Systemansicht** Jede Zeile der SYSSPATIALREFERENCESYSTEM-Systemansicht beschreibt ein räumliches Bezugssystem, das in der Datenbank festgelegt ist. Siehe „SYSSPATIALREFERENCESYSTEM-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **SYSUNITOFMEASURE-Systemansicht** Jede Zeile der SYSUNITOFMEASURE-Systemansicht beschreibt eine Maßeinheit, die in der Datenbank festgelegt ist. Siehe „SYSUNITOFMEASURE-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Konsolidierte ST_GEOMETRY_COLUMNS-Ansicht** Jede Zeile der ST_GEOMETRY_COLUMNS-Systemansicht beschreibt eine Spalte mit räumlichen Daten, die in der Datenbank festgelegt ist. Siehe „Konsolidierte Ansicht ST_GEOMETRY_COLUMNS“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Konsolidierte ST_SPATIAL_REFERENCE_SYSTEMS-Ansicht** Jede Zeile der ST_SPATIAL_REFERENCE_SYSTEMS-Systemansicht beschreibt ein räumliches Bezugssystem, das in der Datenbank festgelegt ist. Siehe „Konsolidierte Ansicht ST_SPATIAL_REFERENCE_SYSTEMS“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Konsolidierte ST_UNITS_OF_MEASURE-Ansicht** Jede Zeile der ST_UNITS_OF_MEASURE-Systemansicht beschreibt eine Maßeinheit, die in der Datenbank festgelegt ist. Siehe „Konsolidierte Ansicht ST_UNITS_OF_MEASURE“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Datenbankoptionen und -eigenschaften Die folgenden Datenbankoptionen und -eigenschaften wurden zur Unterstützung der räumlichen Funktionalität hinzugefügt.

- **st_geometry_asbinary_format-Option** Steuert, wie räumliche Werte vom Geometrieformat in Binärformat umgewandelt werden. Siehe „st_geometry_asbinary_format-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_astext_format-Option** Steuert, wie räumliche Werte vom Geometrieformat in Text umgewandelt werden. Siehe „st_geometry_astext_format-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_asxml_format-Option** Steuert, wie räumliche Werte vom Geometrieformat in XML umgewandelt werden. Siehe „st_geometry_asxml_format-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_describe_type-Option** Steuert, wie räumliche Werte beschrieben werden. Siehe „st_geometry_describe_type-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_on_invalid-Option** Steuert das Verhalten, wenn die Basisvalidierung einer Geometrie fehlschlägt. Siehe „st_geometry_on_invalid-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_asbinary_format-Verbindungseigenschaft** Gibt einen Wert zurück, der darauf hinweist, wie räumliche Werte in einer Geometrie in Binärwerte umgewandelt werden. Siehe [st_geometry_asbinary_format-Verbindungseigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_astext_format-Verbindungseigenschaft** Gibt einen Wert zurück, der darauf hinweist, wie räumliche Werte in einer Geometrie in Text umgewandelt werden. Siehe [st_geometry_astext_format-Verbindungseigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_asxml_format-Verbindungseigenschaft** Gibt einen Wert zurück, der darauf hinweist, wie räumliche Werte in einer Geometrie in XML-Code umgewandelt werden. Siehe [st_geometry_asxml_format-Verbindungseigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_describe_type-Verbindungseigenschaft** Gibt einen Wert zurück, der darauf hinweist, wie räumliche Werte für den Client beschrieben werden. Siehe [st_geometry_describe_type-Verbindungseigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_on_invalid-Verbindungseigenschaft** Gibt einen Wert zurück, der das Verhalten anzeigt, wenn die grundlegende Validierung einer Geometrie fehlschlägt. Siehe [st_geometry_on_invalid-Verbindungseigenschaft](#) [[SQL Anywhere Server - Datenbankadministration](#)].

SYS_SPATIAL_ADMIN_ROLE-Gruppe Mitglieder dieser Gruppe können räumliche Bezugssysteme und Maßeinheiten erstellen, ändern oder löschen.

Weitere Hinweise zur Unterstützung räumlicher Funktionen in SQL Anywhere finden Sie unter „Räumliche Daten“ [[SQL Anywhere Server - Unterstützung für räumliche Daten](#)].

- **Scale-Out mit Schreibschutz** Sie können SQL Anywhere nun in einem schreibgeschützten Scale-Out-System verwenden. In dieser Konfiguration führt ein Datenbankserver (der Stammknoten) eine Lese-Schreib-Kopie der Datenbank aus, während andere Datenbankserver schreibgeschützte Kopien der Datenbank (Kopieknoten) ausführen. Damit können Sie Berichtserstellungs- oder andere Vorgänge auslagern, für die nur Lesezugriff auf die Datenbank erforderlich ist. Das Scale-Out mit Schreibschutz kann entweder allein oder zusammen mit einer Datenbankspiegelung erfolgen. Sie müssen ein Upgrade oder einen Neuaufbau bestehender Datenbanken durchführen, um diese Funktion verwenden zu können.

Im Verzeichnis %SQLANYSDMP12%\SQLAnywhere\DBMirror finden Sie nun ein Beispiel für die Verwendung eines Datenbankspiegelungssystems mit einem Scale-Out-System.

Siehe:

- „SQL Anywhere-Scale-Out mit Schreibschutz“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Verbindungsparameter NodeType (NODE)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „CREATE MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SET MIRROR OPTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- MIRROR SERVER-Klausel: „COMMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSMIRROROPTION-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSMIRRORSERVER-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSMIRRORSERVEROPTION-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- MirrorRole-Datenbankeigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
- MirrorServerState- und MirrorState-Eigenschaften: „sa_mirror_server_status-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Erweiterungen der Datenbankspiegelung** Sie können nun ein Datenbankspiegelungssystem mithilfe von SQL-Anweisungen einrichten, anstatt die Spiegelungseinstellungen in der Befehlszeile des Datenbankservers festzulegen. Sie müssen ein Upgrade oder einen Neuaufbau bestehender Datenbanken durchführen, um diese Funktion verwenden zu können.

Siehe:

- „CREATE MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SET MIRROR OPTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- Neue MIRROR SERVER-Klausel für „COMMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSMIRROROPTION-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSMIRRORSERVER-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSMIRRORSERVEROPTION-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- MirrorRole-Datenbankeigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
- Erweiterte Datenbankeigenschaften: MirrorServerState, MirrorState
- „sa_mirror_server_status-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- Verhaltensänderungen der Datenbankspiegelung und nicht mehr empfohlene Funktionen auf Seite 151

- **Host-Verbindungsparameter** Der neue Host-Verbindungsparameter verwendet einen Hostnamen (oder eine IP-Adresse) und eine optionale Portnummer, die den Client darauf hinweist, wo sich der Datenbankserver befindet. Dieser Verbindungsparameter wird nun für die Verbindung mit Datenbankservern empfohlen, die auf einem anderen Computer ausgeführt werden als der Client. Siehe „Host-Verbindungsparameter“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen der automatischen Statistikverwaltung** SQL Anywhere 12 enthält einen Statistikwächter, der die automatische Verwaltung von Statistiken in Datenbankspalten verbessert. Zustand und Brauchbarkeit jeder Statistik in der Datenbank werden automatisch überprüft und erforderliche Verwaltungsvorgänge werden durchgeführt, sodass die Statistiken selbstüberwachend und selbstheilend sind. Die Statistiken werden im Hintergrund verwaltet, sodass die Performance des Datenbankservers kaum beeinflusst wird. Siehe „Verwaltung von Statistiken durch den Statistikwächter“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Die sa_server_option-Systemprozedur unterstützt nun die folgenden Optionen für die Verwaltung von statistischen Daten: DropBadStatistics, DropUnusedStatistics und StatisticsCleaner. Siehe „sa_server_option-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Sequenzen** SQL Anywhere unterstützt nun die Erstellung von Sequenzen. Sequenzen können von Anwendungen verwendet werden, um eindeutige Schlüsselwerte zu generieren. Die Verwendung von Sequenzwerten kann Anwendungen helfen, Parallelitäts- und Performanceprobleme zu verhindern.

Sie können mit dem SQL Anywhere-Plug-In in Sybase Central auch Sequenzen erstellen, bearbeiten und verwalten. Sie können zum Beispiel mit dem **Assistenten zur Erstellung eines Sequenzgenerators** eine neue Sequenz in der Datenbank erstellen.

Siehe auch:

- „Verwendung einer Sequenz zum Generieren von eindeutigen Werten“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
- „CREATE SEQUENCE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER SEQUENCE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP SEQUENCE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSSEQUENCE-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SYSSEQUENCEPERM-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- SEQUENCE-Klausel, „COMMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- GRANT USAGE ON SEQUENCE-Syntax, „GRANT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- REVOKE USAGE ON SEQUENCE-Syntax, „REVOKE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Sie müssen ein Upgrade oder einen Neuaufbau bestehender Datenbanken durchführen, um Sequenzen verwenden zu können.

- **Erweiterungen der Multiprogramming-Stufe** Der Netzwerk-Datenbankserver (dbsrv12) führt die automatische Steuerung der Multiprogramming-Stufe nun standardmäßig durch. Dadurch kann der Datenbankserver seinen Durchsatz erhöhen und ohne Unterstützung des DBA auf Änderungen der Systemlast reagieren.

Wenn der Datenbankserver startet, erstellt er einen Pool von Worker-Threads, die verwendet werden, um Anforderungen zu verarbeiten. Die Anzahl der Worker-Threads ist die aktuelle Multiprogramming-Stufe des Servers. Der Pool ist auf minimale und maximale Werte beschränkt, und die aktuelle Multiprogramming-Stufe liegt immer innerhalb dieser Vorgaben. Der DBA kann die minimalen und maximalen Werte beim Serverstart mithilfe von Serveroptionen oder während der Ausführung des Servers mithilfe der Systemprozedur `sa_server_option` ändern.

Die folgenden Optionen wurden hinzugefügt, um Ihnen die Steuerung der Multiprogramming-Stufe des Datenbankservers zu ermöglichen:

Datenbankserveroption	sa_server_option-Wert	Beschreibung
„Datenbankserveroption -gn“ [SQL Anywhere Server - Datenbank-administration]	CurrentMultiProgrammingLevel	Legt die Multiprogramming-Ausgangsstufe des Datenbankservers fest.
„Datenbankserveroption -gna“ [SQL Anywhere Server - Datenbank-administration]	AutoMultiProgrammingLevel	Aktiviert und deaktiviert die dynamische Optimierung der Multiprogramming-Stufe des Datenbankservers.

Datenbankserveroption	sa_server_option-Wert	Beschreibung
„Datenbankserveroption -gnh“ [SQL Anywhere Server - Datenbank-administration]	MaxMultiProgrammingLevel	Legt die maximale Anzahl von Aufgaben fest, die der Datenbankserver gleichzeitig ausführen kann
„Datenbankserveroption -gnl“ [SQL Anywhere Server - Datenbank-administration]	MinMultiProgrammingLevel	Legt die minimale Anzahl von Aufgaben fest, die der Datenbankserver gleichzeitig ausführen kann.
„Datenbankserveroption -gns“ [SQL Anywhere Server - Datenbank-administration]	AutoMultiProgrammingLevelStatistics	Steuert, ob statistische Daten über die automatischen Änderungen in der Multiprogramming-Stufe in die Meldungslogdatei des Datenbankservers geschrieben werden.

Weitere Hinweise zur Multiprogramming-Stufe in SQL Anywhere finden Sie unter „Datenbankserverkonfiguration der Multiprogramming-Stufe“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Materialisierte Sofortansichten unterstützen nun Joins** Materialisierte Ansichten mit Outer-Joins in den Definitionen können nun als Sofortansicht deklariert werden. Siehe „Einschränkungen für materialisierte Ansichten“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Auswahl mithilfe von DML-Anweisungen** Sie können nun eine DML-Anweisung in der FROM-Klausel der SELECT-Anweisung angeben. Mit dieser Funktion können Sie SQL-Abfragen über eine abgeleitete Tabelle schreiben, die mit den Zeilen gefüllt wird, die von einer UPDATE-, INSERT-, DELETE- oder MERGE-Anweisung geändert wurden, und Sie können Werte aus diesen aktualisierten Zeilen an die Anwendung zurückgeben.

Diese Funktion wird meist verwendet, um die Werte von Zeilen zu überprüfen oder zu validieren, die von der Anwendung geändert wurden. In früheren Versionen mussten hierzu ein Trigger und mehrere SQL-Anweisungen verwendet werden. Siehe „FROM-Klausel“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „SELECT über eine DML-Anweisung“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

- **Volltextsuchfunktion unterstützt nun externe Vorfilter und Begriffsegmentierer-Bibliotheken** Eine neue API gestattet es, eine Verbindung mit Bibliotheken externer Vorfilter und Begriffsegmentierer herzustellen, wenn vollständige Textindizes erstellt und aktualisiert werden. Das bedeutet, dass Sie Dokumentformate wie XML, PDF und Word verwenden und unerwünschte Tags und Metadaten vor der Indizierung des Inhalts entfernen können. Beispielbibliotheken für Begriffsegmentierer können verwendet werden, um die Sprachen- oder anwendungsspezifische Begriffsegmentierung durchzuführen. Die Beispieldvfilter- und Begriffsegmentierer-Bibliotheken werden bereitgestellt, um Sie bei der Planung eigener Bibliotheken zu unterstützen. Sie können aber

auch Bibliotheken von Drittherstellern verwenden. Siehe „[Fortgeschrittene Aufgaben: Externe Begriffsegmentierer- und Vorfilterbibliotheken](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Wenn Microsoft Office auf dem System installiert ist, auf dem der Datenbankserver ausgeführt wird, sind IFilters for Office-Dokumente, wie z.B. Word und Excel, verfügbar. Wenn auf dem Server Acrobat Reader installiert ist, dann steht wahrscheinlich ein PDF IFilter zur Verfügung.

Die PREFILTER EXTERNAL NAME-Klausel und die TERM BREAKER EXTERNAL NAME-Klausel wurden der ALTER TEXT CONFIGURATION-Anweisung hinzugefügt. Sie gestatten es Ihnen, den Namen und den Speicherort Ihrer externen Bibliotheken festzulegen. Siehe „[ALTER TEXT CONFIGURATION-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Die ISYSTETEXTCONFIG-Systemtabelle speichert nun Informationen über die Eintrittspunkte und die externen Bibliotheken für Token und/oder Vorfilter. Der vorhandene Spaltentyp für Vorfilter wurde in LONG VARCHAR geändert, um die Eintrittspunkte und den Namen für eine externe Vorfilterbibliothek aufnehmen zu können. Darüber hinaus wurde die neue LONG VARCHAR-Spalte external_term_breaker hinzugefügt, damit die Eintrittspunkte und der Name für eine externe Begriffsegmentierer-Bibliothek berücksichtigt werden können. Siehe „[SYSTETEXTCONFIG-Systemansicht](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Sie müssen ein Upgrade der Datenbank vornehmen, um externe Vorfilter- und Begriffsegmentierer-Bibliotheken zu verwenden.

- Prüfsummenerweiterungen** Der Datenbankserver legt nun basierend auf der Datenbankversion fest, ob Schreib-Prüfsummen für Datenbankseiten erstellt werden (Prüfsummen, die nur erstellt werden, wenn die Seiten auf die Festplatte geschrieben werden). Standardmäßig sind für Datenbanken der Versionen 10 und 11 keine globale Prüfsummen aktiviert, und für Datenbanken der Version 12 sind globale Prüfsummen aktiviert. Wenn Sie eine ältere Datenbank auf einem Datenbankserver der Version 12 starten, ist das Standardverhalten des Datenbankservers die Aktivierung von Schreib-Prüfsummen. Bei Datenbanken der Version 12 ist das Standardverhalten des Datenbankservers, Schreib-Prüfsummen nicht zu aktivieren, da standardmäßig für Datenbanken der Version 12 globale Prüfsummen aktiviert sind. Siehe [Prüfsummen sind standardmäßig für neue Datenbanken aktiviert auf Seite 148](#).

Sie können beim Start einer Datenbank oder eines Datenbankservers die CHECKSUM-Klausel der START DATABASE-Anweisung oder die Option -wc verwenden, um das Verhalten des Datenbankservers für Schreib-Prüfsummen zu ändern. Siehe „[Datenbankoption -wc](#)“ [*SQL Anywhere Server - Datenbankadministration*], „[Datenbankserveroption -wc](#)“ [*SQL Anywhere Server - Datenbankadministration*] und „[START DATABASE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Sie können die Prüfsummen für eine Datenbank mit der CHECKSUM-Klausel der ALTER DATABASE-Anweisung deaktivieren. Siehe „[ALTER DATABASE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Datenbankverbindungen

Die folgende Liste enthält die Erweiterungen für die Datenbankverbindungen in SQL Anywhere Version 12.0.0.

- **Temporäre Verbindungen werden benannt** Temporäre Verbindungen ermöglichen zum Beispiel das Durchführen von Sicherungen oder das Initialisieren von Datenbanken. Informationen zu temporären Verbindungen erhalten Sie mithilfe der Systemprozeduren `sa_conn_info` und `sa_conn_list` sowie den Verbindungseigenschaften `Name` und `ParentConnection`. Siehe:
 - „Temporäre Verbindungen“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - `Name`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
 - `ParentConnection`-Verbindungseigenschaft [[SQL Anywhere Server - Datenbankadministration](#)]
 - „`sa_conn_info`-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „`sa_conn_list`-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Verbindungspooling** Der Verbindungsparameter `ConnectionPool` steuert das Verhalten des Clientverbindungs pooling. Siehe „Verbindungsparameter `ConnectionPool` (CPOOL)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Verbindungspooling“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung von Escape-Verbindungsparametern für ODBC-Datenquellen** Der ODBC-Treiber verwendet standardmäßig die Tilde (~) als Escapezeichen. Einige Anwendungen erwarten jedoch den Backslash (\) als Escapezeichen. Mit dem Escape-Verbindungsparameter können Sie das Escapezeichen für Ihre Anwendung festlegen. Siehe [Escape-Verbindungsparameter \(ODBC\)](#) [[SQL Anywhere Server - Datenbankadministration](#)].

Darüber hinaus können Sie den Escape-Verbindungsparameter im Fenster **Verbinden** angeben.

Sicherung und Wiederherstellung

Die folgende Liste enthält alle Erweiterungen zur Sicherung und Wiederherstellung in SQL Anywhere Version 12.0.0.

- **Wiederherstellung von Archivsicherungen mit Version 12** Ab Version 12 können Sie keine Archivsicherungen wiederherstellen, die mit Version 11 oder früheren Datenbankservern erstellt wurden.
- **Eliminierung freier Seiten in Archivsicherungen** Standardmäßig werden bei Archivsicherungen freie Seiten übersprungen, um das Speichervolumen zu reduzieren und den Vorgang zu beschleunigen. Da Transaktionslogdateien keine freien Seiten enthalten, wirkt sich die Eliminierung freier Seiten nicht auf die Sicherung von Transaktionslogdateien aus. Das bedeutet, die Eliminierung freier Seiten hat möglicherweise geringere Auswirkungen auf Datenbanken mit umfangreichen Transaktionslogdateien als auf Datenbanken mit kleinen Transaktionslogdateien. Sie können dieses Verhalten mit der `FREE PAGE ELIMINATION`-Klausel der `BACKUP`-Anweisung oder mit dem **Assistenten zum Sichern einer Datenbank** steuern. Siehe „[BACKUP-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Wenn Sie eine stark verschlüsselte Datenbank wiederherstellen, die mit aktivierter Eliminierung freier Seiten gesichert wurde, müssen Sie den Chiffrierschlüssel für die Datenbank angeben. Hierfür verwenden Sie die `KEY`-Klausel der `RESTORE DATABASE`-Anweisung oder den **Assistenten zum Wiederherstellen einer Datenbank**.

Sicherheit

Die folgende Liste enthält alle Erweiterungen zur Sicherheit in SQL Anywhere Version 12.0.0.

- **FIPS-zertifizierte Algorithmen sind jetzt für 64-Bit-Windows sowie 32- und 64-Bit-Linux verfügbar** FIPS-zertifizierte Algorithmen können jetzt in 64-Bit-Windows sowie 32- und 64-Bit-Linux verwendet werden.

Eine Liste der Plattformen, die FIPS-zertifizierte Algorithmen unterstützen, finden Sie unter „Unterstützte Plattformen“ [[SQL Anywhere 16 - Einführung](#)].

Hinweise zur Verwendung von FIPS-zertifizierten Algorithmen finden Sie unter „FIPS-zertifizierte Verschlüsselungstechnologie“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbanksicherheit

Sie müssen ein Upgrade der Datenbank durchführen oder die Datenbank neu aufbauen, um diese Änderungen verwenden zu können. Siehe „SQL Anywhere-Server-Ugrades“ auf Seite 295.

- **SYS_SPATIAL_ADMIN_ROLE-Gruppe** Mitglieder dieser Gruppe können räumliche Bezugssysteme und Maßeinheiten erstellen, ändern oder löschen.

Datenbank-Dienstprogramme

Die folgende Liste enthält die Erweiterungen der Datenbank-Dienstprogramme in SQL Anywhere Version 12.0.0.

- **Die Datenbankseitengrößen können nun in Kilobyte angegeben werden.** Mit den Dienstprogrammen Extraktion (dbxtract), Initialisierung (dbinit) und Entladen (dbunload) können Sie die Datenbankseitengröße in Byte oder Kilobyte angeben. Bisher war nur eine Angabe in Byte möglich. Siehe:
 - „Extraktionsdienstprogramm (dbxtract)“ [[SQL Remote](#)]
 - „Dienstprogramm Initialisierung (dbinit)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „Dienstprogramm zum Entladen (dbunload)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- **Erweiterungen des Dienstprogramms zum Verschleiern von Dateien (dbfhide)** In Windows unterstützt das Dienstprogramm zum Verschleiern von Dateien (dbfhide) nun die Verwendung einer Windows-Verschlüsselungs-API zur Verschleierung von Konfigurationsdateien. Das Dienstprogramm unterstützt außerdem die Option -q für eine Ausführung im stillen Modus. Siehe „Dienstprogramm zum Verschleiern von Dateien (dbfhide)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen des Dienstprogramms für die Serverlizenzierung (dblic)** Für den Umstieg auf eine andere SQL Anywhere-Edition war es in früheren Versionen notwendig, die Software zu deinstallieren und mit einem neuen Lizenzschlüssel zu installieren. Zu diesem Zweck können Sie nun die Option -k für das Dienstprogramm zur Serverlizenzierung (dblic) verwenden, nachdem Sie einen

neuen Lizenzschlüssel von Sybase erhalten haben. Siehe „Dienstprogramm für die Serverlizenzierung (dblic)“ [\[SQL Anywhere Server - Datenbankadministration\]](#).

- **Erweiterungen des Dienstprogramms für Dienste (dbsvc)** Beim Festlegen von Abhängigkeiten für Dienste mithilfe der Option -rs des dbsvc-Dienstprogramms können Sie unter Windows nun auch den Anzeigenamen oder Dienstnamen angeben. Siehe „Dienstprogramm für Dienste (dbsvc) für Windows“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
- **Erweiterungen des Dienstprogramms für Supportanfragen (dbsupport)** Mithilfe der Option -ac können Sie einen Kommentar hinzufügen, der zum Fehlerbericht hinzugefügt wird. Um eine Datei mit dem Fehlerbericht einzusenden, verwenden Sie die Option -af. Siehe „Dienstprogramm für Supportanfragen (dbsupport)“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
- **Erweiterungen des Dienstprogramms Entladen (dbunload)** Das Dienstprogramm dbunload unterstützt nun die folgenden Optionen:
 - Die Option -kd entlädt eine Datenbank in eine einzelne DBSpace-Datei. Siehe „Dienstprogramm zum Entladen (dbunload)“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
 - Mit der Option -qr wird die Erstellung und Anzeige von Fortschrittsmeldungen beim Laden von Tabellen und Erstellen von Indizes unterdrückt. Siehe „Dienstprogramm zum Entladen (dbunload)“ [\[SQL Anywhere Server - Datenbankadministration\]](#).

Datenbankoptionen

Die folgende Liste enthält die Erweiterungen der Datenbankoptionen in SQL Anywhere Version 12.0.0.

- **uuid_has_hyphens-Option** Die uuid_has_hyphens-Funktion steuert die Formatierung von Werten für eindeutige Bezeichner, wenn sie in Zeichenfolgen konvertiert werden. Diese Funktion wurde in Version 11 entfernt und wird nun wieder eingesetzt. Siehe „uuid_has_hyphens-Option“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
- **blocking_others_timeout-Option** Mit dieser Option legen Sie fest, wie lange die Zeilen und Tabellen der aktuellen Verbindung durch eine andere Verbindung blockiert werden, bevor die aktuelle Verbindung zurückgesetzt wird. So können Sie zum Beispiel vermeiden, dass eine Aufgabe mit niedriger Priorität andere Verbindungen länger als angegeben blockiert. Siehe „blocking_others_timeout-Option“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
- **http_connection_pool_basesize-Option** Diese Option legt eine Basisgröße für HTTP-Verbindungspools fest. Siehe „http_connection_pool_basesize-Option“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
- **http_connection_pool_timeout-Option** Damit legen Sie die maximale Dauer fest, die eine nicht verwendete Verbindung in einem HTTP-Verbindungspool behalten wird. Siehe „http_connection_pool_timeout-Option“ [\[SQL Anywhere Server - Datenbankadministration\]](#).
- **progress_messages-Option** Diese Option steuert, ob Meldungen zum Verarbeitungsfortschritt vom Datenbankserver zum Client gesendet werden. Sie können diese Option als temporäre Option für die Dienstprogramm Datenbank setzen. Siehe „progress_messages-Option“ [\[SQL Anywhere Server -](#)

[Datenbankadministration](#)] und Zulässige Anweisungen für die Dienstprogrammdatenbank [[SQL Anywhere Server - Datenbankadministration](#)].

Standardmäßig zeigt Interactive SQL die Meldungen zum Verarbeitungsfortschritt im Fenster **Meldungen** an. Sie können die SQL Anywhere-Datenbankoption `progress_messages` auch festlegen, indem Sie nacheinander auf **Extras** → **Optionen** → **SQL Anywhere** → **Befehle** und **Fortschrittsmeldungen anzeigen** klicken. Wenn Sie diese Option ausgewählt haben, wird die Datenbankoption `progress_messages` auf "Formatted" gesetzt. Andernfalls ist sie auf "Off" gesetzt. Standardmäßig ist die Option **Fortschrittsmeldungen anzeigen** aktiviert.

- **timestamp_with_time_zone_format-Option** Mit dieser Option legen Sie fest, wie `TIMESTAMP WITH TIME ZONE`-Werte in Zeichenfolgen umgewandelt werden. Siehe „[timestamp_with_time_zone_format-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **reserved_keywords-Option** Diese Option aktiviert einzelne Schlüsselwörter. Siehe „[reserved_keywords-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_asbinary_format-Option** Steuert, wie räumliche Werte vom Geometrieformat in Binärformat umgewandelt werden. Siehe „[st_geometry_asbinary_format-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_astext_format-Option** Steuert, wie räumliche Werte vom Geometrieformat in Text umgewandelt werden. Siehe „[st_geometry_astext_format-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_asxml_format-Option** Steuert, wie räumliche Werte vom Geometrieformat in XML umgewandelt werden. Siehe „[st_geometry_asxml_format-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_describe_type-Option** Steuert, wie räumliche Werte beschrieben werden. Siehe „[st_geometry_describe_type-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **st_geometry_on_invalid-Option** Steuert das Verhalten, wenn die Basisvalidierung einer Geometrie fehlschlägt. Siehe „[st_geometry_on_invalid-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbankserveroptionen

Die folgende Liste enthält die Erweiterungen der Datenbankserveroptionen in SQL Anywhere Version 12.0.0.

- **-xm-Option** Die Datenbankserveroption `-xm` legt fest, wie oft der Datenbankserver nach neuen IP-Adressen sucht. Wenn bei angegebener Option `-xm` ein Computer mit einem neuen Netzwerk verbunden oder von einem vorhandenen Netzwerk getrennt wird, wartet der Datenbankserver auf Verbindungen mit dem neuen Netzwerk, sobald die Änderung erkannt wurde. Siehe „[Datenbankserveroption -xm](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Eigenschaften und Systemmonitorstatistiken

In der folgenden Liste sind die Erweiterungen von Eigenschaften und Systemmonitorstatistiken in SQL Anywhere Version 12.0.0 aufgeführt.

- **Verbindungseigenschaften** Folgende Verbindungseigenschaften wurden in dieser Version hinzugefügt:

- blocking_others_timeout
- http_connection_pool_basesize
- http_connection_pool_timeout
- ParentConnection
- Progress
- progress_messages
- QueryRowsFetched
- reserved_keywords
- st_geometry_asbinary_format
- st_geometry_astext_format
- st_geometry_asxml_format
- st_geometry_describe_type
- st_geometry_on_invalid
- timestamp_with_time_zone_format

Weitere Hinweise zu diesen Eigenschaften finden Sie unter „[Liste der Verbindungseigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Datenbankeigenschaften** Folgende Datenbankeigenschaften wurden in dieser Version hinzugefügt:

- ConnPoolCachedCount
- ConnPoolHits
- ConnPoolMisses
- DriveBus
- DriveModel
- HasTornWriteFix
- HttpConnPoolCachedCount
- HttpConnPoolHits
- HttpConnPoolMisses
- HttpConnPoolSteals
- LastCheckpointTime
- MirrorRole
- QueryRowsFetched
- SynchronizationSchemaChangeActive
- WriteChecksums

Weitere Hinweise zu diesen Eigenschaften finden Sie unter „[Liste der Datenbankeigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Datenbankservereigenschaften** Folgende Datenbankservereigenschaften wurden in dieser Version hinzugefügt:
 - AutoMultiProgrammingLevel
 - AutoMultiProgrammingLevelStatistics
 - CurrentMultiProgrammingLevel
 - IPAddressMonitorPeriod
 - IsPortableDevice
 - MaxMultiProgrammingLevel
 - MinMultiProgrammingLevel
 - ObjectType
 - ThreadDeadlocksAvoided
 - ThreadDeadlocksReported

Weitere Hinweise zu diesen Eigenschaften finden Sie unter „[Liste der Datenbankservereigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Systemprozeduren und Funktionen

In der folgenden Liste sind die Erweiterungen der Systemprozeduren und Funktionen in SQL Anywhere Version 12.0.0 aufgeführt.

- **Neue sa_text_index_vocab_nchar-Systemprozedur** Diese neue Systemprozedur wird mit NCHAR-Textindizes verwendet und ist gleichwertig mit sa_text_index_vocab. Siehe „sa_text_index_vocab_nchar-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue sa_copy_cursor_to_temp_table-Systemprozedur** Kopiert den Inhalt eines Cursors in eine temporäre Tabelle. Siehe „sa_copy_cursor_to_temp_table-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue sa_describe_cursor-Systemprozedur** Gibt den Namen und die Datentypen von Spalten in einem Cursor zurück. Diese Informationen lassen sich zwar über verschiedene Client-Programmierschnittstellen abfragen, konnten bisher aber nicht innerhalb von gespeicherten Prozeduren abgerufen werden. Siehe „sa_describe_cursor-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue sa_install_feature-Systemprozedur** Installiert zusätzliche Funktionen, die bei der Installation von SQL Anywhere nicht in der Datenbank vorhanden waren. Siehe „sa_install_feature-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue sa_list_cursors-Systemprozedur** Gibt eine Ergebnismenge mit einer Zeile für jeden Cursor zurück, der vom Datenbankserver für diese Verbindung aufrechterhalten wird. Die Ergebnismenge enthält den Cursornamen, einen Wert, der angibt, ob der Cursor derzeit geöffnet ist, und andere Metadaten. Siehe „sa_list_cursors-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue sa_mirror_server_status-Systemprozedur** Diese Prozedur berichtet über den Status von Kopieknoten, die dem Datenbankserver, auf dem die Prozedur ausgeführt wird, untergeordnet sind. Siehe „sa_mirror_server_status-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Neue sa_reserved_words-Systemprozedur** Diese Prozedur gibt eine Liste für SQL Anywhere reservierter Wörter zurück. Siehe „sa_reserved_words-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der sa_server_option-Systemprozedur** Die folgenden Optionen wurden der Systemprozedur sa_server_option hinzugefügt:
 - AutoMultiProgrammingLevel
 - AutoMultiProgrammingLevelStatistics
 - CurrentMultiProgrammingLevel
 - DropBadStatistics
 - DropUnusedStatistics
 - IPAddressMonitorPeriod
 - MaxMultiProgrammingLevel
 - MinMultiProgrammingLevel
 - StatisticsCleaner

Siehe „sa_server_option-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Erweiterungen der sa_table_page_usage-Systemprozedur** Wenn die progress_messages-Option auf "Raw" oder "Formatted" gesetzt ist, werden Meldungen zum Verarbeitungsfortschritt vom Datenbankserver an den Client gesendet, während diese Systemprozedur läuft. Siehe „sa_table_page_usage-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der xp_read_file-Systemprozedur** Die Systemprozedur xp_read_file umfasst nun einen optionalen Parameter, mit dem Sie festlegen können, dass die Inhalte einer Datei nur bei Bedarf gelesen werden. Wenn Sie diesen Parameter mit einem Wert ungleich Null angeben, wird die Datei gelesen und danach sofort entsperrt. Siehe „xp_read_file-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der xp_startsmtp-Systemprozedur** Die xp_startsmtp-Systemprozedur unterstützt vier neue Parameter: trusted_certificates, certificate_company, certificate_unit und certificate_name. Diese Parameter ermöglichen es Ihnen, E-Mails über eine sichere SMTP-Verbindung zu senden. Um diese Funktion zu verwenden, müssen Sie ein Upgrade Ihrer Datenbank ausführen. Siehe „xp_startsmtp-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen des OPENXML-Operators** Der OPENXML-Zeichenfolgenoperator unterstützt nun USING FILE- und USING VALUE-Klauseln, mit denen Sie Daten aus einer Datei oder einem Ausdruck vom Typ CHAR, NCHAR, BINARY oder LONG BINARY bzw. einer BLOB-Zeichenfolge laden können. Siehe „OPENXML-Operator“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue MEDIAN-Funktion** Berechnet den Median eines numerischen Ausdrucks für eine Zeilenmenge. Siehe „MEDIAN-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der HASH-Funktion** Die HASH-Funktion unterstützt nun den Algorithmustyp CRC32. Siehe „HASH-Funktion [Zeichenfolge]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Erweiterungen der Funktionen BIT_OR, BIT_AND und BIT_XOR** Die Funktionen BIT_OR, BIT_AND und BIT_XOR unterstützen nun Ganzzahl- und Binärwerte. Sie lassen sich außerdem in parallelen Ausführungsplänen verwenden.

Siehe:

- „BIT_OR-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „BIT_AND-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „BIT_XOR-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

- **Erweiterungen der Funktionen DATEADD, DATEDIFF, DATEPART und DATENAME** Der Mikrosekunden-Datumsteil und der TIMESTAMP WITH TIME ZONE-Datentyp werden nun von den Funktionen DATEADD, DATEDIFF, DATEPART und DATENAME unterstützt. Siehe:
 - „DATEADD-Funktion [Datum und Uhrzeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DATEDIFF-Funktion [Datum und Uhrzeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DATEPART-Funktion [Datum und Uhrzeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DATENAME-Funktion [Datum und Uhrzeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Erweiterungen der Funktion DB_EXTENDED_PROPERTY** Sie können nun die Funktion DB_EXTENDED_PROPERTY mit den Eigenschaften MirrorServerState und MirrorState verwenden, um den Synchronisations- und Verbindungsstatus eines Spiegelservers zu bestimmen. Siehe „DB_EXTENDED_PROPERTY-Funktion [System]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue HTTP_RESPONSE_HEADER-Funktion** Gibt den Wert eines HTTP-Antwortheaders zurück. Siehe „HTTP_RESPONSE_HEADER-Funktion [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der HTTP_VARIABLE-Funktion** Sie können nun das Attribut @BINARY verwenden, um einen x-www-form-urlencoded-Binärdatenwert zurückzugeben. Siehe „HTTP_VARIABLE-Funktion [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue ISENCRYPTED-Funktion** Legt fest, ob eine Zeichenfolge verschlüsselt ist. Siehe „ISENCRYPTED Funktion [System]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue NEXT_HTTP_RESPONSE_HEADER-Funktion** Ruft den Namen des nächsten HTTP-Response-Headers ab. Siehe „NEXT_HTTP_RESPONSE_HEADER-Funktion [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue SWITCHOFFSET-Funktion** Gibt einen TIMESTAMP WITH TIME ZONE-Wert zurück, der vom ursprünglichen Zeitzone-Offset in den angegebenen Zeitzone-Offset umgewandelt wurde. Siehe „SWITCHOFFSET-Funktion [Datum und Zeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue SYSDATETIMEOFFSET-Funktion** Gibt die aktuellen Werte für Datum, Zeit und Zeitzone-Offset des Datenbankservers zurück. Siehe „SYSDATETIMEOFFSET-Funktion [Datum und Zeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue TODATETIMEOFFSET-Funktion** Verwendet den angegebenen Zeitzone-Offset, um einen TIMESTAMP-Wert in einen TIME STAMP WITH TIME ZONE-Wert umzuwandeln. Siehe

„TODATETIMEOFFSET-Funktion [Datum und Zeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Neue COUNT_BIG-Funktion** Zählt die Anzahl der Zeilen in einer Gruppe, abhängig von den angegebenen Parametern. Siehe „COUNT_BIG-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SQL-Anweisungen

Die folgende Liste enthält alle SQL-Erweiterungen in SQL Anywhere Version 12.0.0.

- **Verbesserte Qualifikation für benannte Index-Hints** Ein PRIMARY KEY-Index und ein FOREIGN KEY-Index für eine Tabelle können denselben Namen haben. Wenn dies zutrifft, kann ein benannter Index-Hint nicht eindeutig festgelegt werden. Der Index mit den Hints lässt sich nun aber als PRIMARY KEY-Index oder FOREIGN KEY-Index qualifizieren. Siehe [INDEX-Klausel](#), [FROM-Klausel](#) [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue IS DISTINCT FROM- und IS NOT DISTINCT FROM-Suchbedingungen** Die Suchbedingungen IS DISTINCT FROM und IS NOT DISTINCT FROM ermöglichen es Ihnen, die Auswertung der Gleichheit im Fall von NULL zu steuern. Siehe „[Suchbedingungen IS DISTINCT FROM und IS NOT DISTINCT FROM](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **CREATE SYNCHRONIZATION PROFILE-Anweisung** Diese Anweisung erstellt ein SQL Anywhere-Synchronisationsprofil. Ein Synchronisationsprofil legt fest, wie eine SQL Anywhere-Datenbank mit dem MobiLink-Server synchronisiert wird. Siehe „[CREATE SYNCHRONIZATION PROFILE-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue WITH NULLS NOT DISTINCT-Klausel, CREATE INDEX-Anweisung** Die CREATE INDEX-Anweisung wurde durch die neue Klausel WITH NULLS NOT DISTINCT ergänzt, die Sie bei der Erstellung von UNIQUE-Indizes verwenden können. Damit legen Sie fest, dass Nullwerte in Indexschlüsseln nicht eindeutig sind. Sie müssen ein Upgrade oder einen Neuaufbau bestehender Datenbanken durchführen, um diese Funktion verwenden zu können. Weitere Hinweise finden Sie in der Beschreibung der UNIQUE-Klausel für die „[CREATE INDEX-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Back Quotes als Begrenzer für Bezeichner** Back Quotes (``) können jetzt als Begrenzer für Bezeichner verwendet werden. Siehe „[Bezeichner](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue SAVE OPTION VALUES-Klausel für ALTER TEXT CONFIGURATION-Anweisung** Verwenden Sie diese neue Klausel, um die Werte der Optionen date_format, time_format, timestamp_format und timestamp_with_time_zone_format zu ändern, die mit einem Textkonfigurationsobjekt gespeichert wurden. Siehe „[ALTER TEXT CONFIGURATION-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Anweisungen ALTER SERVER und CREATE SERVER** Mit den neuen Serverklassen IQODBC und IQJDBC können Sie einen SAP Sybase IQ-Server als entfernte Verbindung angeben. Siehe „[ALTER SERVER-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[CREATE SERVER-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Neue LOCK FEATURE-Anweisung** Diese Anweisung verhindert die gemeinsame Verwendung einer Datenbankserverfunktion durch gleichzeitig ausgeführte Verbindungen. Siehe „[LOCK FEATURE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der Anweisungen BEGIN, CREATE VARIABLE und DECLARE** Variablendeklarationen können nun einen Anfangswert für die Variable enthalten. Siehe:
 - „[BEGIN-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[CREATE VARIABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[DECLARE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Erweiterungen der LOAD TABLE-Anweisung** Die Anweisung LOAD TABLE unterstützt nun die Klausel load-option, mit der Sie steuern können, wie Daten geladen werden. Sie unterstützt auch den XML-Wert in der FORMAT-Klausel. Siehe „[LOAD TABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue IF NOT EXISTS-Klausel** Mit der neuen IF NOT EXISTS-Klausel werden keine Änderungen vorgenommen und keine Fehler zurückgegeben, wenn das benannte Objekt bereits vorhanden ist. Siehe:
 - „[CREATE INDEX-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[CREATE PUBLICATION-Anweisung](#) [MobiLink] [SQL Remote]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[CREATE SPATIAL REFERENCE SYSTEM-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[CREATE TABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[CREATE TEXT INDEX-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

- **Neue IF EXISTS-Klausel** Mit der neuen IF EXISTS-Klausel können Sie vermeiden, dass ein Fehler zurückgegeben wird, wenn die DROP-Anweisung versucht, ein Datenbankobjekt zu entfernen, das nicht existiert. Siehe:
 - „DROP EVENT-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP FUNCTION-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP INDEX-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP MATERIALIZED VIEW-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP PROCEDURE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP PUBLICATION-Anweisung [MobiLink] [SQL Remote]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP SPATIAL REFERENCE SYSTEM-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP SPATIAL UNIT OF MEASURE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP SYNCHRONIZATION PROFILE-Anweisung [MobiLink]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP TABLE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP TRIGGER-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP VARIABLE-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DROP VIEW-Anweisung“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- **Erweiterungen der CASE-Anweisung** CASE-Anweisungen werden nun in Transact-SQL-Prozeduren und Batches unterstützt.

- **Neue OR REPLACE-Klausel** Die neue OR REPLACE-Klausel ermöglicht es Ihnen, ein Profil oder eine Variable mit demselben Namen zu erstellen oder ersetzen. Siehe:
 - „CREATE FUNCTION-Anweisung [externer Aufruf]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE FUNCTION-Anweisung [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE FUNCTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE PROCEDURE-Anweisung [externer Aufruf]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE PROCEDURE-Anweisung [Webdienste]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE PROCEDURE-Anweisung [T-SQL]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE PROCEDURE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE SEQUENCE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE SPATIAL REFERENCE SYSTEM-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE SYNCHRONIZATION PROFILE-Anweisung [MobiLink]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE TRIGGER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE VARIABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Unterstützung der neuen LIMIT-Klausel für SELECT-Anweisungen** Mit der neuen LIMIT-Klausel können Sie die Zeilenanzahl und Offsets in einer SELECT-Anweisung angeben. Siehe „SELECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der SET OPTION-Anweisung** Syntax 1 der SET OPTION-Anweisung unterstützt nun die Einstellung einer Option mit dem Inhalt einer Variablen. Siehe „SET OPTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue IF [NOT] OF-Suchbedingung** Die Suchbedingung IF [NOT] OF *type-expression* wurde hinzugefügt. Siehe „Suchbedingungen“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „Syntax räumlicher Datentypen“ [[SQL Anywhere Server - Unterstützung für räumliche Daten](#)].
- **Erweiterungen der INSERT-Anweisung** Die folgenden Erweiterungen wurden der INSERT-Anweisung hinzugefügt. Siehe „INSERT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Unterstützung für mehr als eine Werteliste** Eine INSERT-Anweisung kann nun mehr als eine Werteliste enthalten, sodass mehrere Zeilen gleichzeitig eingefügt werden können. Beispiel:


```
INSERT INTO T (c1,c2,c3)
VALUES (1,10,100), (2,20,200), (3,30,300);
```
 - **Unterstützung für das Einfügen von Zeilen mit Standardwerten** In SQL Anywhere kann die VALUES-Klausel festgelegte Werte für eine Teilmenge der Tabellenspalten enthalten. Allen un spezifizierten Spalten werden Standardwerte zugeordnet, die für die einzelnen Spalten mit den

Klauseln DEFAULT, NULL und COMPUTE in der CREATE TABLE-Anweisung angegeben wurden. In früheren Versionen mussten Sie Eingabewerte für mindestens eine der Tabellenspalten angeben.

Nun können Sie allen Spalten Standardwerte zuordnen, indem Sie eine der folgenden Syntaxerweiterungen verwenden:

INSERT [INTO] *table-name options* **DEFAULT VALUES** ...

INSERT [INTO] *table-name () options* **VALUES () [, () ...]**

Die Klauseln DEFAULT VALUES oder VALUES sind semantisch gleichwertig mit der folgenden Syntax, wobei die Anzahl der Standardeinträge der Anzahl der Tabellenspalten entspricht:

INSERT [INTO] *table-name* **VALUES(default, default, ..., default)**

Die Klausel DEFAULT VALUES ist Teil des SQL/2008-Standards, während es sich bei der VALUES-Klausel um eine Erweiterung des Herstellers handelt.

- **Neue START SERVER-Anweisung** Die START SERVER-Anweisung wurde hinzugefügt. Sie ersetzt die START ENGINE-Anweisung, die nun nicht mehr empfohlen wird. Siehe „[START SERVER-Anweisung \[Interactive SQL\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue STOP SERVER-Anweisung** Die STOP SERVER-Anweisung wurde hinzugefügt. Sie ersetzt die STOP ENGINE-Anweisung, die nun nicht mehr empfohlen wird. Siehe „[STOP SERVER-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue CHECKSUM-Klausel für die ALTER DATABASE-Anweisung** Mit der CHECKSUM OFF-Klausel können Sie globale Prüfsummen für Datenbanken deaktivieren. Globale Prüfsummen sind standardmäßig für neue Datenbanken der Version 12 aktiviert. Siehe „[ALTER DATABASE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der DELETE-Anweisung** Die DELETE-Anweisung unterstützt nun Korrelationsnamen. Siehe „[DELETE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Datentypen

Die folgende Liste enthält die Erweiterungen der in SQL Anywhere Version 12.0.0 eingeführten Datentypen.

- **TIMESTAMP WITH TIME ZONE-Datentyp** Speichert einen Zeitpunkt mit einem Offset für eine Zeitzone. Siehe „[TIMESTAMP WITH TIME ZONE-Datentyp](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Der Alias DATETIMEOFFSET wird ebenfalls unterstützt. Siehe „[DATETIMEOFFSET-Datentyp](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Die Datentypen CHAR, NCHAR, NVARCHAR und VARCHAR unterstützen eine Zeichenlänge von bis zu 32767.** Die Datentypen NCHAR und NVARCHAR können nun mit bis

zu 32767 Zeichen deklariert werden, genauso wie die Datentypen CHAR und VARCHAR mit Zeichenlängensemantik. Wenn die beschriebene Bytelänge im Clientzeichensatz für die Datentypen NCHAR, NVARCHAR, CHAR oder VARCHAR mit Zeichenlängensemantik den Wert 32767 überschreitet, werden diese Typen als LONG NVARCHAR oder LONG VARCHAR beschrieben. Die beschriebene Bytelänge für die Datentypen CHAR und VARCHAR mit Bytelängensemantik bestimmt die maximale Zeichensatzerweiterung bei der Umwandlung von Werten in den Clientzeichensatz. Wenn die resultierende Länge einschließlich der maximalen Erweiterung den Wert 32767 überschreitet, wird der Typ als LONG VARCHAR beschrieben. Eine Erweiterung ist nicht möglich, wenn der Client einen Einbyte-Zeichensatz verwendet oder wenn die Kodierung des Clientzeichensatzes und die des Datenbankzeichensatzes übereinstimmen. Siehe:

- „CHAR-Datentyp“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „NCHAR-Datentyp“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „NVARCHAR-Datentyp“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „VARCHAR-Datentyp“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Datentypen für räumliche Daten** Mehrere neue Datentypen wurden zur Unterstützung räumlicher Daten hinzugefügt. Siehe „[Zugriff auf räumliche Daten und ihre Verarbeitung](#)“ [[SQL Anywhere Server - Unterstützung für räumliche Daten](#)].

Programmierschnittstellen

Die folgende Liste enthält die Erweiterungen der in SQL Anywhere Version 12.0.0 eingeführten Programmierschnittstellen.

- **Verbesserte Performance von Webdiensten** HTTP-Dienste unterstützen nun automatisches Verbindungspooling, um die Wirkung des Plan-Cachings zu maximieren und die Performance zu steigern. Siehe „[Webdienstanwendungen auf einem HTTP-Webserver entwickeln](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Unterstützung für benutzerdefinierte ODBC-Treibernamen** Um die Installation und Registrierung mehrerer unabhängiger Kopien des ODBC-Treibers von SQL Anywhere auf einem Client zu vereinfachen, können Sie dem ODBC-Treiber nun einen benutzerdefinierten Namen zuweisen. Siehe „[ODBC-Treiberkonfiguration](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Reiner ANSI-ODBC-Treiber für Unix** Die Versionen von ODBC-Treibermanagern für Unix, die SQLWCHAR als 32-Bit-Mengen (UTF-32) definieren, können nicht mit dem SQL Anywhere-ODBC-Treiber verwendet werden, der breite Aufrufe unterstützt, da dieser Treiber für 16-Bit-SQLWCHAR erstellt wurde. Für solche Fälle wird eine reine ANSI-Version des SQL Anywhere-ODBC-Treibers zur Verfügung gestellt. Diese Version des ODBC-Treibers unterstützt keine Schnittstellen für breite Aufrufe (z.B. SQLConnectW). Siehe „[UTF-32-ODBC-Treibermanager für Unix](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **DBTools: Das no_reload_status-Bitfeld wurde der an_unload_db-Struktur hinzugefügt** Die Struktur an_unload_db umfasst nun das neue Bitfeld no_reload_status. Sie können no_reload_status verwenden, um Fortschrittmeldungen für Tabellen und Indizes beim Neuladen zu unterdrücken. Siehe [an_unload_db-Struktur](#) [Datenbanktools] [[SQL Anywhere Server - Programmierung](#)].

- **Neuer SQL Anywhere TYPE-2 JDBC-Treiber** JDBC-Anwendungen steht bei der Verbindung mit SQL Anywhere ein neuer TYPE-2 JDBC-Treiber zur Verfügung. Anders als der iAnywhere JDBC-Treiber, der auf ODBC aufgesetzt wird und für die Verbindung mit verschiedenen Servern über ODBC verwendet werden kann, wird der JDBC-Treiber ausschließlich mit SQL Anywhere verbunden. Er erfordert weder die Installation noch die Registrierung des SQL Anywhere ODBC-Treibers.

Der SQL Anywhere JDBC-Treiber wird mit einem JDBC 3.0-Treiber und einem JDBC 4.0-Treiber ausgeliefert.

Hinweis

Anwenden des iAnywhere JDBC-Treibers wird empfohlen, auf den neuen SQL Anywhere JDBC-Treiber umzusteigen.

Der JDBC 4.0 Treiber lädt und registriert sich automatisch selbst.

Um Version 3.0 des SQL Anywhere JDBC-Treibers verwenden zu können, müssen Sie die Klasse `sybase.jdbc.sqlanywhere.IDriver` laden, die die `java.sql.Driver`-Schnittstelle implementiert und den SQL Anywhere JDBC-Treiber beim JDBC-Treibermanager registriert. Anschließend können Verbindungen mit dem SQL Anywhere JDBC-Treiber über die folgende URL hergestellt werden: `jdbc:sqlanywhere:connection-string-parameters`. Die *connection-string-parameters* sind Standardverbindungsparameter, die für die Verbindung mit SQL Anywhere erforderlich sind. Beachten Sie, dass bei Verwendung des SQL Anywhere JDBC-Treibers die Angaben `DRIVER=` oder `DSN=` für die *connection-string-parameters* nicht mehr erforderlich sind. Siehe „JDBC-Unterstützung“ [[SQL Anywhere Server - Programmierung](#)].

- **Neue Unterstützung für Klassenmethoden von JDBC-Anweisungen** In früheren Versionen unterstützten JDBC-Treiber nur die `addBatch`- und `executeBatch`-Methoden der `PreparedStatement`-Klasse. Die JDBC-Treiber unterstützen nun außerdem die Methoden `addBatch`, `clearBatch` und `executeBatch` der `Statement`-Klasse. Da die JDBC-Spezifikation im Hinblick auf das Verhalten der `executeBatch`-Methode der Klasse `Statement` unklar ist, sollten die folgenden Hinweise in Betracht gezogen werden, wenn diese Methode mit dem SQL Anywhere-JDBC-Treiber verwendet wird:
 1. Die Verarbeitung des Batches wird sofort angehalten, wenn eine SQL-Ausnahmebedingung oder eine Ergebnismenge erkannt wird. Beim Anhalten der Batchverarbeitung wird von der `executeBatch`-Methode ein `BatchUpdateException`-Objekt ausgegeben. Mit dem Aufruf der `getUpdateCounts`-Methode für das `BatchUpdateException`-Objekt wird ein Ganzzahl-Array von Zeilenanzahlangaben zurückgegeben, in dem die Menge der Anzahlangaben vor dem Batchfehler eine gültige nicht negative Aktualisierungsanzahl enthält, während alle Anzahlangaben ab dem Batchfehler den Wert -1 enthalten. Das Casting des `BatchUpdateCount`-Objekts in ein `SQLException`-Objekt liefert zusätzliche Einzelheiten dazu, warum die Batchverarbeitung angehalten wurde.
 2. Der Batchinhalt wird nur gelöscht, wenn die `clearBatch`-Methode explizit aufgerufen wurde. Daher wird bei wiederholtem Aufrufen der `executeBatch`-Methode der Batch immer wieder neu ausgeführt. Außerdem wird beim Aufrufen von `execute(sql_query)` oder `executeQuery(sql_query)` zwar die angegebene SQL-Abfrage korrekt ausgeführt, aber der Batchinhalt wird nicht gelöscht. Wenn Sie also die `executeBatch`-Methode aufrufen, gefolgt von `execute(sql_query)` und gefolgt von einem erneuten Aufruf der `executeBatch`-Methode, werden

die im Batch zusammengefassten Anweisungen ausgeführt, anschließend wird die angegebene SQL-Abfrage ausgeführt und schließlich werden die im Batch zusammengefassten Anweisungen erneut ausgeführt.

Siehe „JDBC-Unterstützung“ [[SQL Anywhere Server - Programmierung](#)].

- **Die RESUME-Anweisung gibt die Zeilenanzahl oder den geschätzten Wert für die Zeilenanzahl zurück.** Die RESUME-Anweisung gibt die Zeilenanzahl in der Ergebnismenge oder die geschätzte Zeilenanzahl für die nächste Ergebnismenge in einem Prozeduraufruf zurück. In früheren Versionen war dieser Vorgang mit der RESUME-Anweisung nicht möglich. Um die Zeilenanzahl mit RESUME zu ermitteln, müssen sowohl die Clientanwendung als auch der Datenbankserver SQL Anywhere 12 verwenden.

Diese Änderung hat Auswirkungen auf die folgenden Client-APIs:

API	Funktionsaufruf oder Anweisung	Rückgabe der Zeilenanzahl durch
Embedded SQL	RESUME-Anweisung	SQLCOUNT-Feld
ODBC	SQLMoreResults-Funktion	SQLRowCount-Funktion
OLE DB	IMultipleResults::GetResult-Methode	pcRowsAffected-Ausgabeparameter
PHP	sasql_next_result-Funktion	sasql_num_rows-Funktion
PHP	sasql_stmt_next_result-Funktion	sasql_stmt_num_rows-Funktion
SQL Anywhere C-API	sqlany_get_next_result-Funktion	sqlany_num_rows-Funktion
SQL Anywhere Ruby-API	sqlany_get_next_result-Funktion	sqlany_num_rows-Funktion
SQL Anywhere Python-API	nextset-Methode	rowcount-Attribut

- **PHP \$_SERVER-Variable enthält jetzt alle von CGI/1.1 benötigten Variablen** Die \$_SERVER-Variable in PHP enthält nun alle Variablen, die von CGI/1.1 bei der Ausführung einer HTTP-Anforderung benötigt werden. Siehe „Die externe PHP-Umgebung“ [[SQL Anywhere Server - Programmierung](#)].

Weitere Hinweise zu den Variablen, die von CGI/1.1 benötigt werden, finden Sie im Abschnitt 4 von RFC3875: <http://www.ietf.org/rfc/rfc3875>

- **Genaue Kontrolle der Python-Datentypzuordnung wurde in sqlanydb hinzugefügt** Um zu steuern, wie Datenbanktypen beim Abrufen von Ergebnissen vom Datenbankserver in Python-Objekte zugeordnet werden, können Sie Konvertierungs-Callbacks registrieren. Siehe „Datenbanktypkonvertierung“ [[SQL Anywhere Server - Programmierung](#)].

- **Neue DBLib fill_sqlda_ex-Funktion** Die fill_sqlda_ex-Funktion bietet über fill_sqlda eine zusätzliche Funktionalität. Insbesondere kann ein Parameter bereitgestellt werden, um die DT_LONGVARCHAR-, DT_LONGNVARCHAR- und DT_LONGBINARY-Datentypen beim Füllen des Deskriptors zu bewahren. Siehe „fill_sqlda_ex-Funktion“ [[SQL Anywhere Server - Programmierung](#)].
- **Auswahl der dialogfreien Installation ist nun möglich** Unter Windows können SQL Anywhere-Komponenten oder -Funktionen für die Installation von der Befehlszeile ausgewählt oder weggelassen werden. Geben Sie z.B. AT32=1 für die Auswahl der 32-Bit-Administrationstools an. Siehe „Dialogfreie Installationen mit dem SQL Anywhere-Installationsprogramm“ [[SQL Anywhere Server - Programmierung](#)].
- **TDS unterstützt nun die Gesamtstellenzahl von 6 Ziffern für time- und datetime-Daten** Anwendungen, die unter Verwendung von jConnect 7 und höher oder Open Client 15.5 und höher eine Verbindung zu SQL Anywhere herstellen, können nun bei der Abfrage von time- oder datetime-Daten eine Gesamtstellenzahl von 6 Ziffern abrufen.

Katalogänderungen

Die folgende Liste enthält alle Katalogänderungen in SQL Anywhere Version 12.0.0.

- **Neue ISYSSEQUENCE-Systemtabelle und SYSSEQUENCE-Systemansicht** Die ISYSSEQUENCE-Systemtabelle enthält eine Zeile für jede benutzerdefinierte Sequenz. Siehe „SYSSEQUENCE-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue ISYSSEQUENCEPERM-Systemtabelle und SYSSEQUENCEPERM-Systemansicht** In der ISYSSEQUENCEPERM-Systemtabelle werden die Privilegien erfasst, die Benutzer oder Gruppen für Sequenzen haben. Siehe „SYSSEQUENCEPERM-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **ISYSTEXTCONFIG-Systemtabelle** Die ISYSTEXTCONFIG-Systemtabelle speichert nun Informationen über die Eintrittspunkte und die externen Bibliotheken für Token und/oder Vorfilter. Der vorhandene Spaltentyp für Vorfilter wurde in LONG VARCHAR geändert, um die Eintrittspunkte und den Namen für eine externe Vorfilterbibliothek aufnehmen zu können. Darüber hinaus wurde die neue LONG VARCHAR-Spalte external_term_breaker hinzugefügt, damit die Eintrittspunkte und der Name für eine externe Begriffsegmentierer-Bibliothek berücksichtigt werden können. Siehe „ISYSTEXTCONFIG-Systemansicht“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **ISYSTABCOL-Systemtabelle: neue base_type_str-Spalte** Diese Spalte enthält die zugehörige Typzeichenfolge, die den physischen Spaltentyp darstellt.
- **ISYSPROCARM-Systemtabelle: neue base_type_str-Spalte** Diese Spalte enthält die zugehörige Typzeichenfolge, die den physischen Parametertyp darstellt.
- **ISYSUSERTYPE-Systemtabelle: neue base_type_str-Spalte** Diese Spalte enthält die zugehörige Typzeichenfolge, die den physischen Typ des Benutzertyps darstellt.

- **ISYSOBJECT-Systemtabelle: neue object_type_str-Spalte** Diese Spalte enthält eine Textbeschreibung (zum Beispiel MAT VIEW) des Werts in SYSOBJECT.object_type.

In früheren Versionen enthielt die SYSOBJECT-Ansicht nur object_type, womit der Objekttyp als TINYINT-Datentyp ausgedrückt wurde. Das bedeutet, Sie mussten entweder auf die Dokumentation oder die Ansichtsdefinition zugreifen, um die Ganzzahl in eine Textbeschreibung zu übersetzen. Nun können Sie die Spalte object_type_str abfragen, um die Textbeschreibung des Objekts anzuzeigen.

Erweiterungen für Windows Mobile

Die folgende Liste enthält alle Windows Mobile-Erweiterungen in SQL Anywhere Version 12.0.0.

- **Änderung der standardmäßigen Stack-Größe für interne Ausführungsthreads in Windows Mobile** Die standardmäßige Stack-Größe in Windows Mobile ist nun 96 KByte, die minimale Stack-Größe 64 KByte und die maximale Stack-Größe 512 KByte. Siehe „Datenbankserveroption - gss“ [[SQL Anywhere Server - Datenbankadministration](#)].

Unix- bzw. Linux-Erweiterungen

Die folgende Liste enthält die Erweiterungen für Unix und Linux in SQL Anywhere Version 12.0.0.

- **Dienstprogramm für Performancestatistik (dbstats)** Das Dienstprogramm dbstats gibt den Wert von Performancestatistiken auf Unix-Computern zurück. Es verhält sich ähnlich wie der Windows-Systemmonitor. Siehe „Dienstprogramm für Performancestatistik (dbstats) (Unix)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **PID-Datei für Linux-Dienste erstellt** Wenn ein SQL Anywhere-Dienst unter Linux läuft, wird im Verzeichnis /var/run eine PID-Datei erstellt. Siehe „Dienstprogramm für Dienste (dbsvc) für Linux“ [[SQL Anywhere Server - Datenbankadministration](#)].

Performanceverbesserungen

Die folgende Liste enthält die Performanceverbesserungen in Version 12.0.0, die nicht mit sichtbaren Änderungen der Benutzeroberfläche einhergehen.

- **Verbessertes Sperren bei der Aktualisierung von Primärzeilen** Die Vorgänge zur Aktualisierung einer Spalte ohne Schlüssel in einer Primärzeile und die Bearbeitung von Fremdzeilen, die auf diese Zeile verweisen, überlagern einander nicht mehr. Zum Beispiel können Sie in der Beispieldatenbank eine sales_order-Zeile hinzufügen, während die entsprechende Kundenadresse aktualisiert wird, ohne dass einer der beiden Vorgänge angehalten werden muss. Siehe „sa_locks-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „Sperrentypen“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Weniger Sperren auf den Isolationsstufen 2 und 3** Wenn eine Tabelle im gemeinsamen Modus gesperrt wird (LOCK TABLE...IN SHARE MODE-Anweisung), werden ab sofort keine Lesesperren mehr für einzelne Zeilen eingerichtet. Dadurch lässt sich der Sperren-Overhead auf den Isolationsstufen 2 und 3 reduzieren.

- **Verbesserte Indexperformance** SQL Anywhere 12 umfasst verbesserte Algorithmen und eine Methode zur Datenspeicherung, mit der die Performance beim Löschen großer Mengen an geclusterten sequenziellen Werten aus einem Index gesteigert wird.
- **Verbesserte Performance bei der Validierung** SQL Anywhere 12 bietet zahlreiche Verbesserungen für die Validierung großer Datenbanken.
- **Verbesserte Priorisierung von Anforderungen** SQL Anywhere 12 verbessert die Priorisierung von I/O-Anforderungen, sodass Hardwareressourcen effizienter eingesetzt werden können.
- **Verbesserte Performance bei Ferndatenzugriff** SQL Anywhere 12 umfasst eine Reihe von Erweiterungen zur Verbesserung der Ferndatenzugriff-Performance sowie zur verbesserten Proxytabellen-Performance.
- **Neues Kostenmodell** SQL Anywhere 12 umfasst ein CPU-Kostenmodell, mit dem die Abfrageausführungskosten für moderne Hardware präziser eingeschätzt werden können. Dieses Verhalten kann zu Änderungen in Zugriffsplänen für einige Abfragen führen.
- **Erweiterung von eingebetteten Abfragen in benutzerdefinierten Funktionen** In benutzerdefinierten Funktionen eingebettete SQL-Abfragen können nun vom Abfrageoptimierer gespeichert werden, wodurch ein Prozedurkontextwechsel bei jedem Aufruf vermieden wird und der Optimierer mehr Freiheiten bei der Optimierung der Anweisung hat.
- **Verbesserungen bei der Konvertierung von Ausdrücken in andere Datentypen** Die vom Datenbankserver bei der Konvertierung eines Ausdrucks zu einem anderen Datentyp verwendeten Auswertungsregeln wurden verbessert. Mit den neuen Auswertungsregeln wird die Konvertierung effizienter ausgeführt.

Verschiedenes

Die folgende Liste enthält verschiedene Erweiterungen in SQL Anywhere Version 12.0.0.

- **Neues Skript zum Erstellen von Kopien der Beispieldatenbank** Die Dateien *newdemo.bat* und *newdemo.sh* befinden sich im Verzeichnis *bin32* oder *bin64* des SQL Anywhere-Installationsverzeichnisses. Sie können verwendet werden, um eine Kopie der Beispieldatenbank zu erstellen, die alle Daten aus der Beispieldatenbank enthält. Dieses Skript kann verwendet werden, um die Beispieldatenbank neu zu erstellen oder neue Kopien davon unter einem anderen Namen zu erstellen. Siehe „[Neuerstellung der Beispieldatenbank \(demo.db\)](#)“ [*SQL Anywhere 16 - Einführung*].
- **Neuer Quellentyp für die Selektivitätsschätzung für den Abfrageoptimierer** Der neue Quellentyp der Selektivitätsschätzung JOIN wurde hinzugefügt. Dieser neue Quellentyp wird vom Abfrageoptimierer für Selektivitätsschätzungen von atomaren Prädikaten in der Form $T.X = R.X$ verwendet. Siehe „[ESTIMATE_SOURCE-Funktion \[Verschiedene\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Bessere Behandlung von Überlauffehlern** Arithmetische Operationen (+, -, *, /, SUM, AVG) können einen Überlauf verursachen, weil sich das Ergebnis der Operation nicht im Datentyp darstellen lässt. In früheren Versionen wurde in diesem Fall für Ausdrücke vom Typ INT ein Fehler zurückgegeben, für alle anderen Datentypen ein undefinierter Wert. Nun werden bei allen

arithmetischen Operationen mit allen Datentypen eventuelle Überläufe erkannt. Wenn das Ergebnis nicht im Datentyp ausgedrückt werden kann, wird ein Fehler zurückgegeben.

- **Verwendung der ALTER EXTERNAL ENVIRONMENT-Anweisung zur Angabe des Speicherorts von dbmlsync** Wenn während der Synchronisation die Programmdatei dbmlsync mithilfe der Umgebungsvariable PATH, die der Datenbankserver verwendet, nicht gefunden werden kann, können Sie den Speicherort von dbmlsync anhand der ALTER EXTERNAL ENVIRONMENT-Anweisung angeben. Siehe „ALTER EXTERNAL ENVIRONMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Unterstützung von Back Quotes als Begrenzer** Sie können nun Bezeichner in SQL Anywhere mit Back Quotes (‘) begrenzen. Siehe „Bezeichner“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Anpassung der UCA-Kollation (Unicode Collation Algorithm) für Japanisch** Eine neue Option zur UCA-Kollationsanpassung für Japanisch ist nun verfügbar. Damit können Sie einen Unterschied auf erster Ebene zwischen allen Hiragana- und Katakana-Buchstaben festlegen. Die neue Option ermöglicht korrekte Gleichheitsvergleiche von Hiragana- und Katakana-Buchstaben in Kollationen, in denen die Groß-/Kleinschreibung nicht berücksichtigt wird. Siehe „Optionen der Kollationsanpassung“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Änderungen des Servermeldungen-Fensters und Taskleistensymbols in Windows** Die Titelleiste des Fensters für Datenbankservermeldungen gibt nun an, ob Sie einen Personal Server oder einen Netzwerkserver ausführen. Die QuickInfo für das Taskleistensymbol in Windows zeigt den Typ des Datenbankservers an. Das Datenbankserver-Fenster **Info** weist außerdem auf die ausgeführte SQL Anywhere-Version hin.
- **Erweiterung des Informationsdienstprogramms (dbinfo)** Das Dienstprogramm dbinfo gibt nun Informationen zur CHAR-Kollationsspezifikation, CHAR-Kodierung, NCHAR-Kollationsspezifikation oder NCHAR-Kodierung für eine Datenbank zurück. Siehe „Informations-Dienstprogramm (dbinfo)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Den Umfang des Adressraums für die Verwendung außerhalb des Caches steuern** Die Option -ch stellt nun mehr Adressraum für die Verwendung außerhalb des Cachespeichers bereit, und die maximale Nicht-AWE-Cachegröße wurde auf 32-Bit-Betriebssystemen verringert. Siehe „Datenbankserveroption -ch“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Datenbankserveroption -chx“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verbesserte Prioritätenverwaltung für I/O-intensive Vorgänge im Vergleich zu CPU-intensiven Anforderungen** Der Datenbankserver erkennt nun dynamisch Anforderungen, die I/O-intensiv sind, und erhöht ihre Prioritäten gegenüber CPU-intensiven Tasks, um den Festplattendurchsatz und die Verwendung von Hardwareressourcen zu steigern.
- **Verbesserte Robustheit durch Failover unter Windows** Sie können einen Windows-Registrierungseintrag erstellen, der die Robustheit von Systemen, die bestimmte Intel-Speichertreiber verwenden, beim Deployment von SQL Anywhere bei Stromausfällen verbessert. Wenn dieser Parameter nicht festgelegt wird, kann Datenverlust eintreten und bei Stromausfall können Datenbanken beschädigt werden. Hinweise dazu, ob diese Einträge als Teil des Deployments erforderlich sind, finden Sie unter „Robustheit bei Intel-Speichertreibern verbessern“ [[SQL Anywhere Server - Programmierung](#)].

- **SQL Flagger-Erweiterung** Der SQL Flagger unterstützt nun den SQL/2008-Standard. Siehe „Testen der SQL-Konformität mit dem SQL Flagger“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Meldungen zum Verarbeitungsfortschritt** Einige SQL-Anweisungen unterstützen nun Meldungen zum Verarbeitungsfortschritt von Sendevorgängen vom Datenbankserver an den Client. Siehe [Neue progress_messages-Option auf Seite 130](#).

Verhaltensänderungen von SQL Anywhere

Nachstehend finden Sie eine Liste der Verhaltensänderungen von SQL Anywhere, die in Version 12.0.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Prüfsummen sind standardmäßig für neue Datenbanken aktiviert** Wenn Sie eine neue Datenbank erstellen, werden globale Prüfsummen nun standardmäßig aktiviert. Globale Prüfsummen werden jedes Mal berechnet und überprüft, wenn eine Datenbankseite gelesen oder geschrieben wird. Sie werden verwendet, um zu bestimmen, ob eine Datenbankseite auf der Festplatte verändert wurde. Sie können mit der Option -s[+ | -] des Initialisierungs-Dienstprogramms oder mit der CHECKSUM-Klausel der CREATE DATABASE-Anweisung kontrollieren, ob globale Prüfsummen für eine Datenbank aktiviert wurden. Siehe:
 - [Prüfsummenerweiterungen auf Seite 127](#)
 - „Dienstprogramm Initialisierung (dbinit)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „CREATE DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „START DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Änderungen des Checkpoint-Logs** In früheren Versionen hat SQL Anywhere das Checkpoint-Log vollständig gekürzt, wenn Sie eine Datenbank gestoppt haben. In Version 12 wird eine Verlaufsliste der Checkpoint-Log-Verwendung in der Datenbank verwaltet. Sie wird dazu verwendet, um eine passende Größe für das Checkpoint-Log für die nächste Sitzung zu ermitteln.

Durch die Verwaltung des Checkpoint-Logs über Sitzungen hinweg wird der Overhead für die Zuweisung beim jeweils nächsten Start der Datenbank ebenso vermieden wie die Dateifragmentierung, die auftreten kann, wenn Dateien erweitert werden. Datenbankdateien sind nun größer als in früheren Versionen, wenn die Datenbank heruntergefahren wird, aber der zusätzliche Speicherplatz wird für das Checkpoint-Log wieder verwendet, wenn die Datenbank das nächste Mal neu gestartet wird. Siehe „Checkpoint-Logs“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Netzwerk-Datenbankserver weist nun die maximale Anzahl von Worker-Threads zu** In früheren Versionen wurde beim Start des Netzwerk-Datenbankservers die Anzahl von Worker-Threads zugewiesen, die der Multiprogramming-Stufe des Datenbankservers entsprechen. In Netzwerkservern der Version 12 entspricht die Anzahl von zugewiesenen Worker-Threads der maximalen Multiprogramming-Stufe des Servers. Diese Zunahme der Anzahl von Worker-Threads erhöht die Anforderungen an den Adressraum des Netzwerk-Datenbankservers für Worker Stacks. Außerdem kann sie Auswirkungen auf den Umfang des Datenbankcache haben, den der Datenbankserver zuweisen kann.

Auf 32-Bit-Windows-Plattformen erfordert jeder Worker Thread z.B. 1 MByte Adressraum für seinen Stack. Eine Netzwerkserver der Version 11, der unter Windows startet, würde 20 MByte Adressraum für Worker Stacks erfordern, da seine standardmäßige Multiprogramming-Stufe 20 ist. Ein Netzwerkserver der Version 12, der unter Windows startet, erfordert 80 MByte Adressraum, da die standardmäßige maximale Anzahl von Worker-Threads 80 ist. Diese Änderung hat keine Auswirkungen auf Personal Server oder Windows Mobile. Siehe „[SQL Anywhere-Threading](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Alte Statistiken werden nicht geladen, wenn eine Datenbank neu aufgebaut wird** Beim Neuaufbau einer Datenbank der Version 11 oder früher überspringt die LOAD STATISTICS-Anweisung stillschweigend das Laden von alten Zeichenfolgen-Statistikdaten in die neue Datenbank, führt aber ein Upgrade der Version der Zeichenfolgenstatistiken durch. Siehe „[LOAD STATISTICS-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Beim Upgrade einer Datenbank (mit dem Upgrade-Dienstprogramm) wird kein Upgrade der Version der Zeichenfolgenstatistiken vorgenommen.

- **Positionierte DELETE- und UPDATE-Anweisungen** In früheren Versionen konnten Sie eine TOP- oder FIRST-Klausel für eine positionierte UPDATE- oder DELETE-Anweisung angeben. Die Klauseln wurden jedoch ignoriert. Jetzt gibt die Angabe von TOP oder FIRST in einer positionierten UPDATE- oder DELETE-Anweisung einen Syntaxfehler zurück. Siehe „[UPDATE-Anweisung \(positionsbasiert\)](#) [ESQL] [SP]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[DELETE-Anweisung \(positionsbasiert\)](#) [ESQL] [SP]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **JDBC-Treiber geben jetzt CHAR und VARCHAR anstelle von nur CHAR für beiden Typen zurück** Wenn eine Anwendung in früheren Versionen eine Verbindung unter Verwendung des iAnywhere JDBC-Treibers hergestellt hat und versucht hat, die Metadaten einer Tabelle oder Ergebnismenge mit einer CHAR-Spalte zu beschreiben, gaben die Metadaten als Spaltentyp CHAR zurück. Der SQL-Typ wurde jedoch als Types.VARCHAR zurückgegeben. Wenn in einer JDBC-Anwendung der JDBC-Treiber den SQL-Typ von CHAR-Spalten als Types.CHAR zurückgeben sollte, musste die Anwendung die `odbc_distinguish_char_and_varchar`-Datenbankoption festlegen. Nun geben der neue SQL Anywhere JDBC-Treiber und der nicht mehr empfohlene iAnywhere JDBC-Treiber den Typnamen CHAR und den SQL-Typ Types.CHAR für Tabellen- und Ergebnismengenspalten des Typs CHAR zurück sowie den Typnamen VARCHAR und den SQL-Typ Types.VARCHAR für Spalten vom Typ VARCHAR, und zwar unabhängig von der Einstellung der Datenbankoption.
- **Änderungen von CURRENT UTC TIMESTAMP- und UTC TIMESTAMP-Spezialwerte** Der zu Grunde liegende Datentyp für den CURRENT UTC TIMESTAMP-Spezialwert und der Standardwert für den UTC TIMESTAMP-Spezialwert ist nun TIMESTAMP WITH TIME ZONE. Wenn diese Werte für Spalten verwendet werden, die als TIMESTAMP definiert sind, wird der Zeitzone-Offset gekürzt und es sollte kein Unterschied im Verhalten erkennbar sein. Wenn diese Werte jedoch für CHAR- oder VARCHAR-Spalten verwendet werden, führt der Offset dazu, dass andere Werte als zuvor generiert werden. Siehe „[CURRENT UTC TIMESTAMP-Spezialwert](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[UTC TIMESTAMP-Spezialwert](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Personal Server startet TCP/IP nicht mehr standardmäßig** Der Personal Datenbankserver startet nur noch das Shared Memory-Protokoll standardmäßig. Wenn Sie das TCP/IP-Protokoll

verwenden möchten, müssen Sie dies beim Start des Personal Datenbankservers mithilfe der Serveroption -x angeben. Siehe „[Datenbankserveroption -x](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[Hinweise zum Kommunikationsprotokoll](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **TCP/IP-Verbindungen** Wenn Sie eine Verbindung über TCP/IP herstellen, ist der Datenbankservername (durch den Verbindungsparameter ServerName (SERVER) festgelegt) nicht mehr erforderlich, sofern mit dem Verbindungsparameter HOST ein Hostname angegeben wird. Siehe „[Host-Verbindungsparameter](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Host-Protokolloption (IP)** In früheren Versionen gab die Host-Protokolloption nicht nur einen oder mehrere Hosts an, auf dem bzw. auf denen der Datenbankserver lief, sondern sie diente auch als Hint für die Clientbibliothek. Konnte der Datenbankserver auf den angegebenen Hosts nicht gefunden werden, wurde ein Netzwerk-Broadcast zur Suche des Datenbankservers durchgeführt. Wenn Sie nun die Host-Option angeben, werden nur die festgelegten Hosts nach dem Datenbankserver durchsucht, und der Client führt standardmäßig keinen Broadcast durch.

Dieses Verhalten ist vergleichbar mit der Einstellung 'Direkt' für die DoBroadcast-Protokolloption. Wenn der Datenbankserver auf einem anderen Computer als von der HOST-Protokolloption angegeben läuft, wird er nicht gefunden. Wenn Sie das Verhalten aus früheren Versionen vorziehen, legen Sie in der Verbindungszeichenfolge DoBroadcast=All fest. Siehe „[Host-Protokolloption \(IP\) \(nur clientseitig\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[DoBroadcast-Protokolloption \(DOBROAD\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **ServerPort-Protokolloption (PORT)** In früheren Versionen legte die PORT-Protokolloption nicht nur eine oder mehrere Nummern der Ports fest, an denen der Datenbankserver auf eine Anfrage warten sollte, sondern sie diente auch als Hint für die Clientbibliothek. Beim Senden eines Broadcasts durch die Clientbibliothek wurden die mit der PORT-Protokolloption angegebenen Portnummern verwendet, sowie die Standardportnummer 2638. Wenn Sie nun die PORT-Option angeben, werden nur die festgelegten Ports bei der Suche nach dem Datenbankserver verwendet. Siehe „[ServerPort-Protokolloption \(PORT\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung des Idle-Verbindungsparameters für Shared Memory** Der Idle-Verbindungsparameter legt das Inaktivitäts-Timeout für eine Verbindung fest. Dieser Parameter wird nun bei Verbindungen mit Shared Memory berücksichtigt. Der Standardwert für das Inaktivitäts-Timeout bei Shared Memory-Verbindungen ist Null (kein Inaktivitäts-Timeout). Siehe „[Verbindungsparameter Idle](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Der Datenbankservername kann sich vom Namen im Verbindungsparameter ServerName (Server) unterscheiden** In früheren Versionen stimmte der Wert der Datenbankeigenschaft Name immer mit dem des Verbindungsparameters ServerName (Server) überein. Wenn ein Client eine Verbindung mit einer Datenbank herstellt, den neuen Verbindungsparameter NodeType (NODE) verwendet und zu einem anderen Datenbankserver umgeleitet wird, stimmen die Namen jedoch nicht überein. Siehe „[Verbindungsparameter NodeType \(NODE\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Manche Vorgänge werden bei der Verbindung mit alternativem Servernamen nicht unterstützt** Wenn in früheren Versionen ein Client eine Datenbankverbindung unter Verwendung eines alternativen Servernamens herstellte, konnten andere Datenbanken auf demselben

Datenbankserver erstellt, gestoppt und gelöscht werden. Diese Vorgänge werden nun nicht mehr unterstützt, wenn Sie eine Verbindung mithilfe eines alternativen Servernamens herstellen.

- **Verhaltensänderungen der Datenbankspiegelung und nicht mehr empfohlene Funktionen** Bei der Serveroption -xp zum Festlegen von Datenbankspiegelungsoptionen werden Optionen wie der Name des Arbiterservers, die Authentifizierungszeichenfolge und der Synchronisationsmodus nun nicht mehr empfohlen. Wenn Sie den Datenbankserver in einem Spiegelungssystem verwenden möchten, müssen jedoch nach wie vor **-xp on** angeben. Siehe „Datenbankoption -xp“ [[SQL Anywhere Server - Datenbankadministration](#)].

Sie können die Datenbankspiegelungsoptionen nun mithilfe der folgenden SQL-Anweisungen festlegen:

- „CREATE MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SET MIRROR OPTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Die neue Version umfasst die folgenden Verhaltensänderungen bei Datenbankspiegelungen:

- In früheren Versionen wurde die Statusinformationsdatei für einen Spiegelserver standardmäßig nach dem Servernamen benannt. Nun müssen Sie den Namen der Statusinformationsdatei explizit angeben.
- In früheren Versionen wurden Webdienste-Anforderungen, die an den Spiegelserver gerichtet waren, zum Primärserver umgeleitet. In dieser Version werden diese Anforderungen direkt von dem Server verarbeitet, an den sich gerichtet sind.

Weitere Hinweise zu den Erweiterungen der Datenbankspiegelung in dieser Version finden Sie unter [Erweiterungen der Datenbankspiegelung auf Seite 123](#).

- **Verhaltensänderung des Embedded SQL-Cursors** Embedded SQL-Cursor werden nun standardmäßig auf READ ONLY gesetzt. Explizite FOR READ ONLY- oder FOR UPDATE-Klauseln müssen nun in der PREPARE-Anweisung anstatt in der DECLARE-Anweisung angegeben werden. Siehe:
 - „DECLARE CURSOR-Anweisung [ESQL] [SP]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „PREPARE-Anweisung [ESQL]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Geschlossene Cursor für CREATE OR REPLACE PROCEDURE-Anweisungen** Wenn Sie eine CREATE OR REPLACE PROCEDURE- Anweisung ausführen, werden alle für eine Verbindung geöffneten Cursor geschlossen. Siehe „CREATE PROCEDURE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Back Quotes als Begrenzer für Bezeichner** Back Quotes (``) können jetzt als Begrenzer für Bezeichner verwendet werden. Siehe „Bezeichner“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Verändertes Sperrverhalten** Um die Parallelität zu optimieren, können nun die Zeilenbereiche mit und ohne Schlüssel separat gesperrt werden. Nicht-Schlüsselspalten einer Zeile lassen sich aktualisieren, ohne dass das Einfügen und Löschen von Fremdzeilen, die auf diese Zeile verweisen,

beeinträchtigt wird. Siehe „Funktionsweise von Sperren“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

- **Verhaltensänderung des ODBC-Treibers** Die ODBC-Funktion SQLTables ermöglicht es Ihnen, über das Argument SQL_ALL_SCHEMAS eine Liste aller Schemata (Benutzer) abzurufen. In früheren Versionen umfasste diese Liste nur Benutzer, die Eigentümer einer Tabelle sind. In einer neu initialisierten Datenbank wurden dadurch einige Benutzer ausgeschlossen, insbesondere DBA-Benutzer. Nun wird die vollständige Schemataliste zurückgegeben, einschließlich der Benutzer, die nicht Eigentümer einer Tabelle sind.
- **divide_by_zero_error-Option** Die Option divide_by_zero_error wurde in Version 11 nicht unterstützt. In Version 12 kann sie nun verwendet werden. Wenn Sie materialisierte Ansichten verwenden, müssen Sie diese Option aktivieren (Standardeinstellung), um neue materialisierte Ansichten erstellen zu können. Siehe „divide_by_zero_error-Option“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Einschränkungen für materialisierte Ansichten“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Verbindungsparameter PrefetchBuffer (PBUF)** In früheren Versionen wurde das mit dem Verbindungsparameter PrefetchBuffer (PBUF) festgelegte Speichervolumen für die Zeilenpufferung für alle Verbindungen verwendet. In dieser Version steht das mit diesem Parameter angegebene Speichervolumen für jede einzelne Verbindung zur Verfügung. Siehe „Verbindungsparameter PrefetchBuffer (PBUF)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Automatisches Löschen externer Logins für gelöschte Benutzer** Wenn Sie einen Benutzer aus der Datenbank entfernen, werden nun auch automatisch alle externen Logins für diesen Benutzer gelöscht. In früheren Versionen mussten Sie die externen Logins separat löschen. Siehe „Benutzer löschen (Sybase Central)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Der Datenbankaufräumvorgang und die Datenbankvalidierung können nicht mehr gleichzeitig durchgeführt werden.** In früheren Versionen konnten der Datenbankaufräumvorgang und die Datenbankvalidierung gleichzeitig durchgeführt werden. Bei gleichzeitigem Zugriff auf Datenbankseiten wurden Fehler ausgegeben. In der neuen Version lassen sich diese beiden Vorgänge nicht mehr gleichzeitig für dieselbe Datenbank durchführen. Die Validierung erfolgt erst nach dem Datenbankaufräumvorgang, und der Datenbankaufräumvorgang erfolgt erst nach der Validierung, wenn der Aufräumvorgang durch den Aufruf von sa_clean_database gestartet wurde. Der Datenbankaufräumvorgang kann gleichzeitig mit Tabellen- oder Indexvalidierungen durchgeführt werden.
- **Geänderte Datentypen für Spalten in der Systemprozedur sa_index_density** Der Datentyp der Dichte- und Überhangspalten wurde von numeric(8,6) in double geändert. Siehe „sa_index_density-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **DATEDIFF-Funktion** In früheren Versionen gab die DATEDIFF-Funktion einen INTEGER-Wert für Stunden oder kleinere Einheiten von Datumsteilen zurück. Nun gibt DATEDIFF einen BIGINT-Wert für diese Datumsteile zurück. Siehe „DATEDIFF-Funktion [Datum und Uhrzeit]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **OPENXML-Operator** Der OPENXML-Zeichenfolgenoperator gehört nicht mehr dem Datenbankbenutzer "dbo". Alle Abfragen, die den OPENXML-Operator verwenden und den

Prozedurnamen mit **dbo.** qualifizieren, müssen geändert werden, um **dbo.** zu entfernen. Das Ausführen einer Anweisung ähnlich der folgenden gibt nun einen Fehler zurück:

```
SELECT ... FROM dbo.OPENXML( ... )
```

In früheren Versionen wurden mit dem OPENXML-Operator NCHAR-Daten in die CHAR-Kodierung konvertiert und die Daten im CHAR-Format syntaktisch analysiert. Nun werden NCHAR-Daten im NCHAR-Zeichensatz analysiert, wenn NCHAR-Spalten in der Ausgabe enthalten sind.

In früheren Versionen konnten die XPath-Argumente in der WITH-Klausel nur Literalzeichenfolgen sein. Nun sind Literalzeichenfolgen und Variablen erlaubt. Siehe „OPENXML-Operator“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Datenbanken, für die ein Upgrade von Version 11 oder früher durchgeführt wird, enthalten eine Zeile für OPENXML in der SYS.SYSPROCEDURE-Systemansicht, aber die Definition kann in der Datenbank der Version 12 nicht verwendet werden. Ein Syntaxfehler wird zurückgegeben, wenn Sie versuchen, den Operator folgendermaßen zu verwenden:

```
SELECT * FROM dbo."OPENXML"( ... )
```

- **sa_text_index_vocab-Systemprozedur** Der Versuch, sa_text_index_vocab für einen NCHAR-Textindex aufzurufen, gibt nun einen Fehler zurück. Verwenden Sie stattdessen die neue sa_text_index_vocab_nchar-Systemprozedur. Siehe „sa_text_index_vocab_nchar-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Außerdem:

- Der tab_owner-Parameter ist jetzt optional.
- sa_text_index_vocab kann jetzt mit einer CALL-Anweisung verwendet werden.
- sa_text_index_vocab kann jetzt in einer Anweisung in einer Prozedur verwendet werden.
- Die Parameterwerte können nun Hostvariablen oder Ausdrücke sein.
- **Geänderte Standardkollation für Mac OS X** Wenn in früheren Versionen keine Umgebungsvariablen für Mac OS X vorhanden waren, verwendete das Dienstprogramm Initialisierung (dbinit) als standardmäßigen CHAR-Zeichensatz ISO_8859-1:1987 und für die CHAR-Kollation ISO1LATIN1 (dasselbe Verhalten wie Linux). Nun wird UTF-8 mit der UTF8BIN-Kollation verwendet. Siehe „Empfohlene Zeichensätze und Kollationen“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Reservierte Wörter** Im Folgenden finden Sie eine Liste der reservierten Wörter, die der Datenbank in SQL Anywhere Version 12.0.0 hinzugefügt wurden:
 - datetimeoffset
 - inner
 - openxml

- spatial
- treat

Im Folgenden finden Sie eine Liste der reservierten Wörter, die aus der Datenbank in SQL Anywhere Version 12.0.0 entfernt wurden:

- index_lparen
- lock
- with_cube
- with_lparen
- syntax_error
- with_rollup
- **Verhaltensänderung der Klausel WITH (*index-hint*)** Wenn bislang der Primärschlüsselindex, Fremdschlüsselindex und/oder der normale Index für eine Tabelle denselben Namen hatten, verwendete der Optimierer den Namen des Primär- oder Fremdschlüsselindexes. Nun verwendet der Optimierer den Namen des normalen Indexes. Falls kein solcher Index vorhanden ist, verwendet der Optimierer zunächst den Namen des Primärschlüssels und dann den des Fremdschlüssels. Siehe [WITH Tabellen-Hint-Klausel, FROM-Klausel \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

Nicht mehr empfohlene und nicht mehr weiterentwickelte SQL Anywhere-Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

- **Address Windowing Extensions (AWE) nicht mehr empfohlen** Die Verwendung von Address Windowing Extensions für 32-Bit-Windows wird nicht mehr empfohlen. Wenn Sie einen großen Cache benötigen, sollten Sie die 64-Bit-Version des SQL Anywhere-Datenbankservers auf einem 64-Bit-Betriebssystem verwenden.
- **CALL-Anweisung** Diese Anweisung zum Aufrufen einer Funktion wird nicht mehr empfohlen. Wenn Sie eine Funktion aufrufen und das Ergebnis einer Variablen zuordnen möchten, verwenden Sie eine Zuordnungsanweisung. Beispiel:

```
DECLARE varname INT; SET varname=test( );
```

Siehe „CALL-Anweisung“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

- **STOP ENGINE-Anweisung** Die STOP ENGINE-Anweisung wird nicht mehr empfohlen. Verwenden Sie stattdessen die STOP SERVER-Anweisung. Siehe „[STOP SERVER-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Unterstützung für Windows 2000 entfällt** Mit Version 12.0.0 wird SQL Anywhere auf Windows 2000 nicht mehr unterstützt.
 - **Unterstützung für Dienstprogramm zum Neuaufbau entfällt** Das Dienstprogramm zum Neuaufbau von SQL Anywhere-Datenbanken wird in dieser Version nicht mehr unterstützt. Zum Neuaufbau von Datenbanken verwenden Sie das Dienstprogramm zum Entladen (dbunload). Siehe „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Nicht mehr unterstützte Datenbankeigenschaften** Folgende Eigenschaften stehen in dieser Version nicht mehr zur Verfügung:
 - CheckpointLogBitmapPagesWritten-Datenbankeigenschaft
 - CheckpointLogBitmapSize-Datenbankeigenschaft
 - java_main_userid-Verbindungseigenschaft
 - QueryRowsBufferFetch-Verbindungseigenschaft
 - QueryRowsBufferFetch-Datenbankeigenschaft
 - **JDBC-basierte Serverklassen werden nicht mehr empfohlen** Die folgenden JDBC-basierten Serverklassen werden nicht mehr empfohlen:
 - ASEJDBC
 - IQJDBC
 - SAJDBC
- Um die ODBC-basierten Serverklassen verwenden zu können, müssen Anwendungen aktualisiert werden. Siehe „[Definitionen für externe ODBC-Server](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **SQL Anywhere Explorer wird nicht mehr unterstützt** Der SQL Anywhere Explorer und die SQL Anywhere-Symbolleiste für Visual Studio werden nicht mehr unterstützt. Verwenden Sie stattdessen den Server Explorer von Microsoft.
 - **Embedded SQL-Indikatorvariable vom Typ "short int" wird nicht mehr empfohlen** Um in Zukunft die Verwendung von 32- und 64-Bit-Längen und -Indikatoren zu ermöglichen, wird "short int" für Embedded SQL-Indikatorvariablen nicht mehr empfohlen. Verwenden Sie stattdessen a_sql_len. Siehe „[Indikatorvariablen](#)“ [[SQL Anywhere Server - Programmierung](#)].
 - **EngineName-Verbindungsparameter (ENG) wird nicht mehr empfohlen** Der EngineName-Verbindungsparameter (ENG) wird nicht mehr empfohlen. Verwenden Sie stattdessen den Verbindungsparameter ServerName (Server). Die Kurzform des Verbindungsparameters ServerName wurde von ENG in Server geändert. Siehe „[Verbindungsparameter ServerName \(Server\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **iAnywhere JDBC-Treiber wird nicht mehr empfohlen** Der iAnywhere JDBC-Treiber vom Typ 1 wird nicht mehr empfohlen. Verwenden Sie stattdessen den SQL Anywhere JDBC-Treiber vom Typ 2. Siehe [Neuer SQL Anywhere TYPE-2 JDBC-Treiber auf Seite 142](#).

- **Datenbankserveroption -gu all wird nicht mehr empfohlen** Der Wert **all** für die Datenbankserveroption -gu wird nicht mehr empfohlen. Siehe „Datenbankserveroption -gu“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **dbsrv12-Datenbankoption -sm (nicht mehr empfohlen)** Die Datenbankoption -sm wird nicht mehr empfohlen. Verwenden Sie stattdessen die CREATE MIRROR SERVER-Anweisung. Siehe „CREATE MIRROR SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **SET OPTION-Anweisung** Die Möglichkeit zur Angabe eines Bezeichners als Optionswert in der SET OPTION-Anweisung anstelle eines Zeichenfolgenliterals wird nicht mehr empfohlen.
- **Dienstprogramm für Dienste (dbsvc-Dienstprogramm)** Der Wert Standalone für die Option -t wird nicht mehr empfohlen. Verwenden Sie stattdessen "-t Personal", um einen Dienst für den Personal Datenbankserver zu erstellen. Siehe „Dienstprogramm für Dienste (dbsvc) für Linux“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Dienstprogramm für Dienste (dbsvc) für Windows“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung der Host-Protokolloption (IP) vom Server** Die Unterstützung für die Angabe der Host-Protokolloption (IP) in einem Serverbefehl wurde entfernt. Der folgenden Befehl wird beispielsweise nicht mehr unterstützt und gibt nun einen Fehler zurück: `dbeng12 -x tcpip(host=host-name) "%SQLANY12%\demo.db"`. Sie können jedoch auch weiterhin die Host-Protokolloption (IP) in einem CommLinks-Verbindungsparameter (LINKS) von der Clientseite angeben. Siehe „Host-Protokolloption (IP) (nur clientseitig)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Hinweis

Die Host-Protokolloption unterscheidet sich vom Host-Verbindungsparameter. Die Host-Protokolloption wird vom CommLinks-Verbindungsparameter verwendet. Sie sollten den CommLinks-Verbindungsparameter (LINKS) nur dann verwenden, wenn Sie andere TCP/IP-Optionen als Host und ServerPort (PORT) angeben müssen. Sie können nicht sowohl CommLinks als auch Host in der Verbindungszeichenfolge angeben. Siehe „Verbindungsparameter CommLinks (LINKS)“ [[SQL Anywhere Server - Datenbankadministration](#)].

In den meisten Fällen sollten Sie den HOST-Verbindungsparameter verwenden. Siehe „Host-Verbindungsparameter“ [[SQL Anywhere Server - Datenbankadministration](#)].

Neue MobiLink-Funktionen

Nachstehend finden Sie eine Liste der neuen Funktionen von MobiLink, die in Version 12.0.0 hinzugefügt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Zentrale Administration von entfernten Datenbanken** Die zentrale Administration von entfernten Datenbanken kann für Folgendes verwendet werden:
 - Zentrale Steuerung des Zeitpunkts, zu dem eine entfernte Datenbank mit MobiLink synchronisiert wird. Dies wurde zuvor auf dem Client mit der dbmlsync-Option festgelegt oder fest in eine Anwendung kodiert.

- Push von Schemaänderungen in entfernte Datenbanken.
- Hilfe bei der Diagnose von Problemen mit spezifischen Datenbanken oder mit dem Synchronisationssystem allgemein.

Weitere Hinweise zur zentralen Administration von entfernten Datenbanken finden Sie unter [„Zentrale Administration von entfernten Datenbanken“](#) [*MobiLink - Serveradministration*].

- **Neues MobiLink Replay-Dienstprogramm (mlreplay)** Das MobiLink Replay-Dienstprogramm ist ein Tool, mit dem das vom MobiLink-Server aufgezeichnete MobiLink-Protokoll wiedergegeben werden kann. Siehe [„Dienstprogramm MobiLink Replay \(mlreplay\)“](#) [*MobiLink - Serveradministration*].
- **MobiLink 12-Plug-In für Sybase Central wurde neu gestaltet** Das MobiLink-Plug-In wurde in Version 12 überarbeitet und unterstützt nun die zentrale Administration von entfernten Datenbanken. Die zwei MobiLink-Modi, Modell und Admin, wurden in dem neuen Plug-In kombiniert. Jetzt können Sie das MobiLink-Plug-In verwenden, um Synchronisationsprojekte zu erstellen, die konsolidierte Datenbanken, Gruppen, Synchronisationsmodelle und entfernte Aufgaben enthalten. Alte Synchronisationsmodelle können in Synchronisationsprojekte importiert werden. Siehe [„Zentrale Administration von entfernten Datenbanken“](#) [*MobiLink - Serveradministration*].
- **Konsolidierten Spalten Spezialwerte zuordnen** Sie können nun konsolidierten Spalten auch Spezialwerte (z.B. den ML-Benutzer) anstelle entfernter Spalten zuordnen.
- **Download von Lösch-Teilmengen wird nun unterstützt** Sie können nun Teilmengen von Downloadlöschungen festlegen, die standardmäßig mit der Downloadteilmenge identisch sind.
- **Neue Strukturdarstellung ermöglicht eine Auswahl von Spalten und Tabellen** Sie können nun sowohl die Spalten als auch die Tabellen auswählen, wenn Sie eine konsolidierte Datenbank erstellen oder aktualisieren oder Tabellen für die Synchronisation in einer vorhandenen entfernten Datenbank auswählen. Zur Vereinfachung gibt es nun eine nützliche Strukturdarstellung der Tabellen und Spalten, in die Sie neben den Tabellen und Spalten, die Sie synchronisieren wollen, Häkchen platzieren.
- **Unterstützung räumlicher Datentypen** Synchronisationsmodelle unterstützen nun räumliche Datentypen und den TIMESTAMP WITH TIME ZONE-Datentyp.
- **Schattentabellen haben nun Indizes** Für alle Schattentabellen, die vom Synchronisationsmodell erstellt wurden, werden nun Indizes erstellt. Dies kann download_delete_cursor- und download_cursor-Skripten, die die Schattentabellen verwenden, schneller machen.
- **SQL Anywhere-Monitor unterstützt nun MobiLink-Serverfarmen und Relay Server-Farmen.** Es ist nun möglich, den SQL Anywhere-Monitor zu verwenden, um MobiLink-Serverfarmen und Relay Server-Farmen sowie SQL Anywhere-Datenbanken und MobiLink-Server zu überwachen. Siehe [„SQL Anywhere-Monitor“](#) [*SQL Anywhere Server - Datenbankadministration*].
- **Neues MobiLink-Arbiterserver-Dienstprogramm für Serverfarmen** Bei der serverinitiierten Synchronisation stellt der MobiLink-Arbiters sicher, dass nur ein einziger MobiLink-Server in einer Serverfarm als Primärserver läuft. Siehe [Architektur](#) [*MobiLink - Erste Orientierung*].

- **Erweiterte Unterstützung für Serverfarmen** Eine neue Sperrenlogik für entfernte IDs wird nun verwendet, um redundante Synchronisationen derselben entfernten ID in MobiLink mit Hochverfügbarkeit zu verhindern. Die Option `-ss` muss nicht mehr gesetzt werden. Bei der Verwendung von serverinitiiertem Synchronisation mit einer MobiLink-Serverfarm ist ein Arbiter erforderlich.

Hinweis

Die Ausführung des MobiLink-Servers in einer Serverfarm ist eine Funktion der Hochverfügbarkeitsoption von MobiLink, für die eine separate Lizenz erforderlich ist. Siehe „Getrennt lizenzierbare Komponenten“ [[SQL Anywhere 16 - Einführung](#)].

- **Unterstützung für TLS-Chiffrier-Suite wurde geändert** MobiLink-Server und -Clients unterstützen nun 256-Bit-AES-Chiffrier-Suiten für RSA und ECC. Außerdem wird nun auch die Version RFC 4492 der ECC-Chiffrier-Suiten unterstützt.
- **Dynamisches Caching** Durch dynamisches Caching werden mehr Daten in den Speichercache geschrieben, sodass ein größerer Cache erforderlich sein kann, um eine Auslagerung zu vermeiden. Insgesamt sollte die Speicherverwendung etwa gleich sein.
- **Neuer integrierter Outbound Enabler** Mit dem neuen Outbound Enabler-Protokoll für die Option `-x` für `mlsrv12` können Sie einen integrierten Outbound Enabler anstelle der Standalone-Version des Outbound Enablers verwenden, die mit dem Befehl `rsoe` aufgerufen wird. Siehe „`mlsrv16-Option -x`“ [[MobiLink - Serveradministration](#)].
- **Neues Relay Server-Plug-In für Sybase Central** Mit einem neuen Relay Server-Plug-In für Sybase Central können Sie Backend-Farmen und -Server konfigurieren. Das Relay Server-Plug-In wird nur unter Windows und Linux unterstützt. Siehe „Relay Server-Plug-In für Sybase Central“ [[Relay Server](#)].
- **Anforderungen für Upload- und Download-Datenskripten** Verwenden Sie die gespeicherte Prozedur `ml_add_missing_dnld_scripts`, um das Fehlen von `download_cursor`- und/oder `download_delete_cursor`-Skripten zu beheben. Siehe „`ml_add_missing_dnld_scripts`-Systemprozedur“ [[MobiLink - Serveradministration](#)].

SQL-Anweisungen

Die folgende Liste enthält alle SQL-Erweiterungen für MobiLink in SQL Anywhere Version 12.0.0.

- **Neue SYNCHRONIZE-Anweisung [MobiLink]** Mit dieser neuen Anweisung können Sie eine entfernte SQL Anywhere-Datenbank mit einem MobiLink-Server synchronisieren. Siehe „SYNCHRONIZE-Anweisung [MobiLink]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue START SYNCHRONIZE SCHEMA CHANGE-Anweisung [MobiLink]** Mit dieser neuen Anweisung können Sie eine MobiLink-Synchronisationsschemaänderung in einer entfernten SQL Anywhere-Datenbank starten. Siehe „START SYNCHRONIZATION SCHEMA CHANGE-Anweisung [MobiLink]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue STOP SYNCHRONIZE SCHEMA CHANGE-Anweisung [MobiLink]** Mit dieser neuen Anweisung können Sie eine MobiLink-Synchronisationsschemaänderung in einer entfernten SQL

Anywhere-Datenbank stoppen. Siehe „[STOP SYNCHRONIZATION SCHEMA CHANGE-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Erweiterung der CREATE SYNCHRONIZATION SUBSCRIPTION-Anweisung [MobiLink]** Sie können nun eine Skriptversion und einen Subskriptionsnamen angeben. Siehe „[CREATE SYNCHRONIZATION SUBSCRIPTION-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung [MobiLink]** Sie können nun mit der neuen SET SCRIPT VERSION-Klausel die Skriptversion festlegen, die bei der Synchronisation verwendet werden soll. Außerdem können Sie eine Subskription mit der RENAME-Klausel umbenennen. Siehe „[ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung \[MobiLink\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Konsolidierte Datenbanken

Die folgende Liste enthält die Erweiterungen zur Unterstützung konsolidierter Datenbanken für MobiLink in SQL Anywhere Version 12.0.0.

- **Neue Unterstützung für konsolidierte Datenbanken** Die folgenden neuen konsolidierte Datenbanken werden von MobiLink in Version 12.0.0 unterstützt:
 - IBM DB2 LUW 9.7
 - SQL Anywhere 12
- **Synchronisation von räumlichen Daten** Die Synchronisation räumlicher Datentypen wird nun für folgende konsolidierte Datenbanken unterstützt: SQL Anywhere, Oracle, Microsoft SQL Server, IBM DB/2 und MySQL. Siehe „[Synchronisation von räumlichen Daten](#)“ [[MobiLink - Serveradministration](#)].
- **Unterstützung längerer CHAR-Spalten** Der MobiLink-Server unterstützt nun die Synchronisation von CHAR-, VARCHAR-, NCHAR- und NVARCHAR-Spalten entfernter Datenbanken mit einer Länge von mehr als 32767 Byte.
- **Unterstützung von Synchronisationsspalten vom Datentyp TIMESTAMP WITH TIME ZONE** Der MobiLink-Server unterstützt nun die Synchronisation von Spalten entfernter Datenbanken vom Datentyp TIMESTAMP WITH TIME ZONE.

iAnywhere Solutions 12 - Oracle ODBC-Treiber

Die folgende Liste enthält die in SQL Anywhere Version 12.0.0 eingeführten Erweiterungen des Oracle ODBC-Treibers von iAnywhere Solutions 12

- **VARRAY-Unterstützung für Prozeduren in Packages** Der Oracle ODBC-Treiber von iAnywhere Solutions 12 unterstützt nun die Verwendung von VARRAYs in Paketprozeduren.

MobiLink-Server

Die folgende Liste enthält die in SQL Anywhere Version 12.0.0 eingeführten Erweiterungen des MobiLink-Servers.

Neue mlsrv12-Funktionen

- **Änderung des Servernamens in mlsrv12** Der Name des MobiLink-Servers wurde von mlsrv11 in mlsrv12 geändert.
- **Neue Option -cmax für mlsrv12** Diese neue Option ist Teil der dynamischen Cachedimensionierungsfunktion. Sie legt die maximale Cachegröße des Serverspeichers fest. Siehe „[mlsrv16-Option -cmax](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -cmin für mlsrv12** Diese neue Option ist Teil der dynamischen Cachedimensionierungsfunktion. Sie legt die Ausgangsgröße für den Hauptspeichercache des Servers fest. Siehe „[mlsrv16-Option -cinit](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -cinit für mlsrv12** Diese neue Option ist Teil der dynamischen Cachedimensionierungsfunktion. Sie legt die Mindestgröße für den Hauptspeichercache des Servers fest. Siehe „[mlsrv16-Option -cmin](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Protokolloption für Option -x für mlsrv12** Die Option -x für mlsrv12 unterstützt nun eine neue oe-Protokolloption, die es gestattet, bei der Verwendung des Relay Servers den neuen integrierten Outbound Enabler zu verwenden. Siehe „[mlsrv16-Option -x](#)“ [[MobiLink - Serveradministration](#)].
- **Neuer Standardwert für Java-Option -sl für mlsrv12** Standardmäßig versucht der MobiLink-Server nun, anstatt einer Client-Java-VM eine Server-Java-VM zu laden, sofern eine verfügbar ist. Um diese neue Standardeinstellung aufzuheben, können Sie eine Client-VM explizit anfordern, indem Sie den -sl-Java-Optionen die Option -client hinzufügen. Siehe „[mlsrv16-Option -sl java](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -ca für mlsrv12** Geben Sie den Hostnamen an, der den MobiLink-Arbiter ausführt, wenn mehrere Server mit serverinitiiertem Synchronisation ohne Lightweight-Polling ausgeführt werden. Siehe „[mlsrv16-Option -ca](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -ftru für mlsrv12** Diese Option stellt Unterstützung für den Datei-Upload im MobiLink-Server bereit. Sie können mit dieser neuen Option das Stammverzeichnis für Dateien festlegen, die mit dem mlfiletransfer-Dienstprogramm hochgeladen werden sollen. Siehe „[mlsrv16-Option -ftru](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Metrik für Option -ppv für mlsrv12** Der Option -ppv für mlsrv12 wurden eine Reihe neuer Metriken hinzugefügt. Siehe „[mlsrv16-Option -ppv](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -rrp für mlsrv12** Die neue Option -rrp für mlsrv12 wurde zusammen mit dem neuen MobiLink Replay-Dienstprogramm (mlreplay) hinzugefügt. Die Option bewirkt, dass der MobiLink-Server das mlreplay-Dienstprogramm ausführt und beim Serverstart alle aufgezeichneten Sitzungen im angegebenen Verzeichnis wiedergibt. Siehe „[mlsrv16-Option -rrp](#)“ [[MobiLink - Serveradministration](#)].

- **Neue Option -rp für mlsrv12** Die neue Option -rp für mlsrv12 wurde zusammen mit dem neuen MobiLink Replay-Dienstprogramm (mlreplay) hinzugefügt. Mit dieser Option wird das Verzeichnis angegeben, in dem die Synchronisationen für die Wiedergabe mit dem mlreplay-Dienstprogramm aufgezeichnet werden. Siehe „[mlsrv16-Option -rp](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -vR für mlsrv12** Mit der neuen Option -vR für mlsrv12 kann die entfernte ID in jeder Lognachricht für die Synchronisation angezeigt werden. Siehe „[mlsrv16-Option -v](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -vU für mlsrv12** Mit der neuen Option -vU für mlsrv12 können Sie den MobiLink-Benutzernamen in jeder Lognachricht für die Synchronisation anzeigen. Siehe „[mlsrv16-Option -v](#)“ [[MobiLink - Serveradministration](#)].
- **Neue Option -vk für mlsrv12** Diese neue Option ist Teil der dynamischen Cachedimensionierungsfunktion. Diese Option schreibt eine Zeile ins Log, wenn der Cache anwächst oder abnimmt. Siehe „[mlsrv16-Option -v](#)“ [[MobiLink - Serveradministration](#)].
- **Performanceverbesserungen bei geringem Speicherplatz auf dem MobiLink-Server** Das Verhalten und die Robustheit bei geringem verfügbarem Speicher wurden verbessert.
- **Zwei neue Konflikterkennungsansätze** Die Skripten upload_fetch, upload_fetch_column_conflict, upload_new_row_insert und upload_old_row_insert können verwendet werden, um Konflikte zu erkennen. Siehe „[Konflikterkennung](#)“ [[MobiLink - Serveradministration](#)].
- **Neue ml_add_missing_dnlc_scripts-Systemprozedur** Die neue Systemprozedur kann verwendet werden, um fehlende download_cursor- und download_delete_cursor-Skripten hinzuzufügen. Siehe „[ml_add_missing_dnlc_scripts-Systemprozedur](#)“ [[MobiLink - Serveradministration](#)].

Neue API-Funktionen des MobiLink-Servers

- **Neue TimestampWithTimeZone-Klasse hinzugefügt (Java)** Diese Klasse stellt Methoden bereit, mit denen Sie den Zeitzoneoffset eines Zeitstempelwerts abfragen und festlegen können. Siehe [TimestampWithTimeZone-Klasse \[MobiLink-Server Java\]](#) [[MobiLink - Serveradministration](#)].
- **Neue DateTimeWithTimeZone-Klasse hinzugefügt (.NET)** Diese Klasse stellt Methoden bereit, mit denen Sie den Zeitzoneoffset eines DateTime-Werts abfragen und festlegen können. Siehe [DateTimeWithTimeZone-Klasse \[MobiLink-Server .NET\]](#) [[MobiLink - Serveradministration](#)].
- **Neue SpatialUtilities-Klassen hinzugefügt (.NET und Java)** Diese Klasse stellt Methoden für die Arbeit mit räumlichen Daten bereit. Siehe [SpatialUtilities-Klasse \[MobiLink-Server .NET\]](#) [[MobiLink - Serveradministration](#)] (.NET) und [SpatialUtilities-Klasse \[MobiLink-Server Java\]](#) [[MobiLink - Serveradministration](#)]. (Java)

Neue MobiLink-Skriptfunktionen

- **Neue Systemparameter für Skripten upload_fetch und upload_fetch_column_conflict** Für die Skripten upload_fetch und upload_fetch_column_conflict sind nun die MobiLink-Server-Systemparameter **remote_id** und **username** verfügbar.

- **Neue Parameter für authenticate_file_transfer-Skripten** Für authenticate_file_transfer-Skripten sind neue Parameter verfügbar. Siehe „authenticate_file_transfer (Verbindungsereignis)“ [\[MobiLink - Serveradministration\]](#).
- **Neues authenticate_file_upload-Verbindungsereignis** Ein neues Verbindungsereignis ist zur Unterstützung von Datei-Uploads verfügbar. Siehe „authenticate_file_upload (Verbindungsereignis)“ [\[MobiLink - Serveradministration\]](#).
- **Neues generate_next_last_download_timestamp-Skript** Verwenden Sie dieses Skript anstelle von modify_next_last_download_timestamp, wenn Sie den Standardalgorithmus von MobiLink zur Ermittlung der nächsten letzten Downloadzeit unterdrücken möchten. Siehe „generate_next_last_download_timestamp (Ereignis)“ [\[MobiLink - Serveradministration\]](#).

Serverdienstprogramme

Die folgende Liste enthält die in SQL Anywhere Version 12.0.0 eingeführten Erweiterungen der MobiLink-Serverdienstprogramme.

- **Neues mlarbiter-Dienstprogramm** Mit dem neuen Dienstprogramm mlarbiter wird der MobiLink-Arbiterserver gestartet. Der Arbitrer ist erforderlich, wenn serverinitiierte Synchronisation mit einer MobiLink-Serverfarm verwendet wird. Siehe „MobiLink-Arbiterserver-Dienstprogramm für Windows (mlarbiter)“ [\[MobiLink - Serveradministration\]](#).

MobiLink-Monitor

Die folgende Liste enthält die in SQL Anywhere Version 12.0.0 eingeführten Erweiterungen des MobiLink-Monitors.

- **Neue Einträge im Diagrammfensterausschnitt** Die folgenden Einträge wurden dem Diagrammfensterausschnitt im MobiLink-Monitor hinzugefügt: OE-Arbeitswarteschlange, Notifier-Arbeitswarteschlange und Work-Warteschlange für dynamischen Cache. Siehe „Funktionsweise des Fensterausschnitts Auslastungsdiagramm“ [\[MobiLink - Serveradministration\]](#).
- **Neue Option Datei des vertrauenswürdigen Zertifikats** Die neue Option **Datei des vertrauenswürdigen Zertifikats** und die Schaltfläche **Durchsuchen** wurden dem Fenster **Mit dem MobiLink-Server verbinden** hinzugefügt.

MobiLink-Clients

Die folgende Liste enthält die in SQL Anywhere Version 12.0.0 eingeführten Erweiterungen für MobiLink-Clients.

- **Erweiterte Unterstützung für Hintergrundsynchrisation in dbmlsync** Die Datenbank-Engine kann nun die dbmlsync-Verbindung mit der entfernten Datenbank (und nicht festgeschriebene Vorgänge in dbmlsync) löschen, wenn eine andere Verbindung auf den Zugriff auf eine beliebige Datenbankressource wartet, die von dbmlsync gesperrt wurde. Dadurch kann die andere Verbindung genutzt werden, ohne auf den Abschluss der Synchronisation zu warten.

Verwenden Sie diese Optionen für die Hintergrundsynchrisation:

- „dbmlsync-Option -bk“ [\[MobiLink - Clientadministration\]](#)
 - „dbmlsync-Option -bkr“ [\[MobiLink - Clientadministration\]](#)
 - Optionen Background und BackgroundRetry für „MobiLink-Synchronisationsprofile“ [\[MobiLink - Clientadministration\]](#)
 - **dbmlsync unterstützt nun benannte Subskriptionen** Mit der neuen Option -s für dbmlsync können Sie die Namen von zu synchronisierenden Subskriptionen angeben. Siehe „dbmlsync-Option -s“ [\[MobiLink - Clientadministration\]](#).
 - **HTTP-Antwortpuffer** Mit der neuen Netzwerkprotokolloption http_buffer_responses können Sie HTTP-Pakete von MobiLink in einen Zwischenpuffer einlesen, bevor sie verarbeitet werden, anstatt sie unmittelbar bei ihrem Empfang zu verarbeiten. Siehe „http_buffer_responses“ [\[MobiLink - Clientadministration\]](#).
 - **Clientseitige Zertifikate können nun verwendet werden, um MobiLink-Clients bei Servern von Drittherstellern und Proxys zu authentifizieren** Die Synchronisationsparameter identity und identity_password wurden zur Unterstützung dieser Funktion hinzugefügt.
 - **Keine Synchronisation mehr erforderlich, um Schemaänderungen auf entfernten Datenbanken zu implementieren (nur für SQL Anywhere-Clients)** Sie können Schemaänderungen auf der entfernten Datenbank mit den Anweisungen START SYNCHRONIZATION SCHEMA CHANGE und STOP SYNCHRONIZATION SCHEMA CHANGE begrenzen und damit vereinfachen. Siehe:
 - „START SYNCHRONIZATION SCHEMA CHANGE-Anweisung [MobiLink]“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „STOP SYNCHRONIZATION SCHEMA CHANGE-Anweisung [MobiLink]“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
- Mit der Anweisung CREATE SYNCHRONIZATION SUBSCRIPTION oder ALTER SYNCHRONIZATION SUBSCRIPTION können Sie für jede Synchronisationssubskription eine Skriptversion festlegen. Siehe:
- „CREATE SYNCHRONIZATION SUBSCRIPTION-Anweisung [MobiLink]“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung [MobiLink]“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - **Neue mlfileupload-Methode für UltraLite** Die MLFileUpload-Methode wurde für UltraLite-Clients hinzugefügt.

- **Änderungen des Fensters der dbmlsync-Optionen** Die folgenden Änderungen wurden im dbmlsync-Optionenfenster vorgenommen:
 - **Neuversuch gemäß entferntem Verarbeitungsfortschritt, Entfernter liegt zurück, Entfernter liegt voran, Verbindungen mit Konflikten trennen, Standortskript und Befehlszeilen-Hilfe** wurden entfernt.
 - Die Option **Publikation** wurde durch die Option **Subskription** ersetzt.

Neue dbmlsync-Optionen

- **Neue Option -s für dbmlsync** Mit der neuen Option -s für dbmlsync können Sie die Namen von zu synchronisierenden Subskriptionen angeben. Siehe „dbmlsync-Option -s“ [[MobiLink - Clientadministration](#)].
- **Neue Option -bk für dbmlsync** Diese Option aktiviert die Hintergrundsynchroisation. Siehe „dbmlsync-Option -bk“ [[MobiLink - Clientadministration](#)].
- **Neue Option -bkr für dbmlsync** Diese Option steuert das Verhalten von dbmlsync nach der Unterbrechung einer Hintergrundsynchroisation. Siehe „dbmlsync-Option -bkr“ [[MobiLink - Clientadministration](#)].
- **Neue Option -ci für dbmlsync** Diese Option ist Teil der neuen dynamischen Cachedimensionierungsfunktion. Sie können mit dieser Option die Ausgangsgröße des von dbmlsync für Synchronisationsdaten verwendeten Cachespeichers festlegen. Siehe „dbmlsync-Option -ci“ [[MobiLink - Clientadministration](#)].
- **Neue Option -cl für dbmlsync** Diese Option ist Teil der neuen dynamischen Cachedimensionierungsfunktion. Sie können mit dieser Option die Mindestgröße festlegen, auf die der dbmlsync-Cache reduziert wird. Siehe „dbmlsync-Option -cl“ [[MobiLink - Clientadministration](#)].
- **Neue Option -cm für dbmlsync** Diese Option ist Teil der neuen dynamischen Cachedimensionierungsfunktion. Sie können mit dieser Option die Maximalgröße für die dbmlsync-Cachedatei festlegen. Siehe „dbmlsync-Option -cm“ [[MobiLink - Clientadministration](#)].
- **Neue erweiterte Option BufferDownload für dbmlsync** Diese Option ist Teil der neuen dynamischen Cachedimensionierungsfunktion. Die Option legt fest, ob der gesamte Download vom MobiLink-Server in den Cache eingelesen werden soll, bevor er auf die entfernte Datenbank angewendet wird. Siehe „Erweiterte Option BufferDownload (bd)“ [[MobiLink - Clientadministration](#)].

Neue C++ API-Objekte für dbmlsync

- **CancelSync-Methode** Diese Methode ermöglicht es MobiLink-Clients, eine Synchronisation abubrechen. Siehe [DbmlsyncClient.CancelSync-Methode \[Dbmlsync C++\]](#) [[MobiLink - Clientadministration](#)].
- **DBSC_CancelRet-Enumeration** Diese Enumeration gibt das Ergebnis des Versuchs an, eine Synchronisation abubrechen. Siehe [DBSC_CancelRet-Enumeration \[Dbmlsync C++\]](#) [[MobiLink - Clientadministration](#)].

- **DBSC_ERR_ACTIVE_SYNC_NOT_CANCELED-Mitglied** Dieses Mitglied gibt an, dass der Server die Synchronisationsanforderung nicht abbrechen konnte, da die Systemprozedur aktiv war. Siehe [DBSC_ErrorType-Enumeration \[Dbmlsync C++\]](#) [*MobiLink - Clientadministration*].
- **DBSC_ERR_DEAD_SERVER-Mitglied** Dieses Mitglied gibt an, dass der dbmlsync-Server beim Hochfahren auf einen Fehler gestoßen ist und daher herunterfährt. Siehe [DBSC_ErrorType-Enumeration \[Dbmlsync C++\]](#) [*MobiLink - Clientadministration*].
- **enable status-Eigenschaft** Drei neue Ereignisse (DBSC_EVENTTYPE_ML_CONNECT, DBSC_EVENTTYPE_UPLOAD_COMMITTED und DBSC_EVENTTYPE_DOWNLOAD_COMMITTED) werden an den MobiLink Client gesendet, wenn diese Eigenschaft aktiviert ist. Siehe [DBSC_EventType-Enumeration \[Dbmlsync C++\]](#) [*MobiLink - Clientadministration*], [DbmlsyncClient.SetProperty-Methode \[Dbmlsync C++\]](#) [*MobiLink - Clientadministration*] und [DbmlsyncClient.GetProperty-Methode \[Dbmlsync C++\]](#) [*MobiLink - Clientadministration*].

Neue .NET-API-Objekte für dbmlsync

- **CancelSync-Methode** Diese Methode ermöglicht es MobiLink-Clients, eine Synchronisation abubrechen. Siehe [DbmlsyncClient.CancelSync-Methode \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*].
- **DBSC_CancelRet-Enumeration** Diese Enumeration gibt das Ergebnis des Versuchs an, eine Synchronisation abubrechen. Siehe [DBSC_CancelRet-Enumeration \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*].
- **DBSC_ERR_ACTIVE_SYNC_NOT_CANCELED-Mitglied** Dieses Mitglied gibt an, dass der Server die Synchronisationsanforderung nicht abbrechen konnte, da die Systemprozedur aktiv war. Siehe [DBSC_ErrorType-Enumeration \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*].
- **DBSC_ERR_DEAD_SERVER-Mitglied** Dieses Mitglied gibt an, dass der dbmlsync-Server beim Hochfahren auf einen Fehler gestoßen ist und daher herunterfährt. Siehe [DBSC_ErrorType-Enumeration \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*].
- **"enable status"-Eigenschaft** Drei neue Ereignisse (DBSC_EVENTTYPE_ML_CONNECT, DBSC_EVENTTYPE_UPLOAD_COMMITTED und DBSC_EVENTTYPE_DOWNLOAD_COMMITTED) werden an den MobiLink Client gesendet, wenn diese Eigenschaft aktiviert ist. Siehe [DBSC_EventType-Enumeration \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*], [DbmlsyncClient.SetProperty-Methode \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*] und [DbmlsyncClient.GetProperty-Methode \[Dbmlsync .NET\]](#) [*MobiLink - Clientadministration*].

Neue Clientdienstprogramm-Funktionen

- **Dateien mit mlfiletransfer hochladen** Sie können nun Dateien mit dem mlfiletransfer-Dienstprogramm hochladen. Siehe „[MobiLink-Dienstprogramm für die Dateiübertragung \(mlfiletransfer\)](#)“ [*MobiLink - Clientadministration*].
- **Neue Option -i für mlfiletransfer-Dienstprogramm** Die Option -i wurde mlfiletransfer hinzugefügt, um die Wiederaufnahmefunktion zu deaktivieren. Siehe „[MobiLink-Dienstprogramm für die Dateiübertragung \(mlfiletransfer\)](#)“ [*MobiLink - Clientadministration*].

Relay Server

Folgende Relay Server-Funktionen wurden in dieser Version hinzugefügt:

- **Unterstützung für SQL Anywhere-Monitor** Relay Server-Ressourcen können nun mit dem SQL Anywhere-Monitor überwacht werden. Siehe „[SQL Anywhere-Monitor](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung für die zentrale Administration** Der Relay Server unterstützt die zentrale Administration über Sybase Central.
- **Unterstützung für RSOE unter Mac OS** Der Relay Server Outbound Enabler wird nun unter Mac OS unterstützt. Weitere Hinweise zu unterstützten Plattformen finden Sie unter <http://www.sybase.com/detail?id=1061806>.
- **Windows IIS 7 und 7.5 werden nun unterstützt** Der Relay Server wird nun für IIS 7 unter Windows 2008 und für IIS 7.5 unter Windows Server 2008 R2 unterstützt. Weitere Hinweise zu unterstützten Plattformen finden Sie unter <http://www.sybase.com/detail?id=1061806>.
- **Neue active_cookie-Option** Die Option active_cookie wurde dem Backend-Farm-Abschnitt der Relay Server-Konfiguration hinzugefügt. Siehe [Eigenschaften im Backend-Farm-Abschnitt](#) [[Relay Server](#)].
- **Neue active_header-Option** Die Option active_header wurde dem Backend-Farm-Abschnitt der Relay Server-Konfiguration hinzugefügt. Siehe [Eigenschaften im Backend-Farm-Abschnitt](#) [[Relay Server](#)].
- **Verbesserungen der Relay Server-Fehlerbehandlung** Für eine verbesserte Fehlerbehandlung verfügt der Relay Server nun über standardisierte Relay Server-Fehlercodes mit lokalisierten Fehlermeldungen und selektiven Fehlermeldungen mit Systemfehlercodes und eingebetteten Systemfehlermeldungen. Siehe „[Fehlermeldungen des Relay Servers](#)“ [[Fehlermeldungen](#)].
- **Unterstützung für SQL Anywhere-Datenbank als Backend-HTTP-Server** Der Relay Server unterstützt nun SQL Anywhere-Failover und Scale-Out mit Schreibschutz. Siehe „[SQL Anywhere-Webdienste mit Hochverfügbarkeit und Scale-Out-Lösungen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Dynamische Vorabbelegung von Speicherplatz für Antwortpuffer für den Outbound Enabler** Die dynamische Vorabbelegung von Speicherplatz für den Antwortpuffer hat den Speicher-Overhead des Outbound Enablers signifikant vermindert.
- **Verbesserungen der Benutzeroberfläche für den Outbound Enabler** Der Outbound Enabler-Benutzeroberfläche wurde eine Instanzen-ID im Fenstertitel, Taskleiten-Quickinfo und ein Taskleiten-Menü hinzugefügt. Der Taskleiten-Quickinfo wurde der Status des Outbound Enablers hinzugefügt.
- **Effizientere Verwendung des Shared Memorys** Der Relay Server verwendet das Shared Memory (Einstellung mit der shared_mem-Konfigurationsoption) effizienter. In Deployments mit relativ langsam lesenden Clients, die HTTP-Anforderungen mit umfangreichen Antworten ausführen, kann der Relay Server mit deutlich weniger Shared Memory ausgeführt werden.

- **Der Relay Server Outbound Enabler unterstützt nun periodische Backend-Server-Statusanforderungen mit HTTP** Der RSOE wurde erweitert und unterstützt nun periodische Backend-Server-Statusanforderungen unter Verwendung von HTTP. Eine Backend-Server-Statusanforderung ist eine Alternative zu Verfügbarkeit-Pings und kann verwendet werden, um zu ermitteln, ob der Backend-Server in der Lage ist, Clientanforderungen anzunehmen. Der neue `status_url`-Parameter, der als Teil der `rsoc`-Option `-cs` angegeben wird, wird verwendet, um periodische Backend-Server-Statusanforderungen zu aktivieren.

Verhaltensänderungen von MobiLink

Nachstehend finden Sie eine Liste der Verhaltensänderungen von MobiLink, die in Version 12.0.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Änderungen des MobiLink-Servers

- **MobiLink-Server benötigt *log4j.jar* nicht mehr** Die Datei *log4j.jar* ist für den MobiLink-Server nicht mehr erforderlich und wird nicht mehr mit ihm per Deployment bereitgestellt. Falls Sie *log4j.jar* brauchen, müssen Sie eine eigene Version der JAR-Datei installieren und in den Classpath aufnehmen.
- **Neues Verhalten für Option `-cn`** Die Option `-cn` für `mlsrv12` legt die maximale Anzahl von Datenbankverbindungen fest, die für Datenbank-Worker-Threads verwendet werden. In früheren Versionen vor SQL Anywhere 12 wurde die maximale Anzahl von Datenbankverbindungen mit der `mlsrv12`-Option `-cn` festgelegt. Siehe „[mlsrv16-Option -cn](#)“ [*MobiLink - Serveradministration*].
- **Neues Verhalten für Option `-sl dnet`** In früheren Versionen lud der MobiLink-Server die Workstation-CLR standardmäßig bei der Verwendung von .NET-Skripten. Nun lädt er stattdessen die Server-CLR. Sie können das alte Verhalten wiederherstellen, indem Sie `-clrFlavor=wks` zur `dnet`-Option `-sl` für `mlsrv12` hinzufügen. Siehe „[mlsrv16-Option -sl dnet](#)“ [*MobiLink - Serveradministration*].
- **MobiLink-Server verwendet nun die Oracle-Systemansicht `GV_$TRANSACTION` anstelle von `V_$TRANSACTION`** Das vom MobiLink-Server verwendete Oracle-Konto muss nun die Berechtigung für die Oracle-Systemansicht `GV_$TRANSACTION` anstatt für die Systemansicht `V_$TRANSACTION` haben. Siehe „[Konsolidierte Oracle-Datenbank](#)“ [*MobiLink - Serveradministration*].
- **Upgradeskripten für Version 6.0.x entfernt** Die MobiLink-Upgradeskripten für 6.0.x wurden entfernt. Wenn Sie dieses Upgrade benötigen, wenden Sie sich an den technischen Support (<http://www.sybase.com/support>).
- **`ml_add_column`-Systemprozedur nicht mehr erforderlich** Für Clients der Version 12 oder höher ist die `ml_add_column`-Systemprozedur nicht mehr erforderlich, wenn Sie Spaltennamen in benannten Parametern verwenden wollen. Standardmäßig können Sie nun die Spaltennamen direkt und ohne zusätzliche Setup-Einstellungen referenzieren. Siehe „[ml_add_column-Systemprozedur \(nicht mehr empfohlen\)](#)“ [*MobiLink - Serveradministration*].

- **Neue BIGINT-Datentypunterstützung für Java und .NET-Server-APIs** Der BIGINT SQL-Datentyp wird nun den Datentypen LONG Java und .NET zugeordnet. Siehe „[SQL-Java-Datentypen](#)“ [[MobiLink - Serveradministration](#)] und „[SQL-.NET-Datentypen](#)“ [[MobiLink - Serveradministration](#)].
- **Datenskripten sind nun erforderlich** Um die Wahrscheinlichkeit eines Datenverlusts aufgrund einer versehentlichen Nichterstellung von Skripten zu reduzieren, erfordert der MobiLink-Server nun ein ignoriertes Skript oder ein gültiges Skript für die folgenden Ereignisse. Siehe „[Ignorierte Skripten](#)“ [[MobiLink - Serveradministration](#)].
 - **upload_insert** Wenn eingefügte Zeilen aus entfernten Datenbanken hochgeladen werden und kein handle_UploadData-Verbindungsskript definiert ist.
 - **upload_update** Wenn aktualisierte Zeilen aus entfernten Datenbanken hochgeladen werden und kein handle_UploadData-Verbindungsskript definiert ist.
 - **upload_delete** Wenn gelöschte Zeilen aus entfernten Datenbanken hochgeladen werden und kein handle_UploadData-Verbindungsskript definiert ist.
 - **download_cursor und download_delete_cursor** Wenn kein handle_DownloadData-Verbindungsskript definiert ist und die Synchronisation nicht nur ein reiner Upload ist.

Als Erleichterung bei einem Upgrade auf Version 12 können Sie mit der gespeicherten Prozedur `ml_add_missing_dnld_scripts` ignorierte Skripten hinzufügen, um Fehler aufgrund von fehlenden Downloadskripten zu vermeiden. Siehe „[ml_add_missing_dnld_scripts-Systemprozedur](#)“ [[MobiLink - Serveradministration](#)].

Änderungen des MobiLink-Clients

Im Folgenden finden Sie eine Liste der Verhaltensänderungen von MobiLink-Clients in Version 12.0.0.

- **MobiLink-Synchronisationsclients senden standardmäßig Spaltennamen** In Version 12 senden alle MobiLink-Synchronisationsclients standardmäßig Spaltennamen an den MobiLink-Server. Das bedeutet in den meisten Fällen, dass es nicht mehr erforderlich ist, die gespeicherte Systemprozedur `ml_add_column` zu verwenden, um Spaltennamen zu definieren und die Reihenfolge von benannten Parametern in MobiLink-Skripten festzulegen. Siehe „[ml_add_column-Systemprozedur \(nicht mehr empfohlen\)](#)“ [[MobiLink - Serveradministration](#)].
- **Zusätzliche Unterstützung für die Festlegung der erweiterten TLS-Sitzung** MobiLink-Clients unterstützen nun eine neue TLS-Erweiterung, die eine Sicherung sicherheitsgefährdeter Server anderer Hersteller ermöglicht.
- **MLFileTransfer-Methode umbenannt** Die Methode `MLFileTransfer` für UltraLite-Clients wurde in die Methoden `MLFileUpload` und `MLFileDownload` aufgeteilt.
- **-df in -lf umbenannt** Die Option `-df` für das Dienstprogramm `mlfiletransfer` wurde in `-lf` umbenannt und bezieht sich nun auf eine lokale Datei anstatt auf eine Zieldatei.
- **-dp in -lp umbenannt** Die Option `-dp` für das Dienstprogramm `mlfiletransfer` wurde in `-lp` umbenannt und bezieht sich nun auf einen lokalen Pfad anstatt auf einen Zielpfad.

- **UPLD_ERR_USERID_ALREADY_IN_USE geändert** Dbmlsync gibt nicht mehr UPLD_ERR_USERID_ALREADY_IN_USE als Fehlerursache für den Ereignis-Hook sp_hook_dbmlsync_upload_end zurück. Stattdessen wird der Wert UPLD_ERR_REMOTE_ID_ALREADY_IN_USE zurückgegeben.
- **Palm wird nicht mehr für UltraLite-Clients unterstützt** Palm OS wird nicht mehr unterstützt. Wenn Sie Palm verwenden möchten, sollten Sie weiterhin mit SQL Anywhere 11 arbeiten.

Änderungen des MobiLink-Plug-Ins für Sybase Central

Im Folgenden finden Sie eine Liste der Verhaltensänderungen für das MobiLink-Plug-In für Sybase Central in Version 12.0.0.

- **Änderungen der Zuordnungseeditoren für Synchronisationsmodelle**
 - Die Editoren für die Tabellen- und Spaltenzuordnung für die Synchronisationsmodelle zeigen nun konsolidierte Datenbanken auf der linken Seite und entfernte Datenbanken auf der rechten Seite an. Siehe „[Tabellen- und Spaltenzuordnungen](#)“ [[MobiLink - Erste Orientierung](#)].
 - Nur die synchronisierten konsolidierten Tabellen werden im Tabellenzuordnungseeditor angezeigt. Das Ändern der Zuordnungsrichtung auf **Nicht synchronisiert** entspricht dem Löschen der Tabellenzuordnung. Die betreffende Zeile wird aus dem Editor entfernt, wenn das Modell gespeichert wird, ohne das Datenbankschema zu ändern.
 - Um eine Tabellenzuordnung für nicht synchronisierte konsolidierte Tabellen hinzuzufügen und optional die Tabelle dem entfernten Schema hinzuzufügen, verwenden Sie den Befehl **Neue Tabellenzuordnungen**. Sie können das konsolidierte oder entfernte Schema des Modells mit dem Befehl **Schema aktualisieren** auch ändern.
 - Es werden nun Popupmenüs anstelle von Dropdown-Listen verwendet, um die rechte Seite von Tabellen- und Spaltenzuordnungen zu wählen. Ein optionales Fenster ist verfügbar (Zugriff über die Schaltfläche mit den Auslassungspunkten), wenn nicht alle Optionen in das Menü passen. Für Tabellenzuordnungen werden nur nicht synchronisierte entfernte Tabellen aufgelistet. Für Spaltenzuordnungen sind nicht synchronisierte Spalten in der entfernten Tabelle zusammen mit den Optionen für Wertezuordnungen und die Nicht-Synchronisation der Spalte aufgelistet.
 - Wenn Sie eine Synchronisationsoption für eine Tabellenzuordnung ändern, wechselt der untere Fensterausschnitt automatisch zur Registerkarte für die entsprechenden Unteroptionen.
- **VARBIT- und LONG VARBIT-Spalten** Die Spalten VARCHAR und LONG VARCHAR werden anstelle von VARBIT- und LONG VARBIT-Spalten benutzt, wenn ein Deployment eines Synchronisationsmodells mit einem entfernten Schema auf einer neuen entfernten UltraLite-Datenbank vorgenommen wird.
- **GO nun als Trennzeichen für Anweisungen** In SQL Anywhere- und UltraLite-Datenbanken verwenden Synchronisationsmodelle nun GO als Trennzeichen für Anweisungen anstelle eines Semikolons. Dies gestattet es, den mit Synchronisationsmodellen generierten SQL-Code bei der zentralen Verwaltung von entfernten Aufgaben zu verwenden.

Nicht mehr empfohlene MobiLink-Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

- **IBM DB2 Mainframe in Version 12.0.0 oder höher nicht mehr unterstützt** IBM DB2 Mainframe wird in Version 12.0.0 nicht als konsolidierte Datenbank unterstützt. MobiLink unterstützt dagegen weiterhin DB2 LUW (Linux, Unix und Windows) als konsolidierte Datenbank.
- **Adaptive Server Enterprise 12.5.x nicht mehr unterstützt** Adaptive Server Enterprise 12.5.x wird von MobiLink in Version 12.0.0 nicht mehr unterstützt.
- **IBM DB2 LUW 8.2 nicht mehr unterstützt** IBM DB2 LUW 8.2 wird von MobiLink in Version 12.0.0. nicht mehr unterstützt.
- **Option -xo für mlsrv12 wurde entfernt** Ältere Clients als Version 10 werden nicht mehr unterstützt.
- **Option -f für mlsrv12 wurde entfernt** Mit der Option -zf für mlsrv12 können Sie festlegen, dass der MobiLink-Server zu Beginn jeder Synchronisation auf Skriptänderungen prüfen soll. Siehe „mlsrv16-Option -zf“ [[MobiLink - Serveradministration](#)].
- **Option -nba für mlsrv12 wurde entfernt** Die Blockierung der Downloadbestätigung wird nicht mehr unterstützt. Wenn die entfernte Datenbank eine Downloadbestätigung anfordert, verwendet der MobiLink-Server immer eine nicht blockierende Bestätigung.
- **Option -fr für mlsrv12 wurde entfernt** Die Option -fr für mlsrv12 wird nicht mehr unterstützt. Wenn Sie ein Skript (das den Verlust von Daten verursachen könnte) ignorieren möchten, definieren Sie das Skript als --{ml_ignore}.
- **SQL-Rückgabe von Java- und .NET-Datenskripten wird nicht mehr empfohlen** Die Fähigkeit der Java- und .NET-Skriptenlogik, Zeichenfolgen zurückzugeben, die vom MobiLink-Server als SQL-Skripten interpretiert werden, wird in Datenskripten nicht mehr empfohlen. Wenn Ihre Skripten Änderungen in der konsolidierten Datenbank veranlassen müssen, sollten sie dies direkt aus Java oder .NET tun.

Siehe:

- [Java- und .NET-Datenskripten mit SQL-Rückgabe \(entfernt\)](#) [[MobiLink - Serveradministration](#)]
- [„Direkte Zeilenbehandlung“](#) [[MobiLink - Serveradministration](#)]
- [„Datenskripten“](#) [[MobiLink - Serveradministration](#)]
- **Downloadfehler-Hooks wurde entfernt** Die folgenden dbmlsync-Hooks wurden in Version 12 entfernt: sp_hook_dbmlsync_download_com_error, sp_hook_dbmlsync_download_fatal_sql_error und sp_hook_dbmlsync_download_sql_error.
- **Option -f für das MobiLink-Dienstprogramm für die Dateiübertragung wurde entfernt** Die Option -f für das mlfiletransfer-Dienstprogramm wird nicht mehr unterstützt.

- **Option -r für das MobiLink-Dienstprogramm für die Dateiübertragung wurde entfernt** Die Option -r für das mlfiletransfer-Dienstprogramm wird nicht mehr unterstützt.
- **Erweiterte Optionen Memory (mem) und DownloadBufferSize (dbs) für dbmlsync wurden entfernt** Die erweiterten Optionen Memory (mem) und DownloadBufferSize (dbs) für dbmlsync werden nicht mehr unterstützt. Verwenden Sie stattdessen die Optionen CacheMin, CacheInit und CacheMax.
- **dbmlsync-Unterstützung für SQL-Passthrough wurde entfernt** Die SQL-Passthrough-Funktion wird für MobiLink-Clients nicht mehr unterstützt. Sie wurde durch die zentrale Administrationsfunktion für entfernte Datenbanken ersetzt. Siehe „[Zentrale Administration von entfernten Datenbanken](#)“ [*MobiLink - Serveradministration*].
- **Option -ss für mlsrv12 ist nicht mehr erforderlich** Vor Version 12 wurde die Option -ss für mlsrv12 verwendet, um den MobiLink-Server in einer Serverfarmumgebung auszuführen. Aufgrund der neuen Sperrenlogik für entfernte IDs, die redundante Synchronisationen verhindert, ist die Option -ss für MobiLink-Server, die in einer Serverfarm laufen, nicht mehr erforderlich. Die Option wurde daher entfernt. Bei der Verwendung von serverinitiiertem Synchronisation mit einer MobiLink-Serverfarm ist ein Arbiter erforderlich.

Hinweis

Die Ausführung des MobiLink-Servers in einer Serverfarm ist eine Funktion der Hochverfügbarkeitsoption von MobiLink, für die eine separate Lizenz erforderlich ist. Siehe „[Getrennt lizenzierbare Komponenten](#)“ [*SQL Anywhere 16 - Einführung*].

- **MobiLink Redirector wurde entfernt** Der Redirector ist nicht mehr verfügbar. Verwenden Sie stattdessen den Relay Server. Siehe „[Einführung in den Relay Server](#)“ [*Relay Server*].
- **Empfehlung für die Verwendung von Skriptversionen** Es wird dringend empfohlen, die erweiterte Option ScriptVersion nicht mehr zu verwenden. Verwenden Sie stattdessen die SCRIPT VERSION-Klausel der Anweisungen CREATE SYNCHRONIZATION SUBSCRIPTION und ALTER SYNCHRONIZATION SUBSCRIPTION. Siehe „[CREATE SYNCHRONIZATION SUBSCRIPTION-Anweisung \[MobiLink\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*] und „[ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung \[MobiLink\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Die Option -n für dbmlsync wird nicht mehr empfohlen** Diese Option wird nicht mehr empfohlen. Verwenden Sie stattdessen die dbmlsync-Option -s. Siehe „[dbmlsync-Option -s](#)“ [*MobiLink - Clientadministration*].
- **Die Option -u für dbmlsync wird nicht mehr empfohlen** Diese Option wird nicht mehr empfohlen. Verwenden Sie stattdessen die dbmlsync-Option -s. Siehe „[dbmlsync-Option -s](#)“ [*MobiLink - Clientadministration*].
- **Die Synchronisationsprofil-Option Publication wird nicht mehr empfohlen** Diese Option wird nicht mehr empfohlen. Verwenden Sie stattdessen die dbmlsync-Option -s. Siehe „[dbmlsync-Option -s](#)“ [*MobiLink - Clientadministration*].

- **Die Synchronisationsprofil-Option MLUser wird nicht mehr empfohlen** Diese Option ist nicht mehr empfohlen. Verwenden Sie stattdessen die dbmlsync-Option -s. Siehe „[dbmlsync-Option -s](#)“ [[MobiLink - Clientadministration](#)].
- **Die dbmlsync-Integrationskomponente wurde entfernt** Die dbmlsync-Integrationskomponente wurde entfernt. Verwenden Sie stattdessen die Programmierschnittstelle dbmlsync. Siehe „[Dbmlsync-API](#)“ [[MobiLink - Clientadministration](#)].
- **Der Modus der erzwungenen Konfliktlösung wird nicht mehr empfohlen** Der MobiLink-Server verwendet die erzwungene Konfliktlösung, wenn die Skripten upload_insert, upload_update und upload_delete nicht definiert sind. Diese Funktion wird nicht mehr empfohlen.
- **Konflikterkennung mit upload_update-Skripten wird nicht mehr empfohlen** Sie müssen entweder hochgeladene Updatekonflikte in Ihrem upload_update-Skript erkennen und lösen oder ein upload_fetch-Skript oder ein upload_fetch_column_conflict-Skript verwenden, um Konflikte zu erkennen. Es wird nicht mehr empfohlen, den MobiLink-Server die betroffenen Zeilen von Ihrem upload_update-Skript zählen zu lassen, um einen Konflikt zu erkennen, und dann die Konfliktlösungsskripten aufzurufen. Siehe „[Konflikterkennung mit upload_update-Skripten](#)“ [[MobiLink - Serveradministration](#)].
- **Verwendung von Fragezeichen in SQL-Skripten** Die Verwendung von einfachen Fragezeichen in MobiLink SQL-Skripten wird nicht mehr empfohlen. Verwenden Sie stattdessen benannte Parameter. Siehe [Benannte Skriptparameter](#) [[MobiLink - Serveradministration](#)].

Neue Funktionen in SQL Remote

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in SQL Remote in Version 12.0.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **SQL Remote unterstützt nun die Replikation von räumlichen Werten** SQL Remote unterstützt nun die Replikation von räumlichen Datentypen. Zusätzlich unterstützt SQL Remote die Replikation des neuen TIMESTAMP WITH TIMEZONE-Datentyps. Siehe „[TIMESTAMP WITH TIME ZONE-Datentyp](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **-g-Option dem Extraktionsdienstprogramm (dbxtract) hinzugefügt** Standardmäßig werden materialisierte Ansichten, für die MANUAL REFRESH festgelegt ist, beim Aktualisierungsvorgang nicht initialisiert. Sie können die Option -g mit dbxtract verwenden, um diese materialisierten Ansichten als Teil des Aktualisierungsprozesses zu initialisieren. Siehe „[Extraktionsdienstprogramm \(dbxtract\)](#)“ [[SQL Remote](#)].
- **SQL Remote zeigt jetzt eine Fehlermeldung an, wenn kein Publikationseigentümer definiert ist** Wenn SQL Remote eine Verbindung mit einer Datenbank herstellt, für die zwar entfernte oder konsolidierte Benutzer, aber kein Publikationseigentümer definiert sind, dann gibt SQL Remote eine Fehlermeldung aus, die darauf hinweist, dass kein Publikationseigentümer in der Datenbank festgelegt wurde.

Neue Funktionen von UltraLite

Nachstehend finden Sie eine Liste der neuen Funktionen von UltraLite, die in Version 12.0.0 eingeführt wurden. Hinweise zu den unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Allgemeine Funktionen

- **Unterstützung räumlicher Daten** Die folgenden Funktionen wurden zur Unterstützung der neuen Fähigkeiten für die Verarbeitung räumlicher Daten in UltraLite hinzugefügt:
 - **ST_Geometry-Datentyp** Der Datentyp ST_Geometry unterstützt Funktionen, die auf räumliche Werte angewendet werden können. Siehe „ST_GEOMETRY-Datentyp“ [*UltraLite - Datenbankverwaltung*].
 - **Funktionen für räumliche Daten** UltraLite stellt mehrere Funktionen zur Unterstützung der Verarbeitung räumlicher Daten bereit. Siehe „ST_GEOMETRY-Datentyp“ [*UltraLite - Datenbankverwaltung*].
- **Neue Verschlüsselungsbeispiele** Neue Codebeispiele wurden hinzugefügt, um die UltraLiteJ-Verschlüsselung auf BlackBerry-Smartphones zu veranschaulichen. Weitere Hinweise finden Sie im Verzeichnis %SQLANYSAMPI2%\UltraLiteJ.

Plattformen und Geräte

iPhone- und Mac OS X-Unterstützung

UltraLite-Anwendungen können jetzt unter Verwendung der UltraLite C/C++-API unter Mac OS X für Mac OS X und das iPhone entwickelt werden. Weitere Hinweise finden Sie unter „UltraLite C++-Anwendungsentwicklung“ [*UltraLite - C- und C++-Programmierung*]. Es gibt auch eine detaillierte praktische Einführung, die Sie bei der Entwicklung einer iPhone-Anwendung unterstützt.

Installation auf 64-Bit-Linux

Wenn Sie SQL Anywhere auf einem 64-Bit-Linux-Computer installieren, müssen Sie getrennt davon außerdem das 32-Bit-Subsystem installieren. Dies liegt daran, dass einige 64-Bit-Linux-Betriebssysteme keine vorinstallierten 32-Bit-Kompatibilitätsbibliotheken umfassen. Zur Verwendung von 32-Bit-Clientsoftware müssen Sie möglicherweise 32-Bit-Kompatibilitätsbibliotheken für Ihre Linux-Distribution installieren. Auf Ubuntu müssen Sie z.B. möglicherweise folgenden Befehl ausführen:

```
sudo apt-get install ia32-libs
```

Sicherheit

- **UltraLite unterstützt die Verschlüsselung mit FIPS 140-2-Zertifizierung** Die Verschlüsselung mit FIPS 140-2-Zertifizierung wird jetzt von UltraLite auf 64-Bit-Windows unterstützt.

- **Die UltraLite-Datenbankverschlüsselung verwendet jetzt 256-Bit AES** Die UltraLite-Datenbankverschlüsselung wird nun mit dem 256-Bit AES anstelle des 128-Bit AES durchgeführt.
- **Beim Verschlüsseln der Kennwörter wird jetzt die Salt-Methode verwendet** Ein Zufallswert mit 4 Byte Länge (Salt) wird jetzt jedes Mal generiert, wenn ein neuer Benutzer erstellt wird oder ein bestehender Benutzer sein Kennwort ändert. Siehe „[UltraLite-Verbindungsparameter PWD](#)“ [[UltraLite - Datenbankverwaltung](#)].

Dienstprogramme

- **Verbesserungen der UltraLite-Dienstprogramme** Das Dienstprogramm ulinit wurde erweitert, um eine UltraLite-Datenbank basierend auf Informationen in einer SQL Anywhere-Datenbank zu erstellen, auch wenn das aus der SQL Anywhere-Datenbank extrahierte Schema Elemente enthält, die UltraLite nicht unterstützt (wie beispielsweise Spaltentypen oder Standardwerte). Dies ist nun das Standardverhalten (das mit der Option -f ausgeschaltet werden kann). Siehe „[UltraLite-Dienstprogramm zum Initialisieren einer Datenbank \(ulinit\)](#)“ [[UltraLite - Datenbankverwaltung](#)]. Das ulcreate-Dienstprogramm wird nicht mehr unterstützt.

Änderungen wurden auch an den Dienstprogrammen ulinit, ulerase, ulinfo, ulload und ulsync vorgenommen.

Siehe:

- „[UltraLite-Dienstprogramm zum Initialisieren einer Datenbank \(ulinit\)](#)“ [[UltraLite - Datenbankverwaltung](#)]
- „[UltraLite-Dienstprogramm zum Löschen von Datenbanken \(ulerase\)](#)“ [[UltraLite - Datenbankverwaltung](#)]
- „[UltraLite-Informationsdienstprogramm \(ulinfo\)](#)“ [[UltraLite - Datenbankverwaltung](#)]
- „[UltraLite-Dienstprogramm zum Laden von Daten aus XML-Dateien \(ulload\)](#)“ [[UltraLite - Datenbankverwaltung](#)]
- „[UltraLite-Synchronisationsdienstprogramm \(ulsync\)](#)“ [[UltraLite - Datenbankverwaltung](#)]

SQL

- **CREATE / DROP / ALTER USER hinzugefügt** Siehe:
 - „[CREATE USER-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)]
 - „[DROP USER-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)]
 - „[ALTER USER-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)]
- **Neue Tabellenintegritätsregel für CREATE TABLE und ALTER TABLE** Eine zusätzliche Tabellenintegritätsregel (SYNCHRONIZE ON | OFF | ALL) kann in einer CREATE TABLE- oder ALTER TABLE-Anweisung festgelegt werden. Siehe:
 - „[CREATE TABLE-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)]
 - „[ALTER TABLE-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)]

- **IF EXISTS-Klausel für DROP-Anweisungen hinzugefügt** Die neue IF EXISTS-Klausel kann nun optional in DROP INDEX-, DROP PUBLICATION-, DROP SYNCHRONIZATION PROFILE- und DROP TABLE-Anweisungen angegeben werden. Siehe:
 - „DROP INDEX-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP PUBLICATION-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
- **IF NOT EXISTS-Klausel für CREATE-Anweisungen wurde hinzugefügt** Die neue IF NOT EXISTS-Klausel kann nun optional in CREATE TABLE-, CREATE INDEX-, CREATE PUBLICATION- und CREATE SYNCHRONIZATION PROFILE-Anweisungen angegeben werden. Siehe:
 - „CREATE TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „CREATE INDEX-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „CREATE PUBLICATION-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „CREATE SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
- **CREATE SYNCHRONIZATION PROFILE unterstützt jetzt auch OR REPLACE-Klausel** Die CREATE SYNCHRONIZATION PROFILE-Anweisung unterstützt jetzt auch die Option zum Ersetzen einer Tabelle. Siehe „CREATE SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)].
- **COUNT_UPLOAD_ROWS-Funktion hinzugefügt** COUNT_UPLOAD_ROWS ermöglicht die Abfrage der Anzahl der Zeilen, die in den Upload der nächsten Synchronisation einbezogen werden. Siehe „COUNT_UPLOAD_ROWS-Funktion [Aggregat]“ [[UltraLite - Datenbankverwaltung](#)].

Datentypen

- **ST_Geometry** Der Datentyp ST_Geometry unterstützt Funktionen, die auf räumliche Werte angewendet werden können. Siehe „ST_GEOMETRY-Datentyp“ [[UltraLite - Datenbankverwaltung](#)].
- **TIMESTAMP WITH TIMEZONE-Datentyp** Der Datentyp TIMESTAMP WITH TIMEZONE ermöglicht das gemeinsame Speichern von Datums- und Zeitwerten mit Zeitzonen-Offsets. Ein Zeitzonen-Offset ist die Anzahl der Minuten vor oder nach UTC. Siehe „[UltraLite-Dienstprogramm zum Initialisieren einer Datenbank \(ulinit\)](#)“ [[UltraLite - Datenbankverwaltung](#)] und „[UltraLite-Erstellungsparameter timestamp_with_time_zone_format](#)“ [[UltraLite - Datenbankverwaltung](#)].

Programmierschnittstellen

UltraLite C/C++

- **Neue UltraLite C/C++-API hinzugefügt** Eine neue Version der UltraLite C/C++-API wurde in dieser Version hinzugefügt. Diese API wird in der Header-Datei *ulcpp.h* deklariert. Siehe „[UltraLite C/C++-API-Referenz](#)“ [[UltraLite - C- und C++-Programmierung](#)].

Die folgenden Objekte wurden für diese Version der UltraLite C/C++-API umbenannt:

- `ul_synch_info` wurde umbenannt in `ul_sync_info`.
- `ul_synch_result` wurde umbenannt in `ul_sync_result`.
- `ul_synch_status` wurde umbenannt in `ul_sync_status`.
- `ul_synch_stats` wurde umbenannt in `ul_sync_stats`.
- `ul_synch_observer_fn` wurde umbenannt in `ul_sync_observer_fn`.
- `ul_synch_state` wurde umbenannt in `ul_sync_state`.
- `ULInitSynchInfo` wurde umbenannt in `ULInitSyncInfo`.
- `ULSetSynchInfo` wurde umbenannt in `ULSetSyncInfo`.
- `ULGetSynchResult` wurde umbenannt in `ULGetSyncResult`.
- Alle `UL_SYNCH_STATE`-Objekte wurden umbenannt in `UL_SYNC_STATE`-Objekte.
- `UL_SYNCH_STATUS_FLAG_IS_BLOCKING` wurde umbenannt in `UL_SYNC_STATUS_FLAG_IS_BLOCKING`.
- `ULRegisterErrorCallback` wurde umbenannt in `ULSetErrorCallback`.
- `ULRegisterSynchronizationCallback` wurde umbenannt in `ULSetSynchronizationCallback`.

Die folgenden Methoden wurden aus der C++-API entfernt: `GetDatabaseID`, `SetDatabaseID`, `IsCaseSensitive` und `GetCollationName`. Die Funktion wird nun von `GetDatabaseProperty` und `SetDatabaseOption` verarbeitet. Siehe [ULConnection.GetDatabaseProperty-Methode \[UltraLite C++\]](#) [*UltraLite - C- und C++-Programmierung*] und [ULConnection.SetDatabaseOption-Methode \[UltraLite C++\]](#) [*UltraLite - C- und C++-Programmierung*].

Die `GetDatabasePropertyInt`-Methode wurde hinzugefügt. Siehe [ULConnection.GetDatabasePropertyInt-Methode \[UltraLite C++\]](#) [*UltraLite - C- und C++-Programmierung*].

UltraLite.NET

- **Die neue Methode `ULConnection.ValidateDatabase(ULDBValid)` wurde hinzugefügt.** Diese Methode wurde als gleichwertige Alternative zum Aufruf `ULConnection.ValidateDatabase(ULDBValid, String)` hinzugefügt, wenn ein Nullwert für `tableName` übergeben wird. Siehe [ULConnection.ValidateDatabase\(ULDBValid\)-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].

UltraLite für M-Business Anywhere

- **API-Funktionen** Folgende Objekte wurden in dieser Version hinzugefügt:
 - Die Methode `setPublications` in der Klasse `SyncParms`. Diese Methode legt die Publikation fest, die synchronisiert werden soll.
 - Die Methode `getPublications` in der Klasse `SyncParms`. Diese Methode gibt die Publikation zurück, die synchronisiert werden soll.
 - Das Mitglied `timestampWithTimeZoneFormat` in der Klasse `CreationParms`. Dieses Mitglied legt das Format für den Zeitstempel mit der Zeitzone fest, die aus der Datenbank abgerufen wird.
 - Die Methode `setTimestampWithTimeZoneParameter` in der Klasse `PreparedStatement` class und die Methode `setTimestampWithTimeZone` method in der Klasse `ULTable`. Diese Methoden legen den

Wert für die angegebenen Parameter vom Typ `SQLType.TIMESTAMP_WITH_TIME_ZONE` mit einem Datumsobjekt fest.

- Die Methoden `getTimestampWithTimeZone` in den Klassen `ResultSet` und `ULTable`. Diese Methoden geben den Wert für die angegebenen Spalten als Datumsobjekt zurück.

UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

Nachstehend finden Sie eine Liste der nicht mehr empfohlenen Funktionen und Verhaltensänderungen von UltraLite, die in Version 12.0.0 eingeführt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Nicht mehr empfohlene Funktionen

- **ulcreate-Dienstprogramm wird nicht mehr unterstützt** Das `ulcreate`-Dienstprogramm ist nicht mehr verfügbar. Alle seine Funktionen werden nun von `ulinit` verarbeitet. Siehe „[UltraLite-Dienstprogramm zum Initialisieren einer Datenbank \(ulinit\)](#)“ [[UltraLite - Datenbankverwaltung](#)].
- **SQL-Passthrough wird nicht mehr empfohlen** SQL-Passthrough, ursprünglich in UltraLite 11 eingeführt, wird nicht mehr empfohlen. Diese Funktionen werden jetzt von der Funktion „[Zentrale Administration von entfernten Datenbanken](#)“ [[MobiLink - Serveradministration](#)] übernommen.
- **Option für ALTER SYNCHRONIZATION PROFILE mit OR REPLACE-Klausel wird nicht mehr unterstützt** Die `OR REPLACE`-Klausel wurde aus der `ALTER SYNCHRONIZATION PROFILE`-Anweisung entfernt.
- **UltraLite ODBC-API wird nicht mehr unterstützt** Die UltraLite ODBC-API wird nicht mehr unterstützt. Verwenden Sie stattdessen die UltraLite C/C++-API. Siehe [UltraLite - C- und C++-Programmierung](#).

Nicht mehr unterstützte Plattformen

- **UltraLite für M-Business Anywhere wird nicht mehr unterstützt** Die UltraLite-Unterstützung für M-Business Anywhere wird in UltraLite 12 nicht mehr empfohlen.
- **Das Palm-Betriebssystem wird nicht mehr unterstützt** Das Palm-Betriebssystem wird von UltraLite 12 nicht mehr unterstützt.

Entfernte, nicht mehr empfohlene und geänderte APIs

- **UltraLite C/C++-API ersetzt** Die UltraLite C/C++-API, die in der Headerdatei `uliface.h` definiert wird, wurde durch eine neue Version ersetzt, die in der Headerdatei `ulcpp.h` definiert wird. Die frühere

Version der API ist weiterhin verfügbar. Eine Dokumentation zur nicht mehr empfohlenen UltraLite C/C++-API finden Sie unter http://dcx.sybase.com/1101de/ulc_de11/c-common-apiref.html.

Sie können die alte Implementierung der UltraLite C/C++-API verwenden, indem Sie die Datei `%SQLANY11%\SDK\C\ulcpp11.cpp` Ihrem UltraLite-Anwendungsprojekt hinzufügen, wobei SQLANY11 eine Umgebungsvariable ist, die auf Ihr SQL Anywhere-Installationsverzeichnis zeigt.

- **Geänderte UltraLite C/C++-API-Objekte** Die folgenden Objekte wurden seit der letzten Version geändert und gelten für die neue UltraLite C/C++-API:
 - SQL-Passthrough wird von der API nicht mehr unterstützt. Die folgenden Objekte wurden entfernt:
 - `ul_sql_passthrough_state`
 - `ul_sql_passthrough_status`
 - `ul_sql_passthrough_observer_fn`
- **Geänderte gemeinsame UltraLite C/C++-API-Objekte** Die folgenden Objekte wurden seit der letzten Version geändert:
 - Die Funktion `MLFileTransfer` wurde in `MLFileDownload` umbenannt. Siehe [MLFileDownload-Methode \[UltraLite Embedded SQL\]](#) [*UltraLite - C- und C++-Programmierung*].
 - Das Feld `force_transfer` der Struktur `ml_file_transfer_info` wurde entfernt.
 - Das Feld `enable_resume` field von `ml_file_transfer_info` wird nun standardmäßig auf `TRUE` und nicht auf `FALSE` gesetzt. Siehe [MLFileDownload-Methode \[UltraLite Embedded SQL\]](#) [*UltraLite - C- und C++-Programmierung*].
 - `MLFileDownload` unterstützt das neue Feld `remote_key` von `ml_file_transfer_info`, das an MobiLink-Serverskripten übergeben wird, um eine bessere Kontrolle von Dateiübertragungen zu ermöglichen. Siehe [MLFileDownload-Methode \[UltraLite Embedded SQL\]](#) [*UltraLite - C- und C++-Programmierung*].
- **Embedded Visual C++ wird ab UltraLite 11.0 nicht mehr unterstützt** Die Unterstützung für Visual Studio 2003 endete mit UltraLite 11.0. Die Unterstützung für Embedded Visual C++ wurde daher nach Visual Studio 2005 verschoben.
- **Geänderte Embedded SQL-API-Objekte** Die folgenden Objekte wurden seit der letzten Version geändert:
 - SQL-Passthrough wird von der API nicht mehr unterstützt. Die folgenden Objekte wurden entfernt:
 - `ULGetSQLPassthroughScriptCount`
 - `UExecuteNextSQLPassthroughScript`
 - `UExecuteSQLPassthroughScripts`
 - `ULSetSQLPassthroughCallback`
- **Geänderte UltraLite für M-Business Anywhere-API-Objekte** Die folgenden Objekte wurden seit der letzten Version geändert:

- Die Methoden GetSQLPassthroughScriptCount, ExecuteNextSQLPassthroughScript und ExecuteSQLPassthroughScripts in der Connection-Klasse wurden entfernt.
- Die Syntax für die CreateDatabase-Methode in der DatabaseManager-Klasse hat sich geändert.
- **Geänderte UltraLite.NET-API-Objekte** Die folgenden Objekte wurden seit der letzten Version geändert:
 - Die DatabaseManager-Eigenschaft unter der ULConnection-Klasse wurde entfernt und ist nicht mehr erforderlich. Die ULDatabaseManager-Klasse ist kein Singleton mehr, die Methoden sind jetzt statisch. Siehe [ULDatabaseManager-Klasse \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
 - Die DatabaseOnCE-Eigenschaft der ULConnectionParms-Klasse wurde in DatabaseOnDevice umbenannt. Siehe [ULConnectionParms.DatabaseOnDevice-Eigenschaft \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
 - Die Methode GetOptimalIndex der Klasse ULTableSchema gibt jetzt den Namen des optimalen Index zurück. Siehe [ULTableSchema.GetOptimalIndex-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
 - Die Methode CountUploadRows(String, UInt32) der Klasse ULConnection wurde entfernt. Verwenden Sie stattdessen CountUploadRows(String, Int64). Siehe [ULConnection.CountUploadRows-Methode \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].
 - SQL-Passthrough wird von der API nicht mehr unterstützt. Die folgenden Objekte wurden entfernt:
 - ULConnection.GetSQLPassthroughScriptCount
 - ULConnection.ExecuteNextSQLPassthroughScript
 - ULConnection.ExecuteSQLPassthroughScripts
 - ULSqlPassthroughProgressListener
 - ULSqlProgressData
 - ULSqlProgressState
 - Die ULPublicationSchema-Klasse und ihre Methoden wurden zusammen mit der GetPublicationSchema-Methode aus der ULDatabaseSchema-Klasse entfernt. Die Felder SYNC_ALL_DB und SYNC_ALL_PUBS wurden in die ULConnection-Klasse verschoben. Siehe [ULConnection.SYNC_ALL_DB-Feld \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*] und [ULConnection.SYNC_ALL_PUBS-Feld \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].

Verschiedenes

- **Erhöhung der Begrenzung von Benutzerpublikationen** Die maximale Anzahl von Benutzerpublikationen wurde auf 63 erhöht.
- **Standardkodierung für UltraLite-Datenbanken ist nun UTF-8** UltraLite-Datenbanken werden nun standardmäßig mit UTF-8 kodiert. Siehe „[UltraLite-Erstellungsparameter utf8_encoding](#)“ [*UltraLite - Datenbankverwaltung*].

Neue Funktionen von UltraLiteJ

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in UltraLiteJ in Version 12.0.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Unterstützung für externe Blob-Dateien** UltraLiteJ unterstützt mit BlackBerry OS oder Java SE die Aufteilung von Datenbankdateien, sodass jetzt externe Dateien jetzt verwendet werden können, um große BLOB-Werte zu speichern, wobei die referenzierten Dateien bestimmte Spalten in der Datenbank verwenden. Dies ist als Teil der CREATE TABLE SQL-Funktion implementiert.

Ein einfaches Einsatzbeispiel: Eine BlackBerry-Anwendung nutzt den kleineren, aber schnelleren beständigen Speicher, um eine UltraLiteJ-Datenbank zu speichern, während große BLOB-Werte, beispielsweise Fotos, in größeren (aber langsameren) Flash-Speichern oder SD-Karten gespeichert werden. Ein zusätzlicher Vorteil besteht darin, dass Anwendungen, die Bilder erfassen und in der Datenbank speichern, keine Akkuleistung und Zeit verschwenden, indem sie die Bilder in die Datenbank kopieren.

Siehe „CREATE TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)].

- **Unterstützung für interne Flash-Speicher und SD-Karten für BlackBerry** UltraLiteJ kann UltraLiteJ-Datenbanken in den internen Flash-Speicher oder SD-Karten schreiben und davon lesen.
- **Unterstützung für mehrere Versionen von UltraLiteJ** Damit mehrere Versionen von UltraLiteJ in der BlackBerry- und Java ME-Umgebung nebeneinander existieren können, enthalten UltraLiteJ-JAR-Dateien COD-Dateien und der Java-Paketname enthält jetzt die Nummer der Hauptversion wie folgt:
 - Die JAR-Datei heißt *UltraLiteJ12.jar*
 - Die COD-Dateien heißen *UltraLiteJ12.cod*.
 - Alle öffentlichen Klassen sind im Paket `com.ianywhere.ultralitej12` enthalten
 - Die Klasse `Unsigned64` wurde in das Paket `com.ianywhere.ultralitej12` verlegt.
- **Verschlüsselung und sichere Ausführung von UltraLiteJ-Anwendungen** Eine neues Beispiel, das veranschaulicht, wie Sie eine sehr sichere UltraLiteJ-Anwendung für BlackBerry-Smartphones schreiben, wurde hinzugefügt. Weitere Hinweise finden Sie in *Samples\UltraLite\UltraLiteJ\BlackBerryEncryption\ReadMe.html*. Umfassendere Informationen zur BlackBerry-Sicherheit finden Sie im Whitepaper "UltraLiteJ Security on BlackBerry Devices" unter http://www.sybase.com/detail_list?id=9814.
- **Neuer Datentyp ST_Geometry** Der Datentyp `ST_Geometry` unterstützt Funktionen, die auf räumliche Werte angewendet werden können. Siehe „ST_GEOMETRY-Datentyp“ [[UltraLite - Datenbankverwaltung](#)].
- **TIMESTAMP WITH TIMEZONE-Datentyp** Der Datentyp `TIMESTAMP WITH TIMEZONE` ermöglicht das gemeinsame Speichern von Datums- und Zeitwerten mit Zeitzonen-Offsets. Ein Zeitzonen-Offset ist die Anzahl der Minuten vor oder nach UTC. Siehe „UltraLite-Dienstprogramm zum Initialisieren einer Datenbank (ulinit)“ [[UltraLite - Datenbankverwaltung](#)] und „UltraLite-Erstellungsparameter `timestamp_with_time_zone_format`“ [[UltraLite - Datenbankverwaltung](#)].

- **Unterstützung für die SQL-Funktion RAND** UltraLiteJ unterstützt die SQL-Funktion RAND. Siehe „RAND-Funktion [Numerisch]“ [[UltraLite - Datenbankverwaltung](#)].
- **Unterstützung für CREATE SYNCHRONIZATION PROFILE, ALTER SYNCHRONIZATION PROFILE, DROP SYNCHRONIZATION PROFILE und SYNCHRONIZE** Diese Anweisungen sollen eine alternative Möglichkeit zur Organisation von Synchronisationsparametern und zum Start von Synchronisationen mit SQL ermöglichen. Die bestehenden Objekte Connection.createSyncParm(), Connection.synchronize(SyncParm) und die entsprechenden APIs funktionieren weiterhin. Siehe:
 - „CREATE SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „ALTER SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „SYNCHRONIZE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
- **Neue Tabellenintegritätsregel für CREATE TABLE und ALTER TABLE** Eine zusätzliche Tabellenintegritätsregel (SYNCHRONIZE ON | OFF | ALL) kann in einer CREATE TABLE- oder ALTER TABLE-Anweisung festgelegt werden. Siehe:
 - „CREATE TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „ALTER TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
- **IF EXISTS-Klausel wurde hinzugefügt** Die neue IF EXISTS-Klausel kann nun optional in DROP INDEX-, DROP PUBLICATION-, DROP SYNCHRONIZATION PROFILE- und DROP TABLE-Anweisungen angegeben werden. Siehe:
 - „DROP INDEX-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP PUBLICATION-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „DROP TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
- **IF NOT EXISTS-Klausel wurde hinzugefügt** Die neue IF NOT EXISTS-Klausel kann nun optional in CREATE TABLE-, CREATE INDEX-, CREATE PUBLICATION- und CREATE SYNCHRONIZATION PROFILE-Anweisungen angegeben werden. Siehe:
 - „CREATE TABLE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „CREATE INDEX-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „CREATE PUBLICATION-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „CREATE SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
- **CREATE SYNCHRONIZATION PROFILE unterstützt jetzt auch OR REPLACE-Klausel** Die CREATE SYNCHRONIZATION PROFILE-Anweisung unterstützt jetzt auch die Option zum Ersetzen einer Tabelle. Siehe „CREATE SYNCHRONIZATION PROFILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)].

- **Dateiübertragung durch MobiLink** UltraLiteJ kann Uploads und Downloads von Dateien in die entfernte Datenbank über den MobiLink-Server vornehmen.

Die Desktopversion von UltraLiteJ kann Downloads aller Arten von Dateien aus MobiLink in das lokale Dateisystem oder Uploads aller Arten von Dateien aus dem lokalen Dateisystem in MobiLink vornehmen.

Die BlackBerry OS -Version von UltraLiteJ kann einen Download von gültigen, nicht verschlüsselten, nicht verschleierte Datenbankdateien aus MobiLink vornehmen und in einem Objektspeicher speichern, oder einen Upload dieser Typen von Datenbankdateien auf MobiLink vornehmen. Der Download aller Arten von Dateien aus MobiLink auf die Datenträgerkarte oder in den Flash-Speicher sowie der Upload aller Arten von Dateien aus dem Flash-Speicher und von Datenträgerkarten in MobiLink ist möglich.

Weitere Hinweise zu UltraLiteJ-Dateiübertragungen finden Sie unter [FileTransfer-Schnittstelle \[UltraLiteJ\]](#) [[UltraLite® – Java-Programmierung](#)].

- **Dienstprogramm UltraLiteJ-Datenbankübertragung kann Datenbanken im BlackBerry-Dateisystem öffnen und übertragen** UltraLiteJ entscheidet automatisch über den Standort der Datenbank (Objektspeicher oder Dateisystem) basierend auf dem Datenbanknamen. Wenn der Name mit *file://* (Groß-/Kleinschreibung beachten) beginnt, versucht das Dienstprogramm, die Datenbank im Dateisystem zu finden, sonst im Objektspeicher.
- **BlackBerry-Installationsverzeichnis umbenannt** Das Installationsverzeichnis für BlackBerry-Smartphone-Dateien wurde umbenannt, um die Mindest-Version des BlackBerry-Betriebssystems anzugeben. Das Verzeichnis *UltraLite\UltraLiteJ\Java MERIM11* heißt nun *UltraLite\UltraLiteJ\BlackBerry4.2*, um anzuzeigen, dass Dateien mit BlackBerry OS 4.2 und höher kompatibel sind.
- **Blob-Dateien für ULjLoad und ULjUnload unterstützt** Die Lade- und Entlade-Dienstprogramme von UltraLiteJ unterstützen benutzerdefinierte Implementierungen des blobfile-Datentyps. Eine Musterimplementierung von blobfile-Dateitypen in einer UltraLiteJ-Datenbank finden Sie unter „[Entlade-Dienstprogramm für UltraLite Java Edition-Datenbanken \(uljunload\)](#)“ [[UltraLite - Datenbankverwaltung](#)] und „[Lade-Dienstprogramm für UltraLite Java Edition-Datenbank \(uljload\)](#)“ [[UltraLite - Datenbankverwaltung](#)].
- **Verbesserter Zeilenbegrenzungsalgorithmus** Der Zeilenbegrenzungsalgorithmus wurde verbessert und berücksichtigt nun, dass Zeilen aus Tabellen mit vielen Spalten mehr Ressourcen verwenden als Zeilen aus Tabellen mit wenigen Spalten.
- **Änderungen der Systemtabelle** Die folgenden Änderungen wurden in den UltraLiteJ-Systemtabellen vorgenommen:
 - **column_default_value-Spalte in syscolumn-Systemtabelle unterstützt die DEFAULT AUTOFILENAME-Standardklausel** Diese Spalte kann VARCHAR-Typen verarbeiten, die mit dem Spaltenstandardwert DEFAULT AUTOFILENAME angegeben werden. Siehe „[syscolumn-Systemtabelle](#)“ [[UltraLite - Datenbankverwaltung](#)].
 - **Spalte filename_colid der syscolumn-Systemtabelle hinzugefügt** Diese Spalte speichert die Spalten-ID der referenzierten file_name-Spalte in der Schemadefinition. Sonst hat diese Spalte einen Nullwert. Siehe „[syscolumn-Systemtabelle](#)“ [[UltraLite - Datenbankverwaltung](#)].

- **Spalte `table_partition_size` der systable-Systemtabelle hinzugefügt** Diese Spalte speichert den definierten Partitionsgrößenwert. Siehe „systable-Systemtabelle“ [[UltraLite - Datenbankverwaltung](#)].
- **Änderungen der Verschlüsselungsperformance** Die EncryptionControl-Schnittstelle wurde wie folgt verbessert, um die Performance in langsamen CPU-Umgebungen zu steigern:
 - UltraLiteJ verschlüsselt nun nur daten- und systemkritische Seiten.
 - Die decrypt-Methode akzeptiert nun einen zusätzlichen Parameter, um die Anzahl der Byte festzulegen, die entschlüsselt werden müssen.

Neue UltraLiteJ-API-Funktionen

- **API-Funktionen** Folgende Objekte wurden in dieser Version hinzugefügt:
 - Die ConfigFileME-Schnittstelle wurde hinzugefügt, um Methoden bereitzustellen, mit denen Konfigurationen für beständige Datenbanken in einer Datei im Dateisystem eines Java ME-Geräts gespeichert werden können.
 - Die Methoden `setRowScoreFlushSize` und `setRowScoreMaximum` in der Schnittstelle `ConfigPersistent` wurden hinzugefügt, um die Methoden `setRowMinimumThreshold` und `setRowMaximumThreshold` zu ersetzen. Diese Methoden setzen den neuen Zeilenbegrenzungsalgorithmus um. Der neue Zeilenbegrenzungsalgorithmus verbessert das Ressourcenmanagement von Tabellen mit vielen Spalten. Siehe [ConfigPersistent.setRowScoreFlushSize-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#) und [ConfigPersistent.setRowScoreMaximum-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

Die UUIDValue-Schnittstelle wurde hinzugefügt, um einen eindeutigen Bezeichner zu beschreiben. Siehe [UUIDValue-Schnittstelle \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
 - Die `getLastIdentity`-Methode in der Connection-Schnittstelle wurde hinzugefügt, um den neuesten Wert abzurufen, der über die aktuelle Verbindung in eine DEFAULT AUTOINCREMENT- oder DEFAULT GLOBAL AUTOINCREMENT-Spalte eingefügt wurde. Siehe [Connection.getLastIdentity-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
 - Die Methode `getSyncObserver` in der Schnittstelle `Connection` wurde hinzugefügt, um das aktuell registrierte SyncObserver-Objekt für die Verbindung zurückzugeben. Siehe [Connection.getSyncObserver-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
 - Die Methode `getSyncResult` in der Schnittstelle `Connection` wurde hinzugefügt, um das Ergebnis der letzten SYNCHRONIZE SQL-Anweisung zurückzugeben, die in der Verbindung ausgeführt wurde. Siehe [Connection.getSyncResult-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
 - Die Methode `setSyncObserver` in der Schnittstelle `Connection` wurde hinzugefügt, um ein SyncObserver-Objekt einzurichten, das den Fortschritt von Synchronisationen über eine Verbindung beobachtet. Siehe [Connection.setSyncObserver-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

- Die Methoden `createFileTransfer` und `createObjectStoreTransfer` in der Schnittstelle `DatabaseManager` wurden hinzugefügt, um `FileTransfer`-Objekte zu erstellen, die verwendet werden können, um Datenbankdateien zu übertragen. Siehe [DatabaseManager.createFileTransfer-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#) und [DatabaseManager.createObjectStoreTransfer-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- Die Schnittstelle `FileTransfer` wurde hinzugefügt, um Methoden bereitzustellen, die die Optionen für eine Dateiübertragung festlegen. Siehe [FileTransfer-Schnittstelle \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- Die Schnittstelle `FileTransferProgressData` wurde hinzugefügt, damit Daten an den Observer-Callback übergeben werden können. Siehe [FileTransferProgressData-Schnittstelle \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- Die Schnittstelle `FileTransferProgressListener` wurde als Observer-Objekt für die Dateiübertragung hinzugefügt. Siehe [FileTransferProgressListener-Schnittstelle \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- Die Methode `getPlanTree` in der Schnittstelle `PreparedStatement` wurde hinzugefügt, um den Plan für eine Abfrage in lesbarer Form zu präsentieren, wenn er auf einem Computerbildschirm angezeigt oder gedruckt wird. Siehe [PreparedStatement.getPlanTree-Methode \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- Die `hasShadowPaging`-Methode in der `ConfigPersistent`-Schnittstelle wurde hinzugefügt, um zu ermitteln, ob Shadow Paging aktiviert ist. Siehe [ConfigPersistent.hasShadowPaging-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).

Verhaltensänderungen von UltraLiteJ und nicht mehr empfohlene Funktionen

Nachstehend finden Sie eine Liste der nicht mehr empfohlenen Funktionen und Verhaltensänderungen von UltraLiteJ, die in Version 12.0.0 eingeführt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

Entfernte, nicht mehr empfohlene und geänderte Dienstprogrammoptionen

- **-p-Option für `ULjInfo`, `ULjLoad` und `ULjUnload` ist jetzt optional** `dba` wird als Standardkennwort benutzt, wenn die `-p`-Option nicht angegeben ist.

Nicht mehr unterstützte Plattformen

- **UltraLiteJ unterstützt BlackBerry OS 4.1 nicht mehr** UltraLiteJ unterstützt BlackBerry OS 4.2 oder höher.

Entfernte, nicht mehr empfohlene und geänderte APIs

- **Geänderte UltraLiteJ-API-Objekte** Die folgenden Objekte wurden seit der letzten Version geändert:

- Der anywhere.ultralitej-Paketname wurde geändert. Der Paketname enthält jetzt die Versionsnummer, sodass mehrere Versionen von UltraLiteJ in der BlackBerry- und der Java ME-Umgebung sowie UltraLiteJ-JAR-Dateien und COD-Dateien nebeneinander existieren können. Siehe „UltraLiteJ-API-Referenz“ [[UltraLite® – Java-Programmierung](#)].
- Die Spalte table_autoinc in der UltraLiteJ syscolumnn-Systemtabelle wurde entfernt. Siehe „syscolumnn-Systemtabelle“ [[UltraLite - Datenbankverwaltung](#)].
- Die Schnittstelle CollectionOfValueReaders wurde entfernt. Alle CollectionOfValueReader-Methoden wurden in die Schnittstelle ResultSet verlegt. Siehe [ResultSet-Schnittstelle](#) [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)].
- Die Schnittstelle CollectionOfValueWriters wurde entfernt. Alle CollectionOfValueWriter-Methoden wurden in die Schnittstelle PreparedStatement verlegt. Siehe [PreparedStatement-Schnittstelle](#) [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)].
- Die ConfigPersistent-Methoden setRowMaximumThreshold und setRowMinimumThreshold wurden durch setRowScoreFlushSize und setRowScoreMaximum ersetzt. Siehe [ConfigPersistent.setRowScoreFlushSize-Methode](#) [[BlackBerry](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)] und [ConfigPersistent.setRowScoreMaximum-Methode](#) [[BlackBerry](#)] [[UltraLiteJ](#)] [[UltraLite® – Java-Programmierung](#)].
- Alle schemabezogenen Methoden und Schnittstellen wurden entfernt. Die Schnittstelle ForeignKeySchema und alle Methoden in den Schnittstellen ColumnSchema, IndexSchema und TableSchema wurde entfernt. Die folgenden Methoden wurden aus der Schnittstelle Connection entfernt:
 - createPublication
 - createTable
 - dropForeignKey
 - dropPublication
 - dropTable
 - enableSynchronization
 - renameTable
 - schemaCreateBegin
 - schemaCreateComplete
 - setNeverSynchronized
 - startSynchronizationDelete
 - stopSynchronizationDelete
 - truncateTable

Verwenden Sie von UltraLiteJ unterstützte SQL-Anweisungen wie CREATE TABLE und CREATE PUBLICATION, um Schemavorgänge vorzunehmen. Siehe „[UltraLite-SQL-Anweisungen](#)“ [[UltraLite - Datenbankverwaltung](#)].

- Die Methoden setDatabaseId und getDatabasePartitionSize der Schnittstelle Connection wurden entfernt. Eine Standardpartitionsgröße kann nicht mehr festgelegt werden. Verwenden Sie die DEFAULT GLOBAL AUTOINCREMENT-Anweisung zum Überschreiben der Standard-Partitionsgröße. Siehe „[GLOBAL AUTOINCREMENT-Spalten in UltraLite deklarieren](#)“ [[UltraLite - Datenbankverwaltung](#)].

- Die Methode `getDatabaseID` der Schnittstelle `Connection` hat jetzt dieselbe Wirkung wie der Aufruf von `Connection.getOption(OPTION_DATABASE_ID)`. Siehe [Connection.getOption-Methode \[BlackBerry\] \[UltraLiteJ\] \[UltraLite® – Java-Programmierung\]](#).
- Die Klassen `Value`, `ValueReader` und `ValueWriter` wurden entfernt.

Verschiedenes

- **Erhöhung der Begrenzung von Benutzerpublikationen** Die maximale Anzahl von Benutzerpublikationen wurde auf 63 erhöht.

Neue Funktionen der Administrationstools

Im Folgenden finden Sie eine Liste der neuen Administrationstools, die in SQL Remote in Version 12.0.0 hinzugefügt wurden. Hinweise zu unterstützten Plattformen und Versionen finden Sie unter <http://www.sybase.com/detail?id=1061806>.

- **Fenster "Verbinden"** Das Layout des Fensters **Verbinden** wurde in Version 12.0.0 optimiert und der **Verbindungsassistent** wurde entfernt. Früher mussten Sie wissen, welche Optionen für den gewünschte Verbindungstyp erforderlich sind. Jetzt wählen Sie den Verbindungstyp und das Fenster **Verbinden** zeigt Ihnen die Optionen, die für den angegebenen Verbindungstyp gelten.

Vorsicht

- Sie müssen eine der folgenden Verbindungstypen aus der Dropdown-Liste **Aktion** wählen:

Mit einer laufenden Datenbank auf diesem Computer verbinden Stellt eine Verbindung mit einer Datenbank her, die bereits die auf Ihrem Computer läuft.

Mit einer ODBC-Datenquelle verbinden Stellt eine Verbindung mit einer Datenbank über eine ODBC-Datenquelle her.

Mit einer laufenden Datenbank auf einem anderen Computer verbinden Stellt eine Verbindung mit einer Datenbank her, die auf einem anderen Computer im Netzwerk ausgeführt wird.

Eine Datenbank auf diesem Computer starten und eine Verbindung herstellen Startet eine Datenbank auf diesem Computer und stellt eine Verbindung zu ihr her.

Eine Datenbank auf einem anderen Computer starten und eine Verbindung herstellen Startet eine Datenbank über ein Netzwerk auf einem anderen Computer und stellt eine Verbindung zu ihr her.

Mithilfe einer Verbindungszeichenfolge verbinden Stellt mithilfe einer Verbindungszeichenfolge eine Verbindung zu einer Datenbank her.

Die Optionen unter der Dropdown-Liste **Aktion** ändern sich je nach Ihrer Auswahl.

- In Interactive SQL klicken Sie unter dem Titel **Mit einer SQL Anywhere-Datenbank verbinden** auf **Datenbanktyp ändern**, um den Typ der Datenbank zu ändern, mit der Sie sich verbinden.

Beispiel: Wenn Sie sich mit einer Datenbank über eine ODBC-Datenquelle verbinden, zeigt das Fenster **Verbinden** nur zwei Optionen an: **ODBC-Datenquellenname** und **ODBC-Datenquellendatei**.

Wenn erforderlich, können Sie auf **Erweitert** klicken, um Optionen wie TCP/IP, Verschlüsselung und andere erweiterte Optionen einzugeben. Siehe „Das Fenster Verbinden öffnen (Sybase Central)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Neuer Standardwert für die Prüfung nach Softwareaktualisierungen** Interactive SQL, Sybase Central, das SQL Anywhere-Konsolendienstprogramm und der MobiLink-Monitor prüfen nun standardmäßig täglich auf Aktualisierungen. In früheren Versionen der Software war die Standardvorgabe, dass diese Prüfung nie stattfand. Siehe „Software-Updates“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Bilder, HTML und XML in Interactive SQL und Sybase Central anzeigen** Sie können Bilder, HTML und XML in einer Ergebnismenge anzeigen. Siehe „Bilder und SVGs in Interactive SQL anzeigen“ [[SQL Anywhere Server - Datenbankadministration](#)] und „HTML- und XML-Daten in Interactive SQL anzeigen“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue Symbole in der Taskleiste für Sybase Central und Interactive SQL.** Wenn die Schnelllader-Option von Sybase Central oder Interactive SQL aktiviert ist, erscheint ein Symbol in der Taskleiste. Rechtsklicken Sie auf das Symbol und klicken Sie dann auf **Öffnen**, um die Anwendung zu öffnen, oder klicken Sie auf **Beenden**, um die Anwendung zu schließen (wenn sie läuft) und den Schnelllader-Prozess zu beenden. Siehe „Schnelllader-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Option zur Aktivierung von Eingabehilfen** Die Option für die Aktivierung der Eingabehilfen ist nun standardmäßig installiert. Früher musste diese Option getrennt installiert werden. Die Option "Aktivierung der Eingabehilfen" ermöglicht es, dass Interactive SQL, Sybase Central, das SQL Anywhere-Konsolendienstprogramm und der MobiLink Monitor mit Eingabehilfen wie Bildschirmlesern arbeiten können. Siehe „Eingabehilfen mit Java Access Bridge“ [[SQL Anywhere 16 - Einführung](#)].

Neue Plug-In-Funktionen in Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen der Sybase Central-Plug-Ins in Version 12.0.0.

Neue Funktionen der SQL Anywhere-Plug-Ins

- **Die von Assistenten erstellten SQL-Anweisungen und Dienstprogrammbefehle anzeigen** Die meisten SQL Anywhere-Plug-In-Assistenten, die ein Datenbankobjekt erstellen oder ein Datenbankdienstprogramm ausführen, werden mit einer zusätzlichen Seite am Ende versehen. Diese Seite zeigt die SQL-Anweisungen und -Dienstprogrammbefehle an, die ausgeführt werden, wenn Sie auf **Fertig stellen** klicken. Mit dem Klicken auf **Fertig stellen** werden die SQL-Anweisungen bzw. die Dienstprogrammbefehle ausgeführt.

Sie können die SQL-Anweisungen auch in die Zwischenablage kopieren, auf **Abbrechen** klicken, um den Assistenten zu verlassen, und danach die SQL-Anweisungen über Interactive SQL ausführen. Mit dieser Funktion können Sie die Assistenten verwenden, um SQL-Skripten ohne Änderung der

Datenbank zu erstellen. Siehe [Von Assistenten generierte SQL-Anweisungen und Dienstprogrammbeefehle anzeigen \[SQL Anywhere Server - Datenbankadministration\]](#).

- **Unterstützung für räumliche Daten** Hinweise über neue Plug-In-Funktionen, einschließlich Assistenten, im Zusammenhang mit räumlichen Daten, finden Sie unter [Unterstützung räumlicher Daten auf Seite 119](#).
- **Neue Registerkarte Fragmentierung für Datenbanken** Sie können eine grafische Darstellung der Fragmentierung von Basistabellen und Indizes anzeigen. Die Registerkarte **Fragmentierung** zeigt eine grafische Darstellung der Ergebnisse der Systemprozedur `sa_table_fragmentation` für Basistabellen. Siehe „Die Registerkarte Fragmentierung (SQL Anywhere-Plug-In)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Protokollierung von Datenbankänderungen** Sie können alle SQL-Anweisungen protokollieren, die vom SQL Anywhere-Plug-In ausgeführt werden und die Datenbank ändern. Das Log wird in einer `.sql`-Datei gespeichert. Siehe „SQL-Anweisungen protokollieren“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung für WITH NULLS NOT DISTINCT-Klausel, CREATE INDEX-Anweisung** Sie können mit dem SQL Anywhere-Plug-In Indizes erstellen, anzeigen und ändern, die die WITH NULLS NOT DISTINCT-Klausel verwenden. Der Ordner "Indizes" enthält die Spalte **Nullwerte distinct**. Der Wert in der Spalte ist ein Leerzeichen, wenn der Index ein Primärschlüssel, ein Fremdschlüssel, eine Eindeutigkeits-Integritätsregel oder ein nicht eindeutiger Index ist.

Weitere Hinweise finden Sie in der Beschreibung der UNIQUE-Klausel für die „CREATE INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Begriffsegmentierer** Das SQL Anywhere-Plug-In unterstützt externe Begriffsegmentierer und Vorfilter für Textkonfigurationsobjekte. Siehe „ALTER TEXT CONFIGURATION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Verbesserungen der Datenbankdokumentation** Wenn Sie den **Assistenten zum Erstellen der Datenbankdokumentation** verwenden, enthält die erstellte Dokumentation Informationen über Tabellen. Die Tabelleninformationen umfassen den Namen des Eigentümers, die Prozeduren, die die Tabelle verändern, Spalteninformationen und Kommentare. Siehe „Datenbank dokumentieren“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verbesserungen im Fenster Typfilter konfigurieren** Im Fenster **Typfilter konfigurieren** können Sie festlegen, welche Datenbankobjekte in der Ordnerliste im linken Fensterausschnitt von Sybase Central erscheinen. Sie können Ihre Einstellungen als Standard-Typfilter festlegen. Sybase Central benutzt den Standardfilter, wenn eine Verbindung zu einer Datenbank hergestellt wird, für die noch kein Typfilter festgelegt wurde. Siehe „Sybase Central-Navigation“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verbesserungen beim Assistenten zum Erstellen von Wartungsplänen** Wartungspläne enthalten benutzerdefinierte Vorgänge. Im **Assistenten zum Erstellen von Wartungsplänen** können Sie benutzerdefinierte Vorgänge als SQL-Anweisungen hinzufügen, die vor der Validierung oder nach der Sicherung ausgeführt werden.

Außerdem enthält der Ordner **Wartungspläne** jetzt eine Spalte **Status**. Siehe „Wartungspläne“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Verbesserungen beim Assistenten zum Erstellen von Funktionen** Bisher beschränkte der **Assistent zum Erstellen von Funktionen** den Rückgabebetyp auf einen integrierten Typ. Jetzt haben Sie im Assistenten die Wahl zwischen einem integrierten Typ oder einer Domäne. Siehe „Benutzerdefinierte Funktionen erstellen“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Verbesserungen beim Assistenten zum Sichern einer Datenbank** Standardmäßig aktiviert der **Assistent zum Sichern einer Datenbank** nun die Eliminierung freier Seiten. Siehe „Archiv Sicherungen erstellen (Sybase Central)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verbesserungen der Registerkarte Übersicht** Die Registerkarte **Übersicht** einer Datenbank zeigt Informationen über den Zustand und Statistiken von Kopieservern gemeinsam mit Informationen über die Datenbankspiegelung. Siehe „Zustand und Statistik von Datenbanken“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verbesserungen beim Fenster Eigenschaften der Synchronisationssubskription** Für Datenbanken, die mit SQL Anywhere 12 und höher erstellt wurden, erscheint der Name der Synchronisationssubskription im Fenster **Eigenschaften der Synchronisationssubskription** und auf der Registerkarte **Synchronisationssubskriptionen**. Für Datenbanken, die mit früheren Versionen von SQL Anywhere erstellt wurden, erscheint der Namen als (**unnamed**). Siehe „Erstellen von Synchronisationssubskriptionen“ [[MobiLink - Clientadministration](#)].
- **Neues Fenster Mithilfe des Synchronisationsprofils synchronisieren** Sie können eine SQL Anywhere-Datenbank synchronisieren, indem Sie mit der rechten Maustaste auf ein Synchronisationsprofil klicken und dann auf **Synchronisieren** und **OK** klicken. Siehe „MobiLink-Synchronisationsprofile“ [[MobiLink - Clientadministration](#)].

Verhaltensänderungen von Sybase Central und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Sybase Central in Version 12.0.0.

- **Sybase Central-Version jetzt 6.1.0** SQL Anywhere 12 enthält die Version 6.1.0 von Sybase Central 6.1.0.
- **Sybase Central unterstützt nur Datenbanken der SQL Anywhere-Versionen 10x und höher** Die Unterstützung für Datenbankserver der Version 9 und für Datenbanken, die mit der Version 9 erstellt wurden, wurde aus dem SQL Anywhere-Plug-In entfernt. Wenn Sie die Datenbank entladen und in eine Reload-Datei oder in eine neue oder vorhandene Datenbank neu laden, können Sie dennoch eine Verbindung mit einer Datenbank herstellen, die mit Version 5, 6, 7, 8 oder 9 erstellt wurde und auf einem Datenbankserver der Version 9 oder höher ausgeführt wird.

Bei Datenbanken, die mit Version 5, 6, 7, 8 oder 9 erstellt wurden und auf einem Datenbankserver der Version 6, 7, 8 oder 9 Datenbankserver ausgeführt werden, können Sie sich mit der Datenbank über das SQL Anywhere-Plug-In temporär verbinden, um eine der folgenden Aufgaben auszuführen:

- Die Datenbank in eine Reload-Datei entladen.
- Die alte Datenbank entladen und in eine Datenbank der Version 12 neu laden.
- Die Datenbank entladen und in eine bestehende Datenbank der Version 12 neu laden.

Die Dateien für die Datenbank, die entladen wird, müssen sich auf dem lokalen Computer befinden.

Das SQL Anywhere-Plug-In bietet keine Unterstützung für Datenbanken, die mit Version 4 oder früher erstellt wurden und auf einem Server der Version 5 oder früher ausgeführt werden.

Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

- **Unterstützung für 32- und 64-Bit-Computer** Die Konfigurationsdatei von Sybase Central wurde umbenannt:
 - Auf 32-Bit-Computern hat die Datei `.scRepository600` jetzt die Bezeichnung `.scRepository610_32`.
 - Auf 64-Bit-Computern kann die Datei `.scRepository600` den Namen `.scRepository610_32` und/oder `.scRepository610_64` haben, je nach Ihrer Installation.
- **Beim Deployment von Sybase Central brauchen Sie keine JPR-Dateien für die Plug-Ins mehr zu erstellen.** Bisher mussten Sie beim Deployment von Sybase Central und SQL Anywhere die JPR-Dateien für jedes Plug-In erstellen. Jetzt verwendet Sybase Central die Umgebungsvariable `%SQLANY12%`, um die Plug-Ins zu finden und zu registrieren.

Geänderte Funktionen des SQL Anywhere-Plug-Ins

- **Erweiterungen des Wartungsplans** Die folgenden Erweiterungen wurden für Wartungspläne vorgenommen:
 - Sie können SQL-Anweisungen einbeziehen, die vor der Validierung ausgeführt werden.
 - Sie können SQL-Anweisungen einbeziehen, die nach der Sicherung ausgeführt werden.
 - Sie können den Status des Wartungsplans anzeigen, während er läuft.
 - Sie können jeweils nur einen Wartungsplan gleichzeitig ausführen.

Weitere Hinweise finden Sie unter „[Wartungspläne](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Dynamische Objekte und Eigenschaften automatisch aktualisieren** Die Anzahl von Verbindungen und Sperren sowie die dynamischen Eigenschaften für Ereignisse, Wartungspläne und Windows-Dienste werden standardmäßig alle 10 Sekunden automatisch aktualisiert. Siehe „[Einstellen der Aktualisierungsfrequenz](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neuer Standardwert für den Assistenten zum Erstellen einer Datenbank** Wenn Sie mit dem **Assistenten zum Erstellen einer Datenbank** eine neue Datenbank der Version 12 erstellen, sind nun globale Prüfsummen standardmäßig aktiviert. Wenn Sie mit dem Assistenten zur Erstellung einer Datenbank eine Datenbank der Version 11 oder höher erstellen, sind nun globale Prüfsummen standardmäßig deaktiviert. Globale Prüfsummen sind immer standardmäßig aktiviert, wenn Datenbanken für Windows Mobile erstellt werden. Siehe [Prüfsummen sind standardmäßig für neue Datenbanken aktiviert](#) auf Seite 148.

Neue Funktionen von Interactive SQL

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Interactive SQL in Version 12.0.0 hinzugefügt wurden.

- **Räumliche Daten mit der Räumlichen Vorschau und dem Spatial Viewer anzeigen** Siehe „Anzeigen von räumlichen Daten als Bilder (Interactive SQL)“ [[SQL Anywhere Server - Unterstützung für räumliche Daten](#)].

- **Neue Möglichkeiten zum Ausführen der Anweisungen COMMIT und ROLLBACK** In Interactive SQL können Sie auf **SQL→Festschreiben** klicken, um eine COMMIT-Anweisung auszuführen, und auf **SQL→Zurücksetzen**, um eine ROLLBACK-Anweisung auszuführen. Sie können auch Tastenkürzel verwenden: Strg+Umsch+C für eine COMMIT-Anweisung und Strg+Umsch+R für eine ROLLBACK-Anweisung.

Durch das Ausführen einer COMMIT- oder ROLLBACK-Anweisung über das **SQL**-Menü oder ein Tastenkürzel werden die Inhalte des Fensterausschnitts **SQL-Anweisungen** nicht geändert. Die Registerkarte **Ergebnisse** im Fensterausschnitt **Ergebnisse** wird aber bereinigt. Siehe „Tastenkürzel für Interactive SQL“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Änderungen bei der Auswahl und beim Kopieren von Spalten, Zeilen und Zellen aus den Ergebnismengen** Im Fensterausschnitt **Ergebnisse** von Interactive SQL können Sie mehrere Spalten, Zeilen und Zellen in einer Ergebnismenge auswählen und dann kopieren. Beispiel: Um mehrere Spalten auszuwählen, halten Sie die Strg-Taste und klicken auf die Zellen der Spalte, die Sie kopieren möchten. Danach klicken Sie mit der rechten Maustaste und wählen **Daten kopieren→Spalten**. Siehe „Spalten, Zeilen und Zellen aus Interactive SQL-Ergebnismengen kopieren“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **OEM-Benutzer hindern, Kennwörter in den Favoriten zu speichern** OEM-Deployments können nun verhindern, dass Benutzer Kennwörter in ihren Verbindungsfavoriten in Interactive SQL speichern. Weitere Hinweise finden Sie bei der Option allowPasswordsInFavorites unter „Konfiguration der Administrationstools“ [[SQL Anywhere Server - Programmierung](#)].

- **Bearbeiten, Importieren und Exportieren von Favoriten** Sie können jetzt Ihre SQL-Favoriten bearbeiten, exportieren und importieren. Siehe „SQL-Skriptdateien, SQL-Anweisungen und Verbindungen zu den Favoriten hinzufügen“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Favoriten gemeinsam nutzen“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Ausführungszeiten** Die Statusleiste in Interactive SQL zeigt, wie lange die aktuelle SQL-Anweisung gelaufen ist.

- **Änderungen des Fensterausschnitts Ergebnisse**

- **Ergebnisse als Text oder in einer abrollbaren Tabelle anzeigen** Bisher konnten Sie die Anzeige einer Ergebnismenge nur im Fensterausschnitt **Ergebnisse** konfigurieren, indem Sie Einstellungen im Fenster **Optionen** änderten. Jetzt können Sie auf **Daten→Ergebnisse als abrollbare Tabelle anzeigen** klicken, um die Ergebnismenge in einer abrollbaren Tabelle anzuzeigen. Sie können auch auf **Daten→Ergebnisse als Text anzeigen** klicken, um die Ergebnismenge als Text anzuzeigen. Sie müssen eine Anweisung ausführen, um die Wirkung der

Änderung zu sehen. Siehe „[Interactive SQL anpassen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Spaltengrößen in Interactive SQL ändern** Sie können mit der rechten Maustaste auf eine Ergebnismenge klicken und auswählen, ob die Größe der Spalten an das Fenster oder an die Daten angepasst werden soll. Siehe „[Interactive SQL anpassen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Hilfe für SQL-Funktionen** Im Fensterausschnitt **SQL-Anweisungen** können Sie mit der rechten Maustaste auf eine SQL-Funktion klicken und dann auf **Hilfe für *function-name*** klicken. Daraufhin erscheint die Dokumentation der Funktion.
- **Warnungsmeldungen in Interactive SQL unterdrücken** Sie können bestimmte Warnmeldungen in Interactive SQL deaktivieren. Beispiel: Sie können die Warnung unterdrücken, die erscheint, wenn im Fensterausschnitt **SQL-Anweisungen** nicht gespeicherter Text vorhanden ist.

Sie können die Warnmeldung unterdrücken, indem Sie auf **Extras→Optionen→Meldungen** klicken und die Kontrollkästchen in der Liste **Optionale Meldungen** deaktivieren. Siehe „[Interactive SQL anpassen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Dateien in Interactive SQL wiederherstellen** Interactive SQL versucht, nicht gespeicherte Änderungen an *.sql*-Dateien wiederherzustellen, wenn Interactive SQL unerwartet geschlossen wird. Wenn Sie eine Datei bearbeiten, legt Interactive SQL 30 Sekunden nach der letzten Änderung und vor dem Ausführen einer Anweisung eine Sicherungskopie der Datei an.

Verhaltensänderungen von Interactive SQL und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Interactive SQL in Version 12.0.0.

- **Unterstützung für 32- und 64-Bit-Computer**
 - Sie können jetzt sowohl die 32-Bit- als auch die 64-Bit-Version von Interactive SQL auf demselben Computer installieren.
- **Änderung in der Standardkodierung für Windows** Die folgende Änderung gilt für das Ausführen von Interactive SQL als Konsolenanwendung (ohne Benutzeroberfläche mit Fenstern) auf Windows-Computern, auf denen andere ANSI- und OEM-Kodierungen gelten, beispielsweise auf einem Windows XP-Computer mit der Ländereinstellung U.S. English.
 1. Bisher galt: Wenn Sie Interactive SQL als Konsolenanwendung ausgeführt haben, gingen die Anweisungen INPUT und READ davon aus, dass die Datei mit der OEM-Kodierung (cp437 auf einem Windows XP-Computer mit der Ländereinstellung U.S. English) kodiert ist, wenn nicht eine explizite ENCODING-Klausel vorlag. Die OUTPUT-Anweisung gab die Datei mit der OEM-Kodierung aus.

Jetzt gilt: Wenn Interactive SQL als Konsolenanwendung eingesetzt wird, gehen die Anweisungen INPUT und READ davon aus, dass die Datei mit der ANSI-Kodierung (cp1252 auf

einem Windows XP-Computer mit der Ländereinstellung U.S. English) kodiert wurde. Die OUTPUT-Anweisung gibt die Datei mit der ANSI-Kodierung aus.

An der Eingabeaufforderung müssen Sie zur Verarbeitung einer Datei mit der OEM-Kodierung diese explizit angeben. Beispiel:

```
dbisql READ ENCODING 'cp437' myfile.sql
```

2. Bisher galt: Wenn Sie Interactive SQL als Konsolenanwendung ausgeführt haben, verwendeten die Ergebnisse, die in die Eingabeaufforderung geschrieben und aus ihr gelesen wurden, die ANSI-Kodierung (cp1252 auf einem Windows XP-Computer mit der Ländereinstellung U.S. English). Sonderzeichen wurden dabei falsch angezeigt.

Jetzt gilt: Wenn Sie Interactive SQL als Konsolenanwendung ausführen, verwenden die Ergebnisse, die in die Eingabeaufforderung geschrieben und aus ihr gelesen werden, die OEM-Kodierung (cp437 auf einem Windows XP-Computer mit der Ländereinstellung U.S. English).

Siehe „default_isql_encoding-Option [Interactive SQL]“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Änderung der CLEAR-Anweisung, Menüoption Löschen und Escape-Taste** Jetzt schließt die CLEAR-Anweisung offene Ergebnismengen und lässt die Inhalte des Fensterausschnitts **SQL-Anweisungen** unverändert. Siehe „CLEAR-Anweisung [Interactive SQL]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Auch die Menüoption **Bearbeiten & Ergebnisse schließen**, die dem Ausführen einer CLEAR-Anweisung entspricht, schließt offene Ergebnismengen und lässt die Inhalte des Fensterausschnitts SQL-Anweisungen unverändert.

Die Menüoption **Bearbeiten & Löschen**, mit der die Inhalte des SQL-Anweisungen-Fensterausschnitts gelöscht wurden, ist nicht mehr vorhanden. Daher ist auch das Tastenkürzel für die Menüoption **Löschen**, die Escape-Taste, nicht mehr im Menü vorhanden. Standardmäßig hat nun die Betätigung der Escape-Taste keine Wirkung.

Sie können aber die Esc-Taste mit dem Löschen des Fensterausschnitts **SQL-Anweisungen** und dem Schließen geöffneter Ergebnismengen belegen. Klicken Sie auf

Extras & Optionen & Kompatibilität und dann auf **Durch Drücken der Escape-Taste werden die SQL-Anweisungen gelöscht und Ergebnismengen geschlossen**.

- **Option -codepage wurde entfernt** Wenn Sie wollen, dass Interactive SQL eine Datei mit einer bestimmten Codepage liest, verwenden Sie die ENCODING-Klausel der INPUT-, OUTPUT- oder READ-Anweisung. Die Option -codepage wurde aus der Software entfernt. Siehe:
 - „Interactive SQL-Dienstprogramm (dbisql)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „Interactive SQL-Dienstprogramm für UltraLite (dbisql)“ [[UltraLite - Datenbankverwaltung](#)]
 - „INPUT-Anweisung [Interactive SQL]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „OUTPUT-Anweisung [Interactive SQL]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „READ-Anweisung [Interactive SQL]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Änderung der SET OPTION-Anweisung [Interactive SQL]** Bisher galt: Wenn Sie die SET OPTION-Anweisung verwendet haben, um eine Option zu aktivieren, dabei aber keinen Wert

festlegten, wurde die Option deaktiviert. Jetzt gilt: Wenn der Optionswert nicht vorhanden ist, wird die angegebene Option auf den Standardwert gesetzt. Diese Änderung betrifft folgende Optionen: `auto_commit`, `auto_refetch`, `bell`, `commit_on_exit` und `echo`. Siehe „[SET OPTION-Anweisung \[Interactive SQL\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

- **Änderung an der OUTPUT-Anweisung** Wenn die Ausgabe von Ergebnissen in eine TEXT-Datei erfolgt, können Sie die WITH COLUMN NAMES-Klausel benutzen, um die Spaltennamen am Beginn der Datei einzufügen. Siehe „[OUTPUT-Anweisung \[Interactive SQL\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Änderungen an der INPUT-Anweisung** Wenn Zeilen aus einer Textdatei mit der INPUT-Anweisung eingefügt werden, können Sie jetzt die Klausel SKIP verwenden, um eine Anzahl von Zeilen anzugeben, die ab dem Beginn der Datei nicht eingelesen werden sollen. Siehe „[INPUT-Anweisung \[Interactive SQL\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **READ-Schlüsselwort nicht mehr erforderlich** Wenn Sie Interactive SQL an der Eingabeaufforderung ausführen, ist das Schlüsselwort READ optional, wenn Sie eine `.sql`-Datei zum Ausführen angeben. Wenn die `.sql`-Datei Parameter erfordert, geben Sie die Parameter nach dem Dateinamen an.

Die folgenden beiden Befehle sind beispielsweise gleichwertig:

Mit dem READ-Schlüsselwort	Ohne das READ-Schlüsselwort
<code>READ file.sql parml</code>	<code>file.sql parml</code>

Siehe „[Interactive SQL-Dienstprogramm \(dbisql\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Verbesserungen der Unterstützung des Microsoft Excel ODBC-Treibers** Die nachstehende Liste beschreibt die Änderungen bezüglich des Exports von SQL Anywhere in Excel-Dateien über den Microsoft Excel ODBC-Treiber:
 - Bisher konnten Sie keine Daten exportieren, die als CHAR, LONG VARCHAR, NCHAR, NVARCHAR oder LONG NVARCHAR gespeichert wurden.
 Wenn Sie jetzt Daten, die als CHAR, LONG VARCHAR, NCHAR, NVARCHAR oder LONG NVARCHAR gespeichert sind, aus einer SQL Anywhere-Datenbank mit dem Microsoft Excel ODBC-Treiber exportieren, werden die Daten als VARCHAR (der ähnlichste Typ, der vom Excel-Treiber unterstützt wird) gespeichert.
 Der Microsoft Excel ODBC-Treiber unterstützt Textspaltenbreiten bis zu 255 Zeichen.
 - Sie können Daten exportieren, die als REAL, FLOAT und BIGINT gespeichert sind.
 - Daten, die mit den Datentypen MONEY und SMALLMONEY gespeichert sind, werden mit dem Datentyp CURRENCY exportiert. Ansonsten werden numerische Daten als Zahlen exportiert.
 - Sie können den **Export-Assistenten** zum Export von Tabellen verwenden.

Siehe „[Datenexport](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Neue Funktionen des SQL Anywhere-Monitors

Nachstehend finden Sie eine Liste der Funktionen, die dem SQL Anywhere-Monitor in Version 12.0.0 hinzugefügt wurden.

- **Neue Benutzeroberfläche** Die Monitor-Benutzeroberfläche basiert nun auf Dashboards. Dashboards enthalten Widgets zur Anzeige von Metriken, Alarmen und Ressourceninformationen. Dashboards sind benutzerspezifisch. Jeder Benutzer kann eigene Dashboards hinzufügen, bearbeiten oder löschen. Siehe „Dashboards“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **MobiLink-Serverfarmen und Relay Server-Farmen überwachen** Sie können den Monitor verwenden, um MobiLink-Serverfarmen und Relay Server-Farmen sowie SQL Anywhere-Datenbanken und MobiLink-Server zu überwachen. Siehe „SQL Anywhere-Monitor“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Verbindungen vom Monitor aus schließen** Sie können Verbindungen zu Ressourcen-Datenbanken vom Monitor aus schließen. Siehe „Schließen einer Verbindung mit dem Monitor“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **SQL Anywhere-Ressourcen zum Überwachen importieren** Sie können eine CSV-Datei erstellen, die eine Liste von Ressourcen enthält, und diese Liste in den Monitor importieren. Bisher konnten Sie dem Monitor jeweils nur eine Ressource gleichzeitig hinzufügen. Siehe „Mehrere Ressourcen mit einer CSV-Datei hinzufügen“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Wartung auf Anfrage ausführen** Administratoren können ungeplante Wartungsarbeiten über den Monitor ausführen. Siehe „Sichern des Monitors“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue Sicherungswarnung für SQL Anywhere-Ressourcen** Administratoren können den Monitor so konfigurieren, dass er einen Alarm ausgibt, wenn die SQL Anywhere-Ressource über einen bestimmten Zeitraum (Tage) nicht erfolgreich gesichert wurde. Standardmäßig gibt der Monitor 14 Tage nach der letzten erfolgreichen Sicherung einer Ressource einen Alarm aus. Siehe „Angaben von Alarm-Schwellenwerten“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Ausgabe der Zeit im Zeitformat des lokalen Browsers** Die im Monitor ausgewiesenen Zeiten sind immer die lokalen Zeiten des von Ihnen verwendeten Browsers. Hinweise zur Berechnung der Differenz zwischen der Browserzeit und der Ressourcenzeit finden Sie unter „Einführung in die Zeitanzeige“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Metriken exportieren** Sie können Metriken, mit denen ein Diagramm oder eine Tabelle verknüpft ist, in eine XML-Datei exportieren. Beispiel: Die meisten Metriken im Widget **Wichtigste Performance-Metriken** können exportiert werden. Siehe „Exportieren von Metriken“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Funktionen für die Fehlerbehandlung** Wenn Sie eine Fehlerbehandlung des Monitors durchführen müssen, können Administratoren die Funktionen **Nachrichtenlog** und **Ausnahmebedingungsberichte** verwenden. Siehe „Anzeigen des Nachrichtenlogs“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Ausnahmebedingungsberichte anzeigen“ [[SQL Anywhere Server - Datenbankadministration](#)].

Verhaltensänderungen des SQL Anywhere-Monitors

Nachstehend finden Sie eine Liste der Änderungen, die an SQL Anywhere-Monitor in Version 12.0.0 vorgenommen wurden.

- **Änderungen an der Metriksammlung** Bisher konnten Sie den Typ der Metrik konfigurieren, die der Monitor sammelt, und die Alarm-Schwellenwerte einstellen. Jetzt können Sie nur die Alarm-Schwellenwerte konfigurieren. Siehe „SQL Anywhere-Monitor“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Entfernen von Ressourcen** Sie können eine überwachte Ressource aus dem Monitor entfernen, ohne sie anzuhalten. Früher mussten Sie die Ressource anhalten und konnten sie erst dann entfernen. Siehe „Ressourcen entfernen“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Ressourcen haben keinen Zustand mehr** Ressourcen haben jetzt nur noch einen Status. Siehe „Dashboards“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Änderungen am Alarmstatus** Wenn eine Bedingung, die einen Alarm ausgelöst hat, nicht mehr besteht, ändert sich der Alarmstatus auf **Inaktiv**. Ein Alarm mit dem Status **Inaktiv** zeigt, dass die Bedingung, die den Alarm ausgelöst hat, nicht mehr vorhanden ist, die Bedingung aber nicht manuell von einem Monitor-Benutzer behoben wurde. Außerdem weist der Monitor jedem ausgegebenen Alarm einen Schweregrad zu. Siehe „Dashboards“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Änderungen an den Standardwerten für Alarm-Schwellenwerte** Bisher gab der Monitor Alarm aus, wenn die CPU-Speichernutzung bei zwei aufeinanderfolgenden Abfrageintervallen 85 Prozent der maximalen Cachegröße erreichte. Jetzt ist der Standardwert 90 Prozent und der Zeit-Schwellenwert 30 Sekunden. Siehe „Alarm-Definitionen und -Schwellenwerte“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Alle Benutzer müssen sich beim Monitor anmelden** Bisher war keine Anmeldung beim Monitor erforderlich, um als Benutzer ohne Schreibrechte zuzugreifen. Jetzt müssen sich alle Benutzer beim Monitor anmelden. Siehe „Monitor-Benutzer“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Änderungen der Spracheinstellungen der Benutzer** Wenn sich ein Benutzer anmeldet, verwendet der Monitor die Spracheinstellung des Benutzers, um die Anzeigesprache des Monitors und der Alarme zu konfigurieren. Bisher verwendete der Monitor die Sprache des Browsers. Siehe „Monitor-Benutzer bearbeiten“ [[SQL Anywhere Server - Datenbankadministration](#)].

Erweiterungen der Dokumentation

Nachstehend finden Sie eine Liste der Änderungen, die an der Dokumentation von SQL Anywhere in der Version 12.0.0 vorgenommen wurden.

- **Häufig gestellte Fragen (FAQ)** Eine umfassende Liste häufig gestellter Fragen ist nun verfügbar. Sie richten sich hauptsächlich an neue Benutzer und solche mit bereits vorhandenen Grundkenntnissen. Die Antworten beziehen sich auf Fragen, die häufig in Foren gestellt werden. Siehe „Häufig gestellte Fragen - SQL Anywhere“ [[SQL Anywhere 16 - Einführung](#)].

- **Performance-Warnungen des Datenbankservers** Für die Performancewarnungen werden nun Beschreibungen bereitgestellt, die im Fenster der Datenbankservermeldungen erscheinen. Darüber hinaus werden Links zu Informationen zur Behebung der Performanceprobleme bereitgestellt. Siehe „Fehlerbehandlung bei Performancewarnungen des Datenbankservers“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **DocCommentXchange ist die Standarddokumentation** DocCommentXchange ist das Standard-Dokumentationsformat für SQL Anywhere 12. DocCommentXchange ist eine Online-Plattform für den Zugriff auf und den Austausch über die SQL Anywhere Dokumentation im Web.
- **Neue Developer Center auf Sybase.com** Als Ergänzung zur SQL Anywhere-Dokumentation finden Sie im SQL Anywhere Developer Center unter <http://www.sybase.com/developer/library/sql-anywhere-techcorner> technische Whitepaper sowie FAQs, technische Hinweise, Downloads, Techcasts und weitere Antworten auf Ihre Fragen.

Neue Funktionen in Version 11.0.1

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

Informationen über neue Funktionen und Verhaltensänderungen in den Versionen von SQL Anywhere vor der Version 10 finden Sie unter <http://dcx.sybase.com/html/dbwnde10/dbwnde10.html>.

Neue Funktionen in SQL Anywhere

Im Folgenden finden Sie eine Liste der neuen Funktionen, die für SQL Anywhere-Datenbanken und Datenbankserver in Version 11.0.1 hinzugefügt wurden.

Verbindungseigenschaften

Die folgenden Verbindungseigenschaften wurden in SQL Anywhere Version 11.0.1 hinzugefügt:

- Authenticated
- IsDebugger
- QueryBypassedCosted
- QueryBypassedHeuristic
- QueryBypassedOptimized
- QueryOpened
- QueryDescribedBypass
- QueryDescribedOptimizer
- StatementDescribes
- StatementPostAnnotates
- StatementPostAnnotatesSimple
- StatementPostAnnotatesSkipped
- WaitStartTime
- WaitType

Eine Beschreibung dieser Eigenschaften finden Sie unter „[Liste der Verbindungseigenschaften](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Datenbankeigenschaften

Die folgenden Datenbankeigenschaften wurden in SQL Anywhere Version 11.0.1 hinzugefügt:

- Authenticated
- Prepares
- QueryBypassedCosted
- QueryBypassedHeuristic
- QueryBypassedOptimized
- QueryOpened
- QueryDescribedBypass
- QueryDescribedOptimizer
- StatementDescribes
- StatementPostAnnotates
- StatementPostAnnotatesSimple
- StatementPostAnnotatesSkipped

Eine Beschreibung dieser Eigenschaften finden Sie unter „[Liste der Datenbankereigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbankservereigenschaften

Die folgende Liste enthält die Erweiterungen der Datenbankservereigenschaften in SQL Anywhere Version 11.0.1.

- ServerEdition

Eine Beschreibung dieser Eigenschaften finden Sie unter „[Liste der Datenbankservereigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbank-Dienstprogramme

Die folgende Liste enthält die Erweiterungen der Datenbank-Dienstprogramme in SQL Anywhere Version 11.0.1.

- **Erweiterungen des Dienstprogramms für Dienste (dbsvc)** In Windows können Sie jetzt Dienste für den MobiLink Relay Server (RSHOST), den Relay Server Outbound Enabler (RSOE) und den Volume-Schattenkopiedienst (dbvss11) erstellen, indem Sie das Dienstprogramm für Dienste verwenden. Siehe „[Dienstprogramm für Dienste \(dbsvc\) für Windows](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen des Dienstprogramms Entladen (dbunload)** Das Dienstprogramm dbunload unterstützt nun die folgenden Optionen:
 - **Option -cm** Zeigt den Befehl dbinit oder die Anweisung CREATE DATABASE für die Datenbank an, die entladen wird.
 - **Option -l** Behält den nächsten verfügbaren Wert für die AUTOINCREMENT-Spalten in der neu aufgebauten Datenbank bei.

Siehe „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Systemprozeduren und Funktionen

In der folgenden Liste sind die Erweiterungen der Systemprozeduren und Funktionen in SQL Anywhere Version 11.0.1 aufgeführt.

- **sa_get_table_definition-Systemprozedur** Die neue sa_get_table_definition-Systemprozedur gibt eine LONG VARCHAR-Zeichenfolge mit den SQL-Anweisungen zurück, die für die Erstellung der festgelegten Tabelle und ihrer Indizes, Fremdschlüssel, Trigger und erteilten Berechtigungen erforderlich sind. Siehe „sa_get_table_definition-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **FIRST_VALUE-Funktion [Aggregat]** Die Funktion FIRST_VALUE [Aggregat] enthält jetzt eine RESPECT NULLS-Klausel. Siehe „FIRST_VALUE-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **LAST_VALUE-Funktion [Aggregat]** Die Funktion LAST_VALUE [Aggregat] enthält jetzt eine RESPECT NULLS-Klausel. Siehe „LAST_VALUE-Funktion [Aggregat]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_set_http_option-Systemprozedur** Die Option AcceptCharset ermöglicht nun eine bessere Steuerung der Zeichensatzauswahl. Siehe „sa_set_http_option-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

SQL-Anweisungen

Die folgende Liste enthält alle SQL-Erweiterungen in SQL Anywhere Version 11.0.1.

- **Neue DEFAULT VALUES-Klausel, INSERT-Anweisung** Die neue DEFAULT VALUES-Klausel der INSERT-Anweisung ermöglicht es Ihnen, die Standardwerte für alle Spalten einzufügen. Siehe „INSERT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **CREATE ENCRYPTED DATABASE-Anweisung** Diese Anweisung erstellt eine verschlüsselte Kopie einer vorhandenen Datenbank inklusive aller Transaktionslogs, Spiegellogs und DBSpaces. Sie können diese Anweisung auch verwenden, um eine Kopie einer Datenbank mit aktivierter Tabellenverschlüsselung zu erstellen. Siehe „CREATE ENCRYPTED DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Wenn Sie eine Datenbank, die eine Wiederherstellung erfordert, verschlüsseln möchten, um sie beispielsweise an den technischen Support zu senden, müssen Sie weiterhin die CREATE ENCRYPTED FILE-Anweisung verwenden. Siehe „CREATE ENCRYPTED FILE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **CREATE DECRYPTED DATABASE-Anweisung** Diese Anweisung erstellt eine entschlüsselte Kopie einer vorhandenen Datenbank inklusive aller Transaktionslogs, Spiegellogs und DBSpaces. Siehe „CREATE DECRYPTED DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Wenn Sie eine Datenbank, die eine Wiederherstellung erfordert, entschlüsseln möchten, um sie beispielsweise an den technischen Support zu senden, müssen Sie weiterhin die CREATE

DECRYPTED FILE-Anweisung verwenden. Siehe „CREATE DECRYPTED FILE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Erweiterung der ALTER DATABASE-Anweisung** Der Versuch, eine ALTER DATABASE UPGRADE-Anweisung auf einem Datenbankserver auszuführen, der gerade gespiegelt wird, führt jetzt zu einem Fehler. Siehe „ALTER DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der MESSAGE-Anweisung** Die IMMEDIATE-Klausel veranlasst, dass die Nachricht von der Callback-Routine des Clients innerhalb kurzer Zeit empfangen wird, unabhängig davon, ob die Verbindung inaktiv ist oder Anforderungen stellt. Siehe „MESSAGE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Funktion, Prozedur, Trigger oder Ansicht mit demselben Namen erstellen oder zu ersetzen** Die neue OR REPLACE-Klausel ermöglicht es Ihnen, eine Funktion, Prozedur, einen Trigger oder eine Ansicht mit demselben Namen zu erstellen oder ersetzen. Siehe:
 - „CREATE FUNCTION-Anweisung [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE PROCEDURE-Anweisung [Webdienste]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE TRIGGER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Fehler unterdrücken, wenn eine Anweisung versucht, ein Datenbankobjekt zu entfernen, das nicht existiert** Mit der neuen IF EXISTS-Klausel können Sie vermeiden, dass ein Fehler zurückgegeben wird, wenn die DROP-Anweisung versucht, ein Datenbankobjekt zu entfernen, das nicht existiert. Siehe:
 - „DROP EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP FUNCTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP MATERIALIZED VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP PROCEDURE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP TRIGGER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Neue INTO LOCAL TEMPORARY TABLE-Klausel, SELECT-Anweisung** Die neue INTO LOCAL TEMPORARY TABLE-Klausel der SELECT-Anweisung ermöglicht es Ihnen, eine lokale temporäre Tabelle zu erstellen und mit der Ergebnismenge einer SELECT-Anweisung zu füllen. Zuvor war dies nur unter Verwendung einer INTO-Klausel möglich, wenn der Name der temporären Tabelle mit # begann. Siehe „SELECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue IF NOT EXISTS-Klausel, CREATE TABLE-Anweisung** Die neue IF NOT EXISTS-Klausel der CREATE TABLE-Anweisung ermöglicht es Ihnen, permanente, globale temporäre und lokale temporäre Tabellen zu erstellen, wenn die Tabelle nicht bereits existiert. Siehe „CREATE TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Eigentümer beim Erstellen temporärer Prozeduren oder Funktionen angeben** Mit den Anweisungen CREATE FUNCTION und CREATE PROCEDURE können Sie jetzt optional den Eigentümer für eine temporäre Prozedur oder Funktion angeben. Siehe:
 - „CREATE FUNCTION-Anweisung [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „CREATE PROCEDURE-Anweisung [Webdienste]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Programmierschnittstellen

Die folgende Liste enthält die Erweiterungen der Programmierschnittstellen in SQL Anywhere Version 11.0.1.

- **Neue ASP.NET-Provider** Die folgenden neuen ASP.NET-Provider imitieren die Funktionen der standardmäßigen ASP.NET-Provider, speichern ihre Daten jedoch in einer SQL Anywhere-Datenbank anstatt in einer SQL Server-Datenbank:
 - **Mitgliedschaftsprovider** Ermöglicht das An- und Abmelden sowie das Verwalten von Benutzern und ihren Kennwörtern
 - **Rollen** Ermöglicht die Zuordnung von Benutzern zu Gruppen und vereinfacht damit die Berechtigungsverwaltung
 - **Profilprovider** Speichert Benutzervariablen
 - **Provider für Webseiten-Personalisierung** Verwaltet die Speicherung von Daten aus Webseiten und ermöglicht es Benutzern, ihre Ansicht zu personalisieren
 - **Provider für Webereignisse** Greift auf die Systemüberwachung zurück, um ausgegebene Webereignisdaten in der Datenbank zu protokollieren

Siehe „[SQL Anywhere ASP.NET-Provider](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **Unterstützung für Ruby** SQL Anywhere unterstützt ab sofort die Open Source-Programmiersprache Ruby. Siehe „[SQL Anywhere-Ruby-API-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **OLE DB unterstützt jetzt CATALOGS- und SCHEMATA-Zeilengruppen** Ab sofort werden die CATALOGS- und SCHEMATA-Zeilengruppen für OLE DB unterstützt. Da SQL Anywhere Kataloge nicht unterstützt, gibt der SQL Anywhere OLE DB-Anbieter stattdessen eine Ergebnismenge für CATALOGS zurück, die alle aktuell gestarteten Datenbanken und ihre Speicherorte enthält. Genauso wird für SCHEMATA der Datenbankname als Katalog in der Ergebnismenge verwendet.
- **Neue Unterstützung für ADO.NET Entity Framework** SQL Anywhere bietet nun Unterstützung für das ADO.NET Entity Framework-Modell. Siehe „[Funktionen des SQL Anywhere .NET-Datenproviders](#)“ [[SQL Anywhere Server - Programmierung](#)].

Verschiedenes

Die folgende Liste enthält verschiedene Erweiterungen in SQL Anywhere Version 11.0.1.

- **Bessere Performance für eine einfache, kostengünstige Anweisungsausführung** Die Kosten für die Erstellung eines Ausführungsplans für Anweisungen mit kurzen Ausführungszeiten können manchmal höher sein als die Kosten für die Ausführung der Anweisung an sich. Mit den folgenden Neuerungen kann SQL Anywhere nun eine erweiterte Klasse einfacher Anweisungen mit kurzen Ausführungszeiten erkennen und es Ihnen ermöglichen, den Optimierer zu umgehen.

Weitere Hinweise finden Sie unter „Eignung für ein Überspringen der Abfrageverarbeitungsphase“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

- **Neue FORCE NO OPTIMIZATION-Klausel** In früheren Versionen mussten alle Anweisungen, die eine kostenbasierte Optimierung erforderten, vom Optimierer verarbeitet werden. Mit der neuen FORCE NO OPTIMIZATION-Klausel können Sie jetzt festlegen, dass die Anweisung den Optimierer umgehen soll. Wenn die Anweisung zu komplex für eine Verarbeitung auf diese Weise ist – möglicherweise aufgrund der festgelegten Datenbankoptionen oder Eigenschaften des Schemas bzw. der Abfrage – schlägt die Anweisung fehl und der Datenbankserver gibt einen Fehler zurück.

Die folgenden Anweisungen unterstützen die neue FORCE NO OPTIMIZATION-Klausel:

- „DELETE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „INSERT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SELECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „UPDATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Neue Verbindungseigenschaften**
 - QueryBypassedCosted
 - QueryBypassedHeuristic
 - QueryBypassedOptimized
 - QueryOpened
 - QueryDescribedBypass
 - QueryDescribedOptimizer
 - StatementDescribes
 - StatementPostAnnotates
 - StatementPostAnnotatesSimple
 - StatementPostAnnotatesSkipped

- **Neue Datenbankeigenschaften**
 - Prepares
 - QueryBypassedCosted
 - QueryBypassedHeuristic
 - QueryBypassedOptimized
 - QueryOpened
 - QueryDescribedBypass
 - QueryDescribedOptimizer
 - StatementDescribes
 - StatementPostAnnotates
 - StatementPostAnnotatesSimple
 - StatementPostAnnotatesSkipped
- **Neues Optimierungsmethode-Feld** Das neue Optimierungsmethode-Feld befindet sich im grafischen Plan unter "Optimierer-Statistiken". Das Feld gibt die Ausführungsstrategie des Abfrageoptimierers zurück. Siehe „Komponenten von Ausführungsplänen“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Neue Serveroption verhindert, dass ein Datenbankserver der Standarddatenbankserver wird** Die Serveroption -xd hindert einen Datenbankserver daran, am standardmäßigen TCP-Port zu warten, und vermeidet außerdem, dass der Datenbankserver zum Standarddatenbankserver wird. Siehe „Datenbankserveroption -xd“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung für parallele Archivsicherungen** Der SQL Anywhere-Datenbankserver unterstützt nun parallele Sicherungen für serverseitige Archivkopien. Parallele Datenbanksicherungen nutzen physische I/O-Vorgänge, um Informationen parallel zu lesen und zu schreiben. Dies verbessert die Performance gegenüber sequenziellen Lese- und Schreibvorgängen.

Um parallele Archivsicherungen zu unterstützen, wurde die BACKUP DATABASE-Anweisung durch zwei neue Klauseln ergänzt:
 - **WITH CHECKPOINT LOG [NO] COPY**
 - **MAX WRITE { n | AUTO }**

Datenbankserver der Version 11.0.0 und früher können Archivsicherungen, die mit Datenbankservern der Version 11.0.1 generiert wurden, nicht wiederherstellen. Datenbankserver der Version 11.0.1 können Sicherungen, die mit älteren Datenbankservern generiert wurden, wiederherstellen.

Siehe „BACKUP-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Developer Edition- und Evaluation Edition-Datenbankserver auf Mac OS X ausführen** Sie können nun Developer Edition- und Evaluation Edition-Datenbankserver mithilfe der Administrationstools auf Mac OS X automatisch starten.

Verhaltensänderungen von SQL Anywhere und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine nach Kategorien sortierte Liste der Änderungen für SQL Anywhere-Datenbanken und -Datenbankserver in Version 11.0.1.

Verhaltensänderungen

- **Volltextsuche** Die folgenden Verhaltensänderungen wurden für die Unterstützung der Volltextsuche vorgenommen:
 - **Ab nun unterliegen Operatoren bestimmten Vorrangregeln.** In früheren Versionen unterlagen Operatoren in einer Abfragezeichenfolge keinen Vorrangregeln. Der Vorrang der Operatoren ist jetzt wie folgt geregelt:
 - NEAR- und FUZZY-Operatoren
 - AND NOT-Operator
 - AND-Operator
 - OR-Operator
- Siehe [Operatorvorrang in einer CONTAINS-Suchbedingung \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Argumente für die NEAR-Klausel müssen Begriffe oder Präfixbegriffe sein.** Wenn Sie eine Nachbarschaftssuche durchführen, müssen die Argumente für die NEAR-Klausel Begriffe oder Präfixbegriffe sein. Siehe „CONTAINS-Suchbedingung“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und „Nachbarschaftssuche“ [\[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#).
- **Verwendung von Bindestrich und AND NOT-Klausel** Innerhalb eines Satzes wird ein Bindestrich als Begriffsegmentierer und nicht als Sonderzeichen behandelt. Außerhalb eines Satzes ist die Bindestrichbehandlung von der Syntax um den Bindestrich herum abhängig. Siehe [Zulässige Syntax für den Bindestrich \(-\) \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und [Den AND NOT-Operator in der Volltextsuche verwenden \[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#).
- **Präfixsuche mit Sternchen** Wenn Sie eine Präfixsuche durchführen, hängen Sie das Sternchen an den Begriff an und setzen Sie direkt dahinter ein Leerzeichen, das Ende der Textabfragezeichenfolge oder eines der zugelassenen Sonderzeichen. Siehe [Zulässige Syntax für den Sternchen-Platzhalter \(*\) \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und „Präfixsuche“ [\[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#).
- **Der Versuch, einen mehrfach vorhandenen Textindex zu erstellen, führt nun zu einem Fehler.** Es ist nicht mehr möglich, mehrfach vorhandene Textindizes zu erstellen. Ein Textindex ist mehrfach vorhanden, wenn die folgenden Einstellungen mit denen eines bereits existierenden Textindex übereinstimmen:
 - Die referenzierte Basistabelle.
 - Die Spalten, die mit einem Index versehen werden sollen (Reihenfolge spielt keine Rolle).

- Die Einstellungen für das verwendete Konfigurationsobjekt (Begriffsegmentierer (TERM BREAKER), Begriffmindestlänge (MINIMUM TERM LENGTH), Begriffhöchstlänge (MAXIMUM TERM LENGTH), Stoppliste (STOPLIST), Kollationsinformationen).

Mehrfach vorhandene Textindizes, die mit einer früheren Version als SQL Anywhere 11.0.1 erstellt wurden, können in der Datenbank erhalten bleiben und verursachen keine Fehler, wenn sie mit einem Datenbankserver der Version 11.0.1 gestartet werden. Beim Neuladen einer Datenbank mit mehrfach vorkommenden Textindizes in Version 11.0.1 oder höher wird jedoch ein Fehler zurückgegeben.

Um mehrfach vorhandene Textindizes in einer vorhandenen Datenbank zu ermitteln, führen Sie die folgende Abfrage aus:

```
SELECT LIST( i.index_name )
FROM SYS.SYSIDX i
      JOIN SYS.SYSTEXTIDX t ON i.object_id = t.index_id AND t.sequence =
1
      JOIN SYS.SYSTEXTCONFIG F ON F.object_id = t.text_config
      JOIN (
        SELECT table_id, index_id, LIST( column_id, ', ' ORDER BY
column_id ) col_id
        FROM SYS.SYSIDXCOL
        GROUP BY table_id, index_id) x
      ON x.table_id = i.table_id AND x.index_id = i.index_id
WHERE i.index_category=4
GROUP BY i.table_id, f.term_breaker, f.min_term_length,
f.max_term_length,
      f.collation, ISNULL( f.char_stoplist, '-' ),
      ISNULL( f.nchar_stoplist, '-' ), x.col_id
HAVING count(*) > 1
```

Diese Abfrage funktioniert nur, wenn die Zeichenfolgen für STOPLIST exakt übereinstimmen oder NO STOPLIST angegeben ist. Zum Beispiel sieht diese Abfrage die Stopplisten 'a b c' und 'a - b c' nicht als dieselben an, die jedoch bei einer Suche nach Duplikaten während der Textindexerstellung als übereinstimmend gelten würden.

- **Reguläre Ausdrücke** Für die Suchbedingungen SIMILAR TO und REGEXP sowie für die Funktion REGEXP_SUBSTR wurde das Verhalten geändert. Dadurch bleibt SIMILAR TO weiterhin mit dem ANSI/SQL-Standard konsistent, während das Verhalten von REGEXP und REGEXP_SUBSTR jetzt konsistent mit Perl ist.
- **Datenbankkollationen und Suche nach Übereinstimmungen** In Vorgängerversionen verwendeten REGEXP und REGEXP_SUBSTR die Kollationsäquivalenz und Sortierreihenfolge, um zu erkennen, ob ein Literal- oder Zeichenklassenbereich im Muster mit der entsprechenden Zeichenfolge übereinstimmt. Jetzt greifen REGEXP und REGEXP_SUBSTR auf binäre Vergleiche von Codepunktwerten für die Suche nach Übereinstimmungen und die Bereichsauswertung zurück. Die Änderung wurde vorgenommen, um ein mit Perl 5.0 konsistentes Verhalten zu ermöglichen.

SIMILAR TO verwendet immer noch die Datenbankkollation für die Suche nach Übereinstimmungen und zur Bereichsauswertung. Siehe „Die Suchbedingungen LIKE, REGEXP und SIMILAR TO“ [SQL Anywhere Server - SQL-Referenzhandbuch].

- **Berücksichtigung der Groß- und Kleinschreibung in Datenbanken und `[:upper:]`- bzw. `[:lower:]`-Teilzeichenklassen** Bisher wurde in Datenbanken, in denen die Groß- und Kleinschreibung nicht berücksichtigt wurde, auch bei den Teilzeichenklassen `[:upper:]` und `[:lower:]` von SIMILAR TO und REGEXP nicht zwischen Groß- und Kleinbuchstaben unterschieden. In der neuen Version berücksichtigt `[:upper:]` nur Großbuchstaben und `[:lower:]` nur Kleinbuchstaben, unabhängig von der Berücksichtigung von Groß- und Kleinschreibung in der Datenbank.
- **Behandlung von Einschaltungszeichen (^), Unterstrich (_) und Prozentzeichen (%) als Metazeichen** In der nachfolgenden Tabelle wird die frühere und die neue Verwendung dieser Zeichen als Metazeichen erklärt:

Zeichen	Früheres Verhalten	Neues Verhalten
_ (Unterstrich)	Für SIMILAR TO, REGEXP und REGEXP_SUBSTR diente ein Unterstrich als Metazeichen, das jedes einzelne Zeichen fand.	Für SIMILAR TO dient ein Unterstrich als Metazeichen, das jedes einzelne Zeichen findet. Für REGEXP und REGEXP_SUBSTR dient ein Unterstrich nicht als Metazeichen. Stattdessen wird für REGEXP und REGEXP_SUBSTR ein Punkt (.) verwendet, um jedes einzelne Zeichen zu finden.
%	Für SIMILAR TO, REGEXP und REGEXP_SUBSTR diente ein Prozentzeichen als Metazeichen, das jede Anzahl von beliebigen Zeichen fand.	Für SIMILAR TO dient ein Prozentzeichen als Metazeichen, das jede Anzahl von beliebigen Zeichen findet. Für REGEXP und REGEXP_SUBSTR dient ein Prozentzeichen nicht als Metazeichen. Stattdessen wird für REGEXP und REGEXP_SUBSTR ein Punkt, gefolgt von einem Sternchen (.*) verwendet, um jede Anzahl von beliebigen Zeichen zu finden.

Zeichen	Früheres Verhalten	Neues Verhalten
^	Für SIMILAR TO, REGEXP und REGEXP_SUBSTR diente ein Einschaltungszeichen innerhalb einer Zeichenklasse als Negations- oder Subtraktionszeichen für alles, was rechts davon stand: Damit wurde also nach allen Zeichen gesucht, die nicht in dieser Menge waren.	Für SIMILAR TO dient ein Einschaltungszeichen als Negations- oder Subtraktionszeichen für alle Zeichen, die rechts davon stehen. Beispiel: SIMILAR TO [a-d^c] findet a, b, d, aber nicht c. Für REGEXP und REGEXP_SUBSTR dient ein Einschaltungszeichen nur als Metazeichen, wenn es an erster Stelle innerhalb einer Zeichenklasse steht: In diesem Fall wird es als Negation der Zeichenklasse interpretiert. Beispiel: REGEXP [^abc] findet jedes einzelne Zeichen, das nicht a, b oder c ist, während REGEXP [a-d^c] a, b, c, d und ^ findet.

- **Verhaltensänderung des Upgrade-Dienstprogramms (dbupgrad)** Der Versuch, das Upgrade-Dienstprogramm für das Upgrade einer Datenbank zu verwenden, die an einer Datenbankspiegelung teilnimmt, führt jetzt zu einem Fehler. Siehe „Dienstprogramm zum Upgrade (dbupgrad)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Für Mac OS X ist das Dienstprogramm dbmodenv nicht mehr erforderlich.** Um in früheren Versionen die grafischen Administrationstools mit Mac OS X verwenden zu können, mussten Benutzer in der Datei `$HOME/.MacOSX/environment.plist` den SQL Anywhere-Binär- und Bibliothekspfad zu PATH und DYLD_LIBRARY_PATH hinzufügen. Hierfür wurde das Dienstprogramm dbmodenv eingesetzt. Da SQL Anywhere nun nicht mehr von den Einstellungen in `$HOME/.MacOSX/environment.plist` abhängt, ist es nicht mehr notwendig, dbmodenv auszuführen oder sich nach dem Installieren von SQL Anywhere ab- und wieder anzumelden.
- **Neuer Standard bei der Angabe von NULL durch die OUTPUT-Anweisung in dbisqlc** Bei der Verwendung der OUTPUT-Anweisung von dbisqlc gab die Anweisung bisher den Wert (NULL) für NULL zurück. Ab jetzt gibt die Anweisung standardmäßig eine leere Zeichenfolge für NULL zurück. Mit der Option `output_nulls` können Sie festlegen, wie NULL exportiert wird. Siehe „output_nulls-Option [Interactive SQL]“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Endian-Unterstützung** Nach einem Upgrade müssen Textindizes, die in Versionen vor 11.0.1 auf einem Big-Endian-Computer erstellt wurden, gekürzt und aktualisiert (MANUAL REFRESH- und AUTO REFRESH-Textindizes) oder neu erstellt werden (IMMEDIATE REFRESH-Indizes).

- **Administrationstools in Mac OS X** In Mac OS X verwenden die SQL Anywhere-Verwaltungstools nun 64-Bit JDK 1.6. Die Administrationstools laufen nur auf Intel-basierten Macintosh-Computern mit 64-Bit-Prozessoren, die von Apple JDK 1.6 (Mac OS X 10.5.2 oder höher) unterstützt werden. Beim Deployment der Administrationstools für Mac OS X finden Sie die nativen Bibliotheken unter `$$SQLANY11/System/lib64`. Siehe „[Deployment von Administrationstools unter Linux, Solaris und Mac OS X](#)“ [*SQL Anywhere Server - Programmierung*].
- **Neue Standardgröße für aufgeteilte Übertragungskodierung für HTTP-Clients** Bisher wurde die aufgeteilte Übertragungskodierung standardmäßig angewendet, wenn die Größe der vom HTTP-Client gesendeten Daten mehr als 2048 Byte betrug (oder wenn der Benutzer `CREATE PROCEDURE...SET 'HTTP(CH=auto)'` angab). Die Standardgröße wurde jetzt von 2048 in 8196 Byte geändert. Darüber hinaus wurde der neue Status, 411 **Länge erforderlich**, zu den Kriterien für die erneute Ausgabe der Anforderung ohne Verwendung der aufgeteilten Übertragungskodierung hinzugefügt. Siehe „[CREATE PROCEDURE-Anweisung \[Webdienste\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Unterstützung der ansi_substring-Option** Die ansi_substring-Option wurde in Version 11.0.0 nicht mehr unterstützt und wird nun in Version 11.0.1 wieder unterstützt. Siehe „[ansi_substring-Option](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Nicht weiterentwickelte und nicht mehr unterstützte Funktionen

- **COMMENT ON EXTERNAL ENVIRONMENT OBJECT *object-name*** Die Syntax wurde in `COMMENT ON EXTERNAL OBJECT object-name` geändert. Die alte Syntax wird zwar derzeit noch akzeptiert, in einer zukünftigen Version jedoch eventuell nicht mehr unterstützt. Siehe „[COMMENT-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Neue MobiLink-Funktionen

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in MobiLink in Version 11.0.1 hinzugefügt wurden.

- **Caching von Schemata der entfernten Datenbank** Die neue Funktion für das Caching von Schemata reduziert den Overhead für kleinere Synchronisationen. Entfernte Schemata werden vom MobiLink-Server bei der ersten Synchronisation im Cache abgelegt. Bei nachfolgenden Synchronisationen werden Informationen entfernter Schemata nur an den MobiLink-Server gesendet, wenn diese nicht bereits im Cache abgelegt wurden.
- **Option -vi für mlsrv11** Zeigt den Spaltenwert der einzelnen Upload-Zeilen an. Siehe „[mlsrv16-Option -v](#)“ [*MobiLink - Serveradministration*].
- **Option -vq für mlsrv11** Zeigt die Spaltenwerte der einzelnen Download-Zeilen an. Siehe „[mlsrv16-Option -v](#)“ [*MobiLink - Serveradministration*].
- **Option -vm für mlsrv11** Schreibt beim Abschluss einer Synchronisation die Dauer der Synchronisation und die Dauer der einzelnen Synchronisationsphasen ins Log. Siehe „[mlsrv16-Option -v](#)“ [*MobiLink - Serveradministration*].

- **Option -ppv für mlsrv11** Bewirkt, dass MobiLink neue periodische Überwachungswerte entsprechend dem festgelegten Zeitraum ausgibt. Siehe „[mlsrv16-Option -ppv](#)“ [*MobiLink - Serveradministration*].
- **Präfix ml_ignore** Der MobiLink-Server erkennt SQL-Skripten mit dem Präfix --{ml_ignore} als absichtlich ignorierte Skripten. Siehe „[Ignorierte Skripten](#)“ [*MobiLink - Serveradministration*].
- **Option -sv für dblsn** Legt die Skriptversion fest, die der MobiLink Listener für die Authentifizierung mit einer Datenbank verwendet. Siehe „[dblsn-Option -sv](#)“ [*MobiLink - Serverinitiierte Synchronisation*].
- **Unterstützung von Oracle VArray** Der Oracle ODBC-Treiber von iAnywhere Solutions 11 unterstützt nun die Verwendung von Oracle VARRAY in gespeicherten Prozeduren. Siehe „[Oracle-VARRAY](#)“ [*MobiLink - Serveradministration*].
- **Listener-Schlüsselwortvariablen für den Lightweight-Polling-Modus** Die Listener-Schlüsselwörter poll_connect, poll_notifier, poll_key und poll_every wurden hinzugefügt, um den Lightweight-Polling-Modus zu unterstützen.
- **Aktionsvariablen für den Lightweight-Polling-Modus** Die Aktionsvariablen \$poll_connect, \$poll_notifier, \$poll_key und \$poll_every wurden hinzugefügt, um den Lightweight-Polling-Modus zu unterstützen.
- **Client-Authentifizierung mit Common Access Cards** MobiLink-Clients unterstützen nun die Authentifizierung mit Clientidentitäten von Common Access Cards (CACs). Siehe „[identity_name](#)“ [*MobiLink - Clientadministration*].

Hinweis

Diese Funktion ist Bestandteil der Zusatzkomponente CAC-Authentifizierung und erfordert eine separate Lizenz. Siehe „[Getrennt lizenzierbare Komponenten](#)“ [*SQL Anywhere 16 - Einführung*].

- **Unterstützung für Microsoft SQL Server 2008** Der MobiLink-Synchronisationsserver unterstützt nun konsolidierte Datenbanken, die auf Microsoft SQL Server 2008 laufen. Weitere Hinweise über die Zuordnung der neuen Microsoft-Datentypen DATE, TIME und DATETIME2 finden Sie unter „[Microsoft SQL Server-Datentypzuordnung](#)“ [*MobiLink - Serveradministration*].
- **Neue Methode für .NET DownloadTableData-Schnittstelle** Die getLastDownloadTime-Methode zum Zurückgeben der letzten Downloadzeit für eine Tabelle wurde hinzugefügt. Siehe [DownloadTableData.GetLastDownloadTime-Methode](#) [*MobiLink-Server .NET*] [*MobiLink - Serveradministration*].
- **Ausführlichkeitsstufe des Logs für bestimmte MobiLink-Benutzer und entfernte IDs** Ab sofort können Sie verschiedene Ausführlichkeitsstufen des Logs für einen bestimmten MobiLink-Benutzer oder eine entfernte ID festlegen. Siehe [Ausführlichkeitsstufe des Logs für bestimmte MobiLink-Benutzer und entfernte IDs](#) [*MobiLink - Serveradministration*].
- **Unterstützung für MySQL im Modellmodus** Das MobiLink-Plug-In unterstützt nun konsolidierte MySQL-Datenbanken.

- **Option -c für mlmon** Die Option -c wurde dem mlmon-Befehl für den MobiLink-Monitor hinzugefügt. Die Option -c schließt den MobiLink-Monitor am Ende der Verbindung und speichert die Sitzung in der angegebenen Datenbank.

MobiLink-Verhaltensänderungen und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen, die in MobiLink in Version 11.0.1 vorgenommen wurden.

- **Option -sm für mlsrv11** Die Option -sm für mlsrv11 bietet nun eine ähnliche Funktionalität wie die Option -nc, wenn sie mit einem nichtbeständigen HTTP/HTTPS verwendet wird. Siehe „[mlsrv16-Option -sm](#)“ [*MobiLink - Serveradministration*] und „[mlsrv16-Option -nc](#)“ [*MobiLink - Serveradministration*].
- **Microsoft SQL Server-Datentypen** Seit SQL Server 2005 gelten die Typen TEXT, NTEXT und IMAGE als nicht mehr empfohlen. Verwenden Sie stattdessen VARCHAR(max), NVARCHAR(max) und VARBINARY(max). Siehe „[Microsoft SQL Server-Datentypzuordnung](#)“ [*MobiLink - Serveradministration*].

Neue Funktionen von UltraLite

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in UltraLite in Version 11.0.1 hinzugefügt wurden.

- **Spiegeldatei** UltraLite ermöglicht die Spiegelung von Datenbankdateien, um die Fehlertoleranz in potenziell unzuverlässigen Speichersystemen zu erhöhen. Für diesen Zweck wird die Spiegeldatei verwendet. Alle Änderungen an der Datenbank werden gleichzeitig in die Spiegeldatei und in die Hauptdatenbankdatei geschrieben. Siehe „[UltraLite-Verbindungsparameter MIRROR_FILE](#)“ [*UltraLite - Datenbankverwaltung*].
- **ml_remote_id zurücksetzen** Sie können nun **ml_remote_id** mithilfe von SET OPTION zurücksetzen. Siehe „[SET OPTION-Anweisung \[UltraLite\]](#)“ [*UltraLite - Datenbankverwaltung*] und „[ML_GET_SERVER_NOTIFICATION-Funktion \[System\]](#)“ [*UltraLite - Datenbankverwaltung*].
- **ML_GET_SERVER_NOTIFICATION** Mit dieser Funktion können UltraLite-Benutzer den Lightweight-Polling-Modus verwenden, um einen Notifier auf einem MobiLink-Server für serverinitiierte Synchronisationsanforderungen abzurufen. Siehe „[ML_GET_SERVER_NOTIFICATION-Funktion \[System\]](#)“ [*UltraLite - Datenbankverwaltung*].
- **SYNC_PROFILE_OPTION_VALUE-Funktion** Diese Funktion gibt den Wert bestimmter Optionen innerhalb eines Synchronisationsprofils zurück. Siehe „[SYNC_PROFILE_OPTION_VALUE-Funktion \[System\]](#)“ [*UltraLite - Datenbankverwaltung*].
- **StreamErrorParameters-Eigenschaft für die UltraLite.NET-API** StreamErrorParameters wurde zur Klasse SyncResult hinzugefügt. Dieses Mitglied enthält eine durch Kommas getrennte

Liste mit Fehlerparametern für den Datenstromfehlercode, der in `StreamErrorCode` gemeldet wird. Siehe [ULSyncResult.StreamErrorParameters-Eigenschaft \[UltraLite.NET\]](#) [*UltraLite - .NET-Programmierung*].

- **getStreamErrorParameters-Methode für die UltraLite für M-Business-API** Diese Methode gibt eine durch Kommas getrennte Liste mit Fehlerparametern zurück, die bei der Verarbeitung des Synchronisationsdatenstroms gemeldet wurden.

UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen in UltraLite ab Version 11.0.1.

- **Suspend-Funktionalität für Palm** Wird nicht mehr empfohlen.

Neue Funktionen von UltraLiteJ

Im Folgenden finden Sie eine Liste der neuen Funktionen von UltraLiteJ in Version 11.0.1.

- **Zusätzliche SQL-Unterstützung** UltraLiteJ bietet folgende zusätzliche SQL-Unterstützung:
 - ALTER TABLE
 - CREATE TABLE
 - CREATE INDEX
 - DROP INDEX
 - DROP TABLE
 - TRUNCATE TABLE

Darüber hinaus wurden die folgenden Einschränkungen aufgehoben:

- HAVING wird unterstützt.
- DISTINCT innerhalb von Aggregatfunktionen wird unterstützt.
- CURRENT TIME, CURRENT TIMESTAMP und CURRENT DATE werden unterstützt.

Siehe „[UltraLite-SQL-Anweisungen](#)“ [*UltraLite - Datenbankverwaltung*].

- **Neue DatabaseInfo-Methoden** Zwei neue Methoden, `getPageReads()` und `getPageWrites`, wurden zur Schnittstelle **DatabaseInfo** hinzugefügt. Diese Methoden geben die Anzahl der Seitenlese- und Schreibvorgänge zum Zeitpunkt der Erstellung eines DatabaseInfo-Objekts zurück. Siehe [DatabaseInfo.getPageReads-Methode \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*] und [DatabaseInfo.getPageWrites-Methode \[UltraLiteJ\]](#) [*UltraLite® – Java-Programmierung*].

- **Update des Dienstprogramms für UltraLiteJ-Datenbankübertragung** Das Dienstprogramm für die UltraLiteJ-Datenbankübertragung bietet jetzt die Möglichkeit, eine Datenbank zu löschen, Datenbankinformationen anzuzeigen, die Logdatei zur Datenbankübertragung anzuzeigen oder per E-Mail zu senden und eine Datenbank vom Client zu übertragen. Siehe „[UltraLite Java Edition-Dienstprogramm für die Datenbankübertragung](#)“ [[UltraLite - Datenbankverwaltung](#)].

Verhaltensänderungen von UltraLiteJ und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen, die in UltraLiteJ in Version 11.0.1 vorgenommen wurden.

- **Globale Variable @@identity** Die globale Variable @@identity wird von UltraLiteJ nicht unterstützt.

Neue Funktionen des Administrationstools

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Sybase Central und Interactive SQL in Version 11.0.1 hinzugefügt wurden.

Neue Funktionen in Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen der Sybase Central-Plug-Ins in Version 11.0.1.

Alle Plug-Ins

- **Erweiterungen des Assistenten zum Erstellen eines neuen Dienstes** Ab sofort können Sie den **Assistenten zum Erstellen eines neuen Dienstes** verwenden, um Dienste für den MobiLink Relay Server (rshost), Relay Server Outbound Enabler (RSOE), Volume-Schattenkopiedienst (dbvss11) sowie die Dienstprogramme MobiLink Listener (dblsn) und Broadcast Repeater (dbns11) zu erstellen. Siehe „[Datenbankserver als Dienst oder Daemon ausführen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

SQL Anywhere-Plug-In

- **Status von Ereignissen in Sybase Central anzeigen** Ab jetzt können Sie den aktuellen Ausführungsstatus von Ereignissen in Sybase Central anzeigen. Im Ordner **Ereignisse** befindet sich die neue Spalte **Läuft**, die den aktuellen Ausführungsstatus eines Ereignisses anzeigt. Die Ordnerinhalte werden jedes Mal aktualisiert, wenn der Ordner im linken Fensterausschnitt ausgewählt wird. Zusätzlich steht die Eigenschaft **Läuft** im Fenster **Eigenschaften des Ereignisses** zur Verfügung.

Der Läuft-Wert zeigt **Ja** an, wenn das Ereignis ausgeführt wird, **Nein**, wenn das Ereignis nicht ausgeführt wird, und **Unbekannt**, wenn sich das Ereignis in einer SQL Anywhere-Datenbank der Version 9.0.2 oder früher befindet.

- **Status der Wartungspläne in Sybase Central anzeigen** Ab jetzt können Sie den aktuellen Ausführungsstatus von Wartungsplänen in Sybase Central anzeigen. Im Ordner **Wartungspläne** befindet sich die neue Spalte **Läuft**, die den aktuellen Ausführungsstatus eines Plans anzeigt. Die Ordnerinhalte werden jedes Mal aktualisiert, wenn der Ordner im linken Fensterausschnitt ausgewählt wird.

Der Läuft-Wert zeigt **Ja** an, wenn der Wartungsplan ausgeführt wird, **Nein**, wenn der Plan nicht ausgeführt wird, und **Unbekannt**, wenn sich der Plan in einer SQL Anywhere-Datenbank der Version 9.0.2 oder früher befindet.

- **Erweiterungen des Wartungsplans** Mit dem **Assistenten zum Erstellen von Wartungsplänen** können Sie nun eine Test-E-Mail senden, wenn Sie in den Einstellungen festlegen, dass ein Wartungsplanbericht per E-Mail verschickt werden soll. Siehe „Wartungspläne“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen des Assistenten zum Erstellen von Prozeduren und des Assistenten zum Erstellen von Funktionen** Mit dem **Assistenten zum Erstellen von Funktionen** und dem **Assistenten zum Erstellen von Prozeduren** können Sie jetzt einen der folgenden SQL-Dialekte bzw. eine der folgenden Sprachen auswählen, mit denen die Prozedur oder die Funktion geschrieben werden soll: Watcom-SQL, Transact-SQL, External C/C++ oder externe Umgebung. Wenn Sie Externe Umgebung auswählen, steht Ihnen eine der folgenden Sprachen zur Verfügung: C_ESQL32, C_ESQL64, C_ODBC32, C_ODBC64, CLR, JAVA, PERL oder PHP. In Vorgängerversionen konnten Sie nur zwischen Watcom-SQL und Transact-SQL wählen. Wenn Sie C/C++ oder Java verwenden, wird ein Codegerüst für die Funktion oder Prozedur mit der EXTERNAL NAME-Klausel generiert. Siehe „CREATE FUNCTION-Anweisung [externer Aufruf]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „CREATE PROCEDURE-Anweisung [externer Aufruf]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

MobiLink-Plug-In

- **Erweiterung des MobiLink-Modells für Oracle-Datenbanken** Wenn Sie mithilfe des **Assistenten zum Erstellen eines Synchronisationsmodells** ein Synchronisationsmodell erstellen, das eine konsolidierte Oracle-Datenbank verwendet, können Sie jetzt entweder das gesamte Schema laden oder eine Teilmenge von Eigentümern bestimmen, deren Tabellen Sie für die Synchronisation benötigen. Durch das Auswählen einer Teilmenge von Eigentümern können Sie den Zeitaufwand zum Laden des Schemas reduzieren.
- **Unterstützung des MobiLink-Modellmodus für MySQL** Der MobiLink-Modellmodus unterstützt nun die Verwendung von konsolidierten MySQL-Datenbanken.

Verhaltensänderungen des Administrationstools und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen, die in Sybase Central und Interactive SQL in Version 11.0.1 vorgenommen wurden.

Verhaltensänderungen von Sybase Central und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Sybase Central in Version 11.0.1.

SQL Anywhere-Plug-In

- **Verbesserung der schreibgeschützten Datenbank** Ab sofort erhalten Sie eine Warnung, wenn Sie eine Verbindung mit einer schreibgeschützten Datenbank herstellen. Wenn Sie diese Warnung unterdrücken möchten, klicken Sie auf **Extras&rarrSQL Anywhere 11&rarrVoreinstellungen**.

Verhaltensänderungen von Interactive SQL und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Interactive SQL in Version 11.0.1.

- **Automatische Freigabe von Datenbanksperren konfigurieren** Ab nun kann Interactive SQL automatisch versuchen, die Datenbankschemasperren freizugeben, die bei der Anzeige Ihrer Ergebnismenge erstellt werden. Klicken Sie dafür in Interactive SQL auf **Extras&rarrOptionen&rarrSQL Anywhere** und anschließend auf **Datenbanksperren automatisch freigeben**.

Nachdem Sie diese Option ausgewählt haben, überprüft Interactive SQL jedes Mal, wenn Sie eine Anweisung mit darauf folgender Rückgabe einer Ergebnismenge ausgeführt haben, ob im Rahmen Ihrer Verbindung nicht festgeschriebene Änderungen in der Datenbank vorhanden sind. Wenn keine gefunden werden, gibt Interactive SQL Ihre Schemasperren frei; andernfalls bleiben die Schemasperren erhalten. Das heißt, Interactive SQL gibt Ihre Schemasperren nicht frei, wenn nicht festgeschriebene Änderungen in der Datenbank vorhanden sind.

Produktweite neue Funktionen

Im Folgenden finden Sie eine Liste der produktweiten neuen Funktionen, die in Version 11.0.1 hinzugefügt wurden.

- **Neues Online-Dokumentationsforum: DocCommentXchange (DCX)** Ab nun steht das neue Online-Forum DocCommentXchange zur Verfügung. DocCommentXchange ist eine gemeinsam genutzte Plattform für den Zugriff auf und den Austausch über die SQL Anywhere-Dokumentation.

DocCommentXchange bietet Ihnen folgende Möglichkeiten:

- Dokumentation anzeigen
- Nach Erklärungen anderer Benutzer zu einzelnen Abschnitten der Dokumentation suchen
- Vorschläge und Korrekturen bereitstellen, um die Dokumentation in zukünftigen Versionen für alle Benutzer zu verbessern

Besuchen Sie <http://dcx.sybase.com>.

- **SQL Anywhere-Monitor** Der SQL Anywhere-Monitor ist ein browserbasiertes Administrationstool, das Informationen über den Zustand und die Verfügbarkeit von SQL Anywhere-Datenbanken und MobiLink-Servern bereitstellt.

Für MobiLink-Benutzer gilt, dass dieses neue Tool weder die Funktionalität des MobiLink-Monitors ersetzt noch sich mit dieser überschneidet.

Siehe „SQL Anywhere-Monitor“ [[SQL Anywhere Server - Datenbankadministration](#)].

Erweiterungen der Dokumentation

- **Erweiterungen der Dokumentation zum Sichern und Wiederherstellen von SQL Anywhere** Die Dokumentation für die Verwendung der Sicherungs- und Wiederherstellungstools in SQL Anywhere wurde für diese Version neu geschrieben. Siehe „[Sicherung und Datenwiederherstellung](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen der Dokumentation zur Volltextsuche** Die Dokumentation zur Volltextsuche wurde neu strukturiert und enthält jetzt mehr Beispiele und praktische Einführungen. Siehe „[Volltextsuche](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Dokumentation zur serverinitiierten MobiLink-Synchronisation** Die Dokumentation zur serverinitiierten MobiLink-Synchronisation wurde seit Version 11.0.0 überarbeitet. Zu den Verbesserungen gehören zum Beispiel ein umstrukturiertes Inhaltsverzeichnis, detailliertere Erklärungen zur serverinitiierten Synchronisation und Referenzmaterial in einem neuen Format. Siehe [MobiLink - Serverinitiierte Synchronisation](#).
- **Dokumentation zu SQL Remote** Die Dokumentation zu SQL Remote wurde umstrukturiert und neu geschrieben, um eine bessere Navigation und höhere Benutzerfreundlichkeit zu gewährleisten. Siehe [SQL Remote](#).

Neue Funktionen in Version 11.0.0

Informationen über neue Funktionen und Verhaltensänderungen in den Versionen von SQL Anywhere vor der Version 10 finden Sie unter <http://dcx.sybase.com/html/dbwnde10/dbwnde10.html>.

Hinweis

Wie bei jeder Aussage, die zukünftige Entwicklungen betrifft, kann bei den Listen der nicht mehr empfohlenen Funktionen keine Gewähr für Vollständigkeit übernommen werden. Änderungen sind vorbehalten.

SQL Anywhere

- „Neue Funktionen in SQL Anywhere“ auf Seite 220
- „Verhaltensänderungen von SQL Anywhere“ auf Seite 250
- „Nicht mehr empfohlene und nicht mehr weiterentwickelte SQL Anywhere-Funktionen“ auf Seite 259

MobiLink

- „Neue MobiLink-Funktionen“ auf Seite 263
- „MobiLink-Verhaltensänderungen und nicht mehr empfohlene Funktionen“ auf Seite 267

SQL Remote

- „Neue Funktionen in SQL Remote“ auf Seite 269
- „Verhaltensänderungen von SQL Remote und nicht mehr empfohlene Funktionen“ auf Seite 269

UltraLite

- „Neue Funktionen von UltraLite“ auf Seite 270
- „UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen“ auf Seite 275

Sybase Central und Interactive SQL

- „Neue Funktionen in Sybase Central und Interactive SQL“ auf Seite 277
- „Verhaltensänderungen und nicht mehr empfohlene Funktionen von Sybase Central und Interactive SQL“ auf Seite 280

Erweiterungen der Dokumentation

- „Erweiterungen der Dokumentation“ auf Seite 286

Produktweite Funktionen

- „Produktweite neue Funktionen“ auf Seite 287
- „Produktweite Verhaltensänderungen“ auf Seite 287

SQL Anywhere

In den folgenden Abschnitten finden Sie Beschreibungen der neuen Funktionen und Verhaltensänderungen in SQL Anywhere für Version 11.0.0 und Informationen dazu, welche Funktionen nicht mehr empfohlen werden.

Neue Funktionen in SQL Anywhere

Im Folgenden finden Sie eine Liste der neuen Funktionen, die für SQL Anywhere-Datenbanken und -Datenbankserver in Version 11.0.0 hinzugefügt wurden.

Wichtigste Funktionen

Im Folgenden finden Sie eine Liste der wichtigsten Funktionen, die in SQL Anywhere Version 11.0.0 eingeführt wurden.

- **Unterstützung für die Zusammenführung von Tabellen** Sie können in SQL Anywhere nun Tabellen, Ansichten und Systemprozedurergebnisse in einer Tabelle oder Ansicht zusammenführen. Siehe „[MERGE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Unterstützung von Login-Richtlinien** SQL Anywhere unterstützt nun Login-Richtlinien. Eine Login-Richtlinie umfasst eine Reihe von Optionen für Regeln, die beim Herstellen einer Datenbankverbindung für einen Benutzer angewendet werden müssen. SQL Anywhere unterstützt eine Stammrichtlinie (sogenannte "Root-Richtlinie"), die zum Speichern von Standardwerten für alle Login-Richtlinien verwendet wird. Eigene Login-Richtlinien können erstellt und gespeichert werden, mit denen die Optionen der Root-Login-Richtlinien außer Kraft gesetzt werden. Jedem Benutzer wird eine Login-Richtlinie zugewiesen. Login-Richtlinien werden nicht durch die Gruppenmitgliedschaft vererbt. Die Zuweisung einer Richtlinie an einen bestimmten Benutzer kann nach Bedarf geändert werden. Siehe „[Login-Richtlinien](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Zur Nutzung dieser Funktion ist ein Datenbank-Upgrade erforderlich. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

- **Unterstützung der Volltextsuche** SQL Anywhere unterstützt nun die Volltextsuche. Bei der Volltextsuche können alle Instanzen eines Begriffs (Worts) in einer Datenbank schnell gefunden werden. Die Volltextsuche unterscheidet sich von der normalen Suche durch die Verwendung von Prädikaten, wie z. B. LIKE, REGEXP und SIMILAR TO, da sie begriffsbasiert ist und einen Textindex verwendet, anstatt Tabellenzeilen zu durchsuchen. Siehe „[Volltextsuche](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Sie müssen ein Upgrade der Datenbank durchführen, um die Volltextsuchfunktion verwenden zu können. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

- **Unterstützung regulärer Ausdrücke** SQL Anywhere bietet nun mit den zwei neuen Suchbedingungen REGEXP und SIMILAR TO Unterstützung für reguläre Ausdrücke. Siehe „[REGEXP-Suchbedingung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[SIMILAR TO-Suchbedingung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Siehe „Überblick über reguläre Ausdrücke“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „Die Suchbedingungen LIKE, REGEXP und SIMILAR TO“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Einstellungen von Datenbankoptionen werden nun im Transaktionslog aufgezeichnet** Die während eines LOAD-Vorgangs gültigen Datenbankoptionseinstellungen werden nun im Transaktionslog aufgezeichnet. Dadurch wird sichergestellt, dass die Daten nicht inkonsistent werden, falls Optionen, z.B. date_order und nearest_century, zwischen dem ursprünglichen LOAD-Vorgang und dem LOAD-Vorgang bei der Anwendung des letzten Transaktionslogs während der Wiederherstellung geändert wurden.
- **Erweiterungen für die Verwendung von Indizes durch den Optimierer** Die Indizierungsfähigkeiten von SQL Anywhere wurden auf verschiedene Weise erweitert. Sie müssen ein Upgrade der Datenbank durchführen, um diese neuen Funktionen verwenden zu können. Siehe „SQL Anywhere-Server-Ugrades“ auf Seite 295.

- **Unterstützung des mehrfachen Index-Scans** Der Optimierer wurde erweitert und untersucht nun mehrere Indizes (bis zu vier), um Daten aus einer Basistabelle basierend auf mehreren Prädikaten für diese Tabelle abzurufen. Bislang konnten Sie nur einen Index als Index-Hint für eine Abfrage festlegen. Mithilfe eines neuen Index-Hints in der WITH-Klausel einer SELECT-Anweisung können Sie nun festlegen, dass ein mehrfacher Index-Scan verwendet wird. Siehe „FROM-Klausel“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Ein neuer Tabellenzugriffsalgorithmus, der mehrfache Index-Scan, wurde hinzugefügt.

- **Unterstützung für den reinen Indexabruf** Der Optimierer wurde erweitert und unterstützt nun reine Indexabrufe. Beim reinen Indexabruf wird die Abfrage mithilfe von Daten aus den Indizes ausgeführt. Dabei muss nicht auf entsprechende Zeilen in den Tabellen zugegriffen werden. Der Optimierer führt, wenn möglich, immer einen reinen Indexabruf aus. Mit dem Hint INDEX ONLY { ON | OFF } kann gesteuert werden, ob ein reiner Indexabruf ausgeführt wird. Siehe „FROM-Klausel“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen zum Laden und Entladen von Daten** Folgende Erweiterungen zum Laden und Entladen von Daten wurden hinzugefügt:
 - **Daten von einem Clientcomputer herunterladen und Daten auf einen Clientcomputer entladen** SQL-Anweisungen und Funktionen werden zum Lesen und Schreiben von Daten verwendet, die sich auf dem Datenbankserver befinden. Es wurden neue Funktionen implementiert, die diese Funktionalität auf Dateien erweitern, die auf dem Clientcomputer gespeichert sind, ohne Clientdateien auf den Datenbankservercomputer kopieren zu müssen. Die Datenübertragung wird effizient ausgeführt, wobei die Sicherheit und die Zugriffssteuerung für Daten auf dem Clientcomputer gewährleistet sind.

Das tatsächliche Lesen von Dateien auf dem Clientcomputer wird von den Clientbibliotheken auf transparente Weise durchgeführt. Das bedeutet, dass vorhandene Clientanwendungen die Vorteile dieser neuen Funktion mithilfe der SQL-Sprachunterstützung sofort nutzen können.

Zur Nutzung dieser neuen Funktionen müssen sowohl der Client als auch der Datenbankserver die SQL Anywhere Version 11.0.0 verwenden. Außerdem muss der Client das Command Sequence-Kommunikationsprotokoll (CmdSeq) verwenden.

Siehe „Zugriff auf Daten auf Clientcomputern“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

- **Daten in eine Variable entladen** Die UNLOAD-Anweisung wurde erweitert und bezieht nun die INTO VARIABLE-Klausel ein, die es gestattet, Daten in eine Variable zu entladen. Siehe „UNLOAD-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Daten aus einer Spalte in einer anderen Tabelle laden** Die LOAD TABLE-Anweisung wurde erweitert und umfasst nun die USING COLUMN-Klausel. Mit ihr können Daten aus einer Spalte einer anderen Tabelle geladen werden. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „Datenimport mit der LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Zur Nutzung dieser neuen Funktion ist ein Datenbank-Upgrade erforderlich. Siehe „SQL Anywhere-Server-Ugrades“ auf Seite 295.

- **Daten aus einem Wert laden (BLOB)** Die LOAD TABLE-Anweisung wurde erweitert und umfasst nun die USING VALUE-Klausel. Mit ihr können Daten aus einem Wertausdruck geladen werden, wie etwa die Ergebnisse einer Funktion oder Systemprozedur. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „Datenimport mit der LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Zur Nutzung dieser neuen Funktion ist ein Datenbank-Upgrade erforderlich. Siehe „SQL Anywhere-Server-Ugrades“ auf Seite 295.

- **Erweiterungen für die Wiederherstellung mit der LOAD TABLE-Anweisung und die Spiegelung** Bislang wurde in einer gespiegelten Datenbankkonfiguration das Laden von Daten aus einer Datei mit der LOAD TABLE-Anweisung nicht unterstützt. Dies lag daran, dass nur die LOAD TABLE-Anweisung im Transaktionslog gespeichert wurde und nicht die geladenen Daten. Zudem konnten Daten, die mit einer LOAD TABLE-Anweisung geladen worden waren, nicht wiederhergestellt werden, wenn die ursprüngliche Ladedatei bei der Wiederherstellung nicht vorhanden war.

Die LOAD TABLE-Anweisung wurde erweitert und umfasst nun drei neue Protokolloptionen: WITH CONTENT LOGGING, WITH ROW LOGGING und WITH FILE NAME LOGGING. Mit diesen Klauseln können Sie steuern, ob die geladenen Daten im Transaktionslog aufgezeichnet werden. In einem Datenbankspiegelungssystem können diese Daten verwendet werden, um die Spiegeldatenbank zu laden. Außerdem muss die Ladedatei bei der Wiederherstellung nicht mehr vorhanden sein. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Zur Nutzung dieser Funktion ist ein Datenbank-Upgrade erforderlich. Siehe „SQL Anywhere-Server-Ugrades“ auf Seite 295.

- **Erweiterungen materialisierter Ansichten** Die Unterstützung für materialisierte Ansichten wurde folgendermaßen erweitert:
 - **Unterstützung für materialisierte Sofortansichten** Sie können nun materialisierte Ansichten so konfigurieren, dass sie sofort aktualisiert werden, wenn Datenänderungen in den zugrunde liegenden Tabellen Auswirkungen auf die materialisierte Ansicht haben. Ansichten mit diesem Aktualisierungstyp werden als **Sofortansichten** bezeichnet. Nicht sofort aktualisierte

Ansichten werden nun als **manuelle Ansichten** bezeichnet. Materialisierte Ansichten, die vor dieser Version erstellt wurden, werden als manuelle Ansichten behandelt. Neu erstellte materialisierte Ansichten sind standardmäßig ebenfalls manuelle Ansichten.

Weitere Hinweise zu manuellen und Sofortansichten finden Sie unter „[Materialisierte Ansichten](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)] und „[Einstellen des Aktualisierungstyps auf "Manuell" oder "Sofort"](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Zur Verwendung der Systemprozeduren, die diese Funktion unterstützen, ist ein Datenbank-Upgrade erforderlich. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

- **Gleichzeitige Aktualisierbarkeit mehrerer materialisierter Ansichten** Bislang mussten materialisierte Ansichten einzeln aktualisiert werden. Änderungen der zugrunde liegenden Daten zwischen den einzelnen Aktualisierungsvorgängen konnten zu Inkonsistenzen zwischen den materialisierten Ansichten führen. Nun können Sie zum Aktualisieren mehrerer materialisierter Ansichten mit den gleichen Daten eine Liste materialisierter Ansichten für die Anweisung REFRESH MATERIALIZED VIEW festlegen. Siehe „[REFRESH MATERIALIZED VIEW-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[Materialisierten Ansichten manuell aktualisieren](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Neue WITH SHARED MODE-Klausel für die REFRESH MATERIALIZED VIEW-Anweisung** Die neue WITH SHARE MODE-Klausel wurde der REFRESH MATERIALIZED VIEW-Anweisung hinzugefügt. Dieser Modus bietet anderen Transaktionen während des Aktualisierungsvorgangs Lesezugriff auf zugrunde liegende Tabellen. Wenn diese Klausel angegeben wird, werden vor dem Ausführen des Aktualisierungsvorgangs gemeinsam genutzte Tabellensperren für alle Basistabellen gesetzt. Der Standardmodus ist nun WITH SHARE MODE, außer die materialisierte Ansicht ist als IMMEDIATE REFRESH festgelegt oder die Snapshot-Isolation ist für die Datenbank aktiviert. Weitere Hinweise zum Standardaktualisierungsverhalten finden Sie unter „[REFRESH MATERIALIZED VIEW-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Unterstützung für die Abfrage einer Datei oder BLOB-Zeichenfolge** Mit der neuen OPENSTRING-Unterklausel der FROM-Klausel können Sie nun Daten aus einer Datei oder BLOB-Zeichenfolge abfragen. Mit der OPENSTRING-Klausel können Sie neben dem abzufragenden Objekt auch das Schema sowie andere syntaktische Analyseinformationen für die Daten festlegen. Siehe „[FROM-Klausel](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Wenn der OPENSTRING-Vorgang ausgeführt wird, erscheint im Ausführungsplan das neue Planobjekt OpenString.

- **Verbesserte Unterstützung von komprimierten Indizes** Aufgrund der verbesserten Unterstützung von komprimierten Indizes ist eine neu aufgesetzte Datenbank, nachdem sie entladen und neu geladen wurde, möglicherweise kleiner als die ursprüngliche Datenbank. Die geringere Datenbankgröße ist *kein* Hinweis auf ein Problem oder einen Datenverlust.
- **Reiner Lesezugriff für auf einem Spiegelserver ausgeführte Datenbanken** Wenn Sie die Datenbankspiegelung verwenden, können Sie nun eine Verbindung mit der Datenbank herstellen, die auf dem Spiegelserver ausgeführt wird. Auf diese Weise können Sie Berichterstellungsvorgänge, die möglicherweise ressourcenintensiv sind, auf den Spiegelserver übertragen, während der Primärserver

weiterhin verfügbar ist. Sie können eine Verbindung mit einer Spiegeldatenbank herstellen, indem Sie einen Datenbankservernamen mit der Serveroption `-sm` angeben. Mithilfe dieser Option können Sie auf die schreibgeschützte Spiegeldatenbank zugreifen. Siehe „[Datenbankoption -sm \(nicht mehr empfohlen\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Datenbankverbindungen

Die folgende Liste enthält die Erweiterungen für die Datenbankverbindungen in SQL Anywhere Version 11.0.0.

- **Erweiterungen der AppInfo-Verbindungsparameter** Der AppInfo-Verbindungsparameter unterstützt nun den Schlüssel `OSUSER`. Dieser Schlüssel gibt den Betriebssystem-Benutzernamen zurück, der dem Clientprozess zugewiesen ist. Linux und Solaris unterstützen nun den `EXE`-Schlüssel. Siehe „[Verbindungsparameter AppInfo \(APP\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- **Verbindungsparameter Elevate** Der Elevate-Verbindungsparameter erhöht die Rechte des automatisch gestarteten SQL Anywhere-Datenbankservers auf Windows Vista. Siehe „[Verbindungsparameter Elevate](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- **NewPassword-Verbindungsparameter [NEWPWD]** Der neue Verbindungsparameter `NewPassword` ermöglicht es Benutzern, Kennwörter ohne Unterstützung des DBAs zu ändern, selbst wenn diese abgelaufen sind. Siehe „[Verbindungsparameter NewPassword \(NEWPWD\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- **Prefetch-Erweiterungen** Die Standardwerte für den PrefetchBuffer-Verbindungsparameter (`PBUF`) wurden geändert. Auf Windows Mobile ist der Standardwert nun 64 kB. Auf allen anderen Plattformen beträgt der Standardwert 512 kB. Dieser Verbindungsparameter akzeptiert Werte zwischen 64 kB und 8 MB. Siehe „[Verbindungsparameter PrefetchBuffer \(PBUF\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

In früheren Versionen basierte die maximale Anzahl der vorab abgerufenen Zeilen auf der maximal vorab abrufbaren Datenmenge. Nun berücksichtigt die maximale Anzahl vorab abgerufener Zeilen die tatsächlich vorab abgerufene Datenmenge sowie den vom PrefetchBuffer-Verbindungsparameter festgelegten Datengrenzwert. Das kann zu einer erheblichen Performancesteigerung führen, wenn die Datenmenge in einer Spalte deutlich geringer ist als die Länge der Hostvariablen und `DESCRIBE`-Länge.

Der Prefetch erhöht auch in Fällen, die wahrscheinlich zu einer Performancesteigerung führen, die Anzahl der Prefetch-Zeilen dynamisch. Siehe „[Prefetch](#)“ [*SQL Anywhere Server - Programmierung*].

Sicherung und Wiederherstellung

Die folgende Liste enthält alle Erweiterungen zur Sicherung und Wiederherstellung in SQL Anywhere Version 11.0.0.

- **Unterstützung des Microsoft Volumeschattenkopie-Diensts (VSS)** SQL Anywhere ist mit VSS (Microsoft Volumeschattenkopie-Dienst) kompatibel. Zur Verwendung von VSS müssen Sie alle

vorhandenen Datenbanken neu erstellen. Siehe „[SQL Anywhere-Volumeschattenkopie-Dienst \(VSS\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Sicherheit

Die folgende Liste enthält alle Erweiterungen zur Sicherheit in SQL Anywhere Version 11.0.0.

- **Die Systemtabellen ISYSUSER und ISYSEXTNLOGIN werden nun verschlüsselt, wenn die Tabellenverschlüsselung aktiviert ist.** Bislang wurde die Systemtabelle ISYSCOLSTAT beim Verschlüsseln einer Datenbank oder beim Erstellen einer Datenbank mit aktivierter Tabellenverschlüsselung automatisch verschlüsselt. Nun werden auch die Systemtabellen ISYSUSER und ISYSEXTNLOGIN verschlüsselt, um zusätzliche Sicherheit bereitzustellen.
- **Auditing-Erweiterungen** Nun kann das Auditing von Sybase Central gesteuert werden. Im Fenster **Datenbankeigenschaften** können Sie das Auditing aktivieren und deaktivieren sowie festlegen, welche Informationen überprüft werden sollen. Auditing-Informationen werden in Sybase Central im Register **Auditing** im rechten Fensterausschnitt angezeigt. Siehe „[Auditing konfigurieren \(Sybase Central\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[Audit-Daten \(Sybase Central\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Wenn das Auditing aktiviert ist, werden nun Fehler für fehlgeschlagene Verbindungen zusammen mit der jeweiligen Ursache protokolliert.

- **Unterstützung für 256-Bit-AES-Verschlüsselung** SQL Anywhere unterstützt nun die 256-Bit-AES-Verschlüsselung für Datenbanken, Tabellen, Dateien und Daten. Diese Erweiterung hat Auswirkungen auf mehrere Bereiche, die im Folgenden beschrieben sind:
 - **Datenbank- und Tabellenverschlüsselung** Sie können nun AES256 und AES256_FIPS für die ENCRYPTION-Klausel der CREATE DATABASE-Anweisung festlegen. Siehe „[CREATE DATABASE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Sie können AES256 und AES256_FIPS auch für die Option -ea des Initialisierungs-Dienstprogramms (dbinit) und des Entlade-Dienstprogramms (dbunload) festlegen. Siehe „[Dienstprogramm Initialisierung \(dbinit\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **FIPS-zertifizierte Algorithmen** Sie können nun 256-Bit FIPS-zertifizierte AES-Algorithmen verwenden. Siehe „[Datenbankserveroption -fips](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Daten verschlüsseln und entschlüsseln** Beim Verschlüsseln von Daten mit den Funktionen ENCRYPT und DECRYPT können Sie nun AES256 und AES256_FIPS angeben. Siehe „[ENCRYPT-Funktion \[Zeichenfolge\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[DECRYPT-Funktion \[Zeichenfolge\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Verschlüsselte Kopien von Datenbanken, Transaktionslogs und DBSpaces erstellen** Wenn Sie eine verschlüsselte Kopie einer verschlüsselten oder unverschlüsselten Datenbank, eines Transaktionslogs oder eines DBSpace mit der Anweisung CREATE ENCRYPTED FILE erstellen, können Sie nun einen 256-Bit-AES-Algorithmus festlegen

(AES256 oder AES256_FIPS). Siehe „[CREATE ENCRYPTED FILE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **DBTools-Unterstützung für 256-Bit-AES-Verschlüsselung** Die Strukturen a_create_db und an_unload_db wurden erweitert und unterstützen nun AES256 und AES256_FIPS als Werte für das Mitglied encryption_algorithm. Siehe [a_create_db-Struktur](#) [[Datenbanktools](#)] [[SQL Anywhere Server - Programmierung](#)] und [an_unload_db-Struktur](#) [[Datenbanktools](#)] [[SQL Anywhere Server - Programmierung](#)].

Siehe auch:

- „[Datenbankverschlüsselung und -entschlüsselung](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „[Liste der Datenbankeneigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- **Unterstützung der Kennwortverschlüsselung für jConnect und Open Client** Die Kennwortverschlüsselung wird nun für jConnect- und Open Client-Verbindungen unterstützt. Siehe:
 - „[jConnect-JDBC-Treiber](#)“ [[SQL Anywhere Server - Programmierung](#)]
 - „[Deployment von JDBC-Clients](#)“ [[SQL Anywhere Server - Programmierung](#)]
 - „[Bekannte Open Client-Einschränkungen von SQL Anywhere](#)“ [[SQL Anywhere Server - Programmierung](#)]

Sicherheitsänderungen

Sie müssen ein Upgrade der Datenbank durchführen, um diese Änderungen verwenden zu können. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

- **Unterstützung der Vererbung** SQL Anywhere unterstützt nun die Vererbung der Datenbankberechtigungen PROFILE, READCLIENTFILE, READFILE und WRITECLIENTFILE.
- **Neue Datenbankberechtigungen:**
 - PROFILE
 - READFILE
 - READCLIENTFILE
 - WRITECLIENTFILE
- **Unterstützung des CREATE-Privilegs für DBSpaces** SQL Anywhere unterstützt nun die CREATE ON-Berechtigung für die Erstellung eines Datenbankobjekts im angegebenen DBSpace. Das CREATE-Privileg kann dem Benutzer zugewiesen oder über die Gruppenmitgliedschaft vererbt werden.

Datenbank-Dienstprogramme

Die folgende Liste enthält die Erweiterungen der Datenbank-Dienstprogramme in SQL Anywhere Version 11.0.0.

- **Erweiterung der Konfigurationsdatei** Sie können nun mit dem Zeichen "&" in Konfigurationsdateien darauf hinweisen, dass das vorherige Token auf der nächsten Zeile fortgesetzt wird. Siehe „Konfigurationsdateien“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen des Dienstprogramms Entladen (dbunload)** Die folgenden Erweiterungen wurden für dbunload hinzugefügt:
 - Die neue Option -cp wurde hinzugefügt. Mit ihr können Sie festlegen, dass die Datenausgabedatei von dbunload komprimiert wird.
 - Bislang wurde ein Fehler zurückgegeben, wenn die Verschlüsselungsoptionen -ek, -ep oder -ea ohne die Option -an oder -ar verwendet wurden. Nun akzeptiert dbunload die Verschlüsselungsoptionen und wendet sie auf die erstellte Ausgabedatei an.
 - Die Option -g aktualisiert nun Textindizes, für die MANUAL REFRESH festgelegt ist. Siehe „Konzepte und Referenz zu Textindizes“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
 - Standardmäßig werden Textindizes, für die MANUAL REFRESH festgelegt ist, beim Aktualisierungsvorgang nicht initialisiert. Zum Aktualisieren dieser Textindizes können Sie die dbunload-Option -g verwenden.
 - Mit der Option -no können Sie Objektdefinitionen in alphabetischer Reihenfolge gruppiert nach Objekttyp entladen. Dies kann nützlich sein, wenn die *reload.sql*-Dateien für Datenbanken verglichen werden.

Siehe „Dienstprogramm zum Entladen (dbunload)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Erweiterungen des Validierungs-Dienstprogramms (dbvalid)** Bislang validierte das Dienstprogramm dbvalid standardmäßig alle Tabellen und materialisierten Ansichten. Nun führt dbvalid auch die VALIDATE DATABASE-Anweisung aus.

Wenn eine Datenbank mithilfe des Validierungs-Dienstprogramms automatisch gestartet wird, wird sie im schreibgeschützten Modus ausgeführt. Auf diese Weise wird verhindert, dass die Datenbank geändert wird, während sie im Rahmen eines Sicherungs- und Wiederherstellungsplans validiert wird.

Siehe „Validierungs-Dienstprogramm (dbvalid)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Verwalten der generierten .sql-Datei, wenn das Logkonvertierungs-Dienstprogramm (dbtran) eine Beschädigung feststellt** Mit der Option -k des Logkonvertierungs-Dienstprogramms können Sie festlegen, dass partielle .sql-Dateien gelöscht werden sollen, wenn die Konvertierung der Logdatei aufgrund einer Beschädigung der Transaktionslogdatei fehlschlägt. Siehe „Dienstprogramm zur Logkonvertierung (dbtran)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbankoptionen

Die folgende Liste enthält die Erweiterungen der Datenbankoptionen in SQL Anywhere Version 11.0.0.

- **allow_read_client_file-Option** Diese Option steuert, ob das Lesen von Dateien auf einem Clientcomputer zugelassen ist. Siehe „allow_read_client_file-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **allow_write_client_file-Option** Diese Option steuert, ob das Schreiben von Dateien auf einen Clientcomputer zugelassen ist. Siehe „allow_write_client_file-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **login_procedure-Option** Sie können jetzt eine Fehlermeldung über ein abgelaufenes Kennwort ausgeben. Siehe „login_procedure-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **max_priority-Option** Die Option steuert die maximale Prioritätsstufe für Datenbankverbindungen. Siehe „max_priority-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **priority-Option** Diese Option steuert die Prioritätsstufe, in der Anforderungen einer Verbindung ausgeführt werden. Siehe „priority-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **query_mem_timeout-Option** Diese Option steuert, wie lange eine Anforderung auf eine Speicherzuordnung wartet. Siehe „query_mem_timeout-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].

Datenbankserveroptionen

Die folgende Liste enthält die Erweiterungen der Datenbankserveroptionen in SQL Anywhere Version 11.0.0.

- **Serveroption -es** Wenn in früheren Versionen von SQL Anywhere der Datenbankserver mit der Option -ec gestartet wurde (zur Unterstützung der Transportschichtssicherheit), aber die Liste zugelassener Verschlüsselungsprotokolle NONE oder SIMPLE nicht enthielt, wurde der Shared Memory-Port nicht gestartet, da er keine Transportschichtssicherheit unterstützt. Das bedeutete, dass alle Verbindungen zum Datenbankserver über TCP/IP mit starker Verschlüsselung hergestellt werden mussten.

Die Serveroption -es weist den Datenbankserver an, nicht verschlüsselte Datenbankverbindungen über Shared Memory zuzulassen. Siehe „Datenbankserveroption -es“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Serveroption -gb** Die Serveroption -gb zum Steuern der Prioritätsklasse des Serverprozesses wird nun unter UNIX ebenso wie unter Windows unterstützt. Siehe „Datenbankserveroption -gb“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Serveroption -im** Sie können die Datenbank vollständig im Speicher ausführen, wenn der Verlust aller Datenbankvorgänge für die Anwendung akzeptabel ist. Diese Funktion ist für Situationen geeignet, in denen SQL Anywhere als schneller, temporärer Datenspeicher verwendet wird, in den Daten in einem hohen Tempo eingefügt werden. Siehe „Datenbankserveroption -im“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Aus Dateien auf einem Clientcomputer lesen und in sie schreiben** Mit der Serveroption -sf kann nun die Fähigkeit gesteuert werden, aus Dateien auf einem Clientcomputer zu lesen und in diese

Dateien zu schreiben. Siehe „[Datenbankserveroption -sf](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Serveroption -um** Die Option -um ermöglicht eine Verbindung mit der *DBLauncher.app*-Instanz, wenn diese läuft, und zeigt Datenbankservermeldungen in einem neuen Fenster in *DBLauncher.app* an. Diese Option ist nur für Mac OS X verfügbar. Siehe „[Datenbankserveroption -um](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Optionen des Windows-Systemmonitors** Die folgenden Serveroptionen wurden hinzugefügt, um den Windows-Systemmonitor weiter zu konfigurieren:
 - **Option -ks** Deaktiviert die Erstellung von Shared Memory, das vom Systemmonitor verwendet wird, um Leistungsindikatoren vom Datenbankserver zu sammeln. Siehe „[Datenbankserveroption -ks](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Option -ksc** Legt die maximale Anzahl von Verbindungen fest, die der Systemmonitor überwachen kann. Siehe „[Datenbankserveroption -ksc](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Option -ksd** Legt die maximale Anzahl von Datenbanken fest, die der Systemmonitor überwachen kann. Siehe „[Datenbankserveroption -ksd](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Eigenschaften und Systemmonitorstatistiken

In der folgenden Liste sind die Erweiterungen von Eigenschaften und Systemmonitorstatistiken in SQL Anywhere Version 11.0.0 aufgeführt.

- **Neue Verbindungseigenschaften** Folgende Verbindungseigenschaften wurden in dieser Version hinzugefügt:

- allow_read_client_file
- allow_write_client_file
- AuthType
- CacheReadWorkTable
- ClientNodeAddress
- DiskReadWorkTable
- DiskSyncRead
- DiskSyncWrite
- DiskWaitRead
- DiskWaitWrite
- DiskWriteHint
- DiskWriteHintPages
- LockIndexID
- LockRowID
- max_priority
- OSUser
- priority
- query_mem_timeout
- QueryMemActiveCurr
- QueryMemExtraAvail
- QueryMemGrantFailed
- QueryMemGrantGranted
- QueryMemGrantWaiting
- QueryMemGrantRequested
- QueryMemWaited
- ServerNodeAddress
- ReadHint
- ReadHintScatter

Eine Beschreibung dieser Eigenschaften finden Sie unter „[Liste der Verbindungseigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Neue Datenbankservereigenschaften** Folgende Datenbankservereigenschaften wurden in dieser Version hinzugefügt:

- DiskRetryRead
- DiskRetryReadScatter
- DiskRetryWrite
- EventTypeDesc
- EventTypeName
- HttpAddresses
- HttpsAddresses
- HttpNumActiveReq
- HttpNumConnections
- HttpNumSessions
- HttpsNumActiveReq
- HttpsNumConnections
- MaxEventType
- MaxRemoteCapability
- MessageCategoryLimit
- OptionWatchAction
- OptionWatchList
- QueryMemActiveCurr
- QueryMemActiveEst
- QueryMemActiveMax
- QueryMemExtraAvail
- QueryMemGrantBase
- QueryMemGrantBaseMI
- QueryMemGrantExtra
- QueryMemGrantFailed
- QueryMemGrantGranted
- QueryMemGrantWaiting
- QueryMemGrantRequested
- QueryMemPages
- QueryMemPercentOfCache
- QueryMemWaited
- ReadHintScatterLimit
- RemoteCapability
- StreamsUsed
- TcpIpAddresses
- WebClientLogFile
- WebClientLogging

Eine Beschreibung dieser Eigenschaften finden Sie unter „[Liste der Datenbankservereigenschaften](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Neue Datenbankeigenschaften** Folgende Datenbankeigenschaften wurden in dieser Version hinzugefügt:

- AlternateMirrorServerName
- CacheReadWorkTable
- DiskReadWorkTable
- DiskRetryReadScatter
- DiskSyncRead
- DiskSyncWrite
- DiskWaitRead
- DiskWaitWrite
- DiskWriteHint
- DiskWriteHintPages
- HasEndianSwapFix
- MirrorMode
- ReadHint
- ReadHintScatter

Eine Beschreibung dieser Eigenschaften finden Sie unter „Liste der Datenbankeigenschaften“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Neue Systemmonitorstatistiken** In dieser Version wurden die folgenden Systemmonitorstatistiken hinzugefügt:

- Cachelesevorgänge: Arbeitstabelle
- Platten-Lesezugriffe: Arbeitstabelle

Systemprozeduren und Funktionen

In der folgenden Liste sind die Erweiterungen der Systemprozeduren und Funktionen in SQL Anywhere Version 11.0.0 aufgeführt.

- **sa_get_dtt_groupreads-Systemprozedur** Mit der neuen sa_get_dtt_groupreads-Systemprozedur können Sie schätzen, wie hoch die Kosten der Durchführung von Gruppenlesevorgängen auf einem Datenbankserver sind. Siehe „sa_get_dtt_groupreads-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der PROPERTY_NAME-Funktion** Die Funktion gibt nun den Namen der Eigenschaft mit der Eigenschafts-ID für die festgelegte Verbindungsstufe zurück. Siehe „PROPERTY_NAME-Funktion [System]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **READ_CLIENT_FILE-Funktion** Die neue Funktion READ_CLIENT_FILE liest Daten aus der angegebenen Datei auf dem Clientcomputer. Siehe „READ_CLIENT_FILE-Funktion“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **WRITE_CLIENT_FILE-Funktion** Die neue Funktion WRITE_CLIENT_FILE schreibt Daten in die angegebene Datei auf dem Clientcomputer. Siehe „WRITE_CLIENT_FILE-Funktion [Zeichenfolge]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **REGEXP_SUBSTR-Funktion** Mit der neuen Funktion REGEXP_SUBSTR können Sie nach einer Teilzeichenfolge innerhalb einer Zeichenfolge suchen. Diese neue Funktion hat einen regulären Ausdruck als Argument. Siehe „[REGEXP_SUBSTR-Funktion \[Zeichenfolge\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_char_terms-Systemprozedur** Die neue Systemprozedur sa_char_terms teilt eine CHAR-Zeichenfolge in Begriffe auf und gibt jeden Begriff zusammen mit seiner Position zurück. Siehe „[sa_char_terms-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_nchar_terms-Systemprozedur** Die neue Systemprozedur sa_nchar_terms teilt eine NCHAR-Zeichenfolge in Begriffe auf und gibt jeden Begriff zusammen mit seiner Position zurück. Siehe „[sa_nchar_terms-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_refresh_text_indexes-Systemprozedur** Die neue Systemprozedur sa_refresh_text_indexes aktualisiert alle Textindizes, die als MANUAL REFRESH oder AUTO REFRESH definiert sind. Siehe „[sa_refresh_text_indexes-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_text_index_stats-Systemprozedur** Die neue Systemprozedur sa_text_index_stats gibt Statistikinformationen für alle Textindizes in der Datenbank zurück, einschließlich des letzten Aktualisierungszeitpunkts und der Größe ausstehender Änderungen. Siehe „[sa_text_index_stats-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_text_index_vocab-Systemprozedur** Die neue Systemprozedur sa_text_index_vocab listet alle in einem Textindex enthaltenen Begriffe sowie die Gesamtanzahl von indizierten Werten auf, in denen die einzelnen Begriffe erscheinen. Siehe „[sa_text_index_vocab-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Die beiden neuen Systemprozeduren sa_internal_text_index_vocab und sa_internal_text_index_postings wurden ebenfalls hinzugefügt. Sie sind jedoch nur für die Verwendung durch die Systemprozedur sa_text_index_vocab vorgesehen.

- **sa_text_index_postings-Systemprozedur** Diese neue Systemprozedur ist nur für den internen Gebrauch gedacht.
- **sa_text_index_handles-Systemprozedur** Diese neue Systemprozedur ist nur für den internen Gebrauch gedacht.
- **sa_get_user_status-Systemprozedur** Mit der neuen Systemprozedur sa_get_user_status können Sie den aktuellen Login-Status eines Benutzers ermitteln. Siehe „[sa_get_user_status-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Laufende Prozeduren und Funktionen als Aufrufer** Beim Erstellen einer Prozedur oder Funktion können Sie nun festlegen, ob die Prozedur oder Funktion mit den Berechtigungen des aufrufenden Benutzers (Aufrufers) ausgeführt werden oder mit denjenigen des Eigentümers (Definierers). Hierzu müssen Sie die SQL SECURITY-Klausel der Anweisung CREATE PROCEDURE oder CREATE FUNCTION verwenden. Siehe „[CREATE FUNCTION-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[CREATE PROCEDURE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

Diese Änderung gilt auch für externe Prozeduren und Funktionen.

- **sa_disk_free_space-Systemprozedur** Die Systemprozedur sa_disk_free_space gibt nun die neue Spalte total_space zurück. Diese Spalte gibt den Gesamt-Speicherplatz an, der auf dem Laufwerk verfügbar ist, auf dem der DBSpace gespeichert ist. Bei Datenbanken, die mit älteren Versionen als SQL Anywhere 11.0.0 erstellt wurden, wird die Spalte total_space nur zurückgegeben, wenn ein Upgrade der Datenbank durchgeführt wurde. Siehe „sa_disk_free_space-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_external_library_unload-Systemprozedur** Mit der neuen Systemprozedur sa_external_library_unload können Sie externe, nicht verwendete Bibliotheken entladen. Siehe „sa_external_library_unload-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_index_density-Systemprozedur gibt nun Schiefe (Skew) zurück** Die neue Systemprozedur sa_index_density gibt die im Index vorhandene Schiefe (Skew) zurück. Ein hohes Maß an Schiefe kann negative Auswirkungen auf die Performance haben (verglichen mit einem ausgeglichenen Index). Siehe „Indexfragmentierung und -schiefe (Skew) reduzieren“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)] und „sa_index_density-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der sa_materialized_view_info-Systemprozedur** Die Informationen in der von sa_materialized_view_info zurückgegebenen Spalte "Status" wurden in die beiden Spalten "Status" und "DataStatus" aufgeteilt. Die Spalte "Status" gibt nun Informationen darüber zurück, ob die Ansicht aktiviert oder deaktiviert ist. Die neue Spalte "DataStatus" gibt Informationen darüber zurück, ob die Ansicht Daten enthält und wie aktuell die Daten sind. Außerdem wurde die Spalte "RefreshType" hinzugefügt. Sie gibt an, ob es sich um eine manuelle Ansicht oder eine Sofortansicht handelt. Siehe „sa_materialized_view_info-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **sa_materialized_view_can_be_immediate-Systemprozedur** Neu erstellte materialisierte Ansichten sind standardmäßig manuelle Ansichten. Sie können jedoch in Sofortansichten geändert werden, sofern sie keine Einschränkungen für Sofortansichten verletzen. Mit der neuen Systemprozedur sa_materialized_view_can_be_immediate können Sie testen, ob eine manuelle Ansicht in eine Sofortansicht geändert werden kann. Siehe „sa_materialized_view_can_be_immediate-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und Einschränkungen beim Ändern des Typs einer materialisierten Ansicht von "Manuell" auf "Sofort" [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **sa_post_login_procedure-Systemprozedur** Es wurde eine neue Systemprozedur hinzugefügt, mit der festgelegt werden kann, ob eine Warnung ausgegeben werden soll, dass das Kennwort eines Benutzers demnächst abläuft. Siehe „sa_post_login_procedure-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der Funktion EVENT_PARAMETER** Die Funktion EVENT_PARAMETER unterstützt nun den Wert "abnormal" für die Eigenschaft DisconnectReason. Dieser neue Wert zeigt an, dass eine Trennung einer Verbindung aufgetreten ist, weil entweder die Clientanwendung vor der Trennung von der Datenbank abnormal beendet wurde oder weil es einen Verbindungsfehler zwischen dem Client- und dem Servercomputer gegeben hat. Siehe „EVENT_PARAMETER-Funktion [System]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der sa_server_option-Systemprozedur** Die beiden neuen Eigenschaften OptionWatchList und OptionWatchAction wurden der Systemprozedur sa_server_option hinzugefügt.

Sie können mit diesen Eigenschaften überwachen, wann versucht wird, die Einstellung einer Datenbankoption zu ändern, und die durchzuführende Aktion festlegen. Siehe „[Optionseinstellungen überwachen](#)“ [*SQL Anywhere Server - Datenbankadministration*] und „[sa_server_option-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

- **Erweiterungen der sa_db_properties-Systemprozedur** Die Systemprozedur sa_db_properties gibt nun gültige Eigenschaften zurück, die NULL enthalten. Siehe „[sa_db_properties-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Erweiterungen der sa_conn_properties-Systemprozedur** Die Systemprozedur sa_conn_properties gibt nun gültige Eigenschaften zurück, die NULL enthalten. Siehe „[sa_conn_properties-Systemprozedur](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

SQL-Anweisungen

Die folgende Liste enthält alle SQL-Erweiterungen in SQL Anywhere Version 11.0.0.

- **Neue CALIBRATE GROUP READ-Klausel, ALTER DATABASE-Anweisung** Mit der neuen CALIBRATE GROUP READ-Klausel der Anweisung ALTER DATABASE können Sie Gruppenlesekalibrierung im temporären DBSpace durchführen. Siehe „[ALTER DATABASE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Neue CHECK-Klausel-Anweisung CREATE MATERIALIZED VIEW** Mit der neuen CHECK-Klausel der Anweisung CREATE MATERIALIZED VIEW können Sie die Anweisung vor der Erstellung der Ansicht validieren. Siehe „[CREATE MATERIALIZED VIEW-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Neue RECOMPILE-Klausel-Anweisung ALTER FUNCTION** Mit der neuen RECOMPILE-Klausel der Anweisung ALTER FUNCTION können Sie eine benutzerdefinierte Funktion neu kompilieren. Siehe „[ALTER FUNCTION-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Neue RECOMPILE-Klausel-Anweisung ALTER PROCEDURE** Mit der neuen RECOMPILE-Klausel der Anweisung ALTER PROCEDURE können Sie eine gespeicherte Prozedur neu kompilieren. Siehe „[ALTER PROCEDURE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Neue REFRESH-Klausel-Anweisung ALTER MATERIALIZED VIEW** Mit der neuen REFRESH-Klausel der Anweisung ALTER MATERIALIZED VIEW können Sie den Aktualisierungstyp für die materialisierte Ansicht festlegen. Siehe „[ALTER MATERIALIZED VIEW-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Erweiterungen der Anweisung LOAD TABLE zur Unterstützung von Wiederherstellung und Spiegelung** Die folgenden Klauseln wurden der Anweisung LOAD TABLE zur Unterstützung von Wiederherstellung und Spiegelung hinzugefügt:
 - **WITH CONTENT LOGGING-Klausel** Die WITH CONTENT LOGGING-Klausel weist den Datenbankserver an, den Inhalt der Datenquelle im Transaktionslog aufzuzeichnen. Die Daten werden in kleinen Abschnitten gespeichert, da die Eingabe von LOAD TABLE verarbeitet wird.

Die Abschnitte können von einer Spiegeldatenbank oder bei der Wiederherstellung mithilfe des Transaktionslogs wieder zu Zeilen zusammengefügt werden. Die WITH CONTENT LOGGING-Klausel kann nützlich sein, wenn die Originaldateien nicht für eine spätere Wiederherstellung verwaltet werden sollen. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **WITH ROW LOGGING-Klausel** Die WITH ROW LOGGING-Klausel weist den Datenbankserver an, alle geladenen Zeilen als eine Reihe von INSERT-Anweisungen zu speichern. Diese Stufe ist besonders für Datenbanken geeignet, die an der Synchronisation beteiligt sind, sowie für Situationen, in denen die Tabelle, in die geschrieben wird, nicht-deterministische Werte enthält, z. B. berechnete Spalten oder CURRENT TIMESTAMP-Standardwerte. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **WITH FILE NAME LOGGING-Klausel** Die WITH FILE NAME LOGGING-Klausel weist den Datenbankserver an, nur die LOAD TABLE-Anweisung aufzuzeichnen. Dies ist das Standardverhalten, das mit dem Protokollierungsverhalten früherer Versionen von SQL Anywhere übereinstimmt. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue Klauseln zum Laden und Entladen von Clientdateien** Die Erweiterungen der Anweisungen LOAD TABLE und UNLOAD TABLE wurden zur Unterstützung der neuen Lade-/Entladefunktion von Clientdateien hinzugefügt:
 - **Neue USING CLIENT FILE-Klausel für die Anweisung LOAD TABLE** Mit dieser Klausel können Sie eine Tabelle mit Daten aus einer Datei auf einem Clientcomputer laden. Siehe „LOAD TABLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Neue INTO CLIENT FILE-Klausel für die Anweisung UNLOAD TABLE** Mit dieser Klausel können Sie eine Datei auf dem Clientcomputer festlegen, in die Daten entladen werden sollen. Siehe „UNLOAD-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue Anweisungen für Login-Richtlinie** Die folgenden Anweisungen wurden zur Unterstützung der neuen Login-Richtlinien-Funktion hinzugefügt:
 - CREATE LOGIN POLICY-Anweisung. Siehe „CREATE LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - ALTER LOGIN POLICY-Anweisung. Siehe „ALTER LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - DROP LOGIN POLICY-Anweisung. Siehe „DROP LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - CREATE USER-Anweisung. Siehe „CREATE USER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - ALTER USER-Anweisung. Siehe „ALTER USER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - DROP USER-Anweisung. Siehe „DROP USER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue Anweisungen und Klauseln für Volltextsuche** Die folgenden Anweisungen wurden zur Unterstützung der neuen Volltext-Suchfunktion hinzugefügt:

- **Neue CONTAINS-Suchbedingung** Mit der CONTAINS-Suchbedingung kann eine angegebene Liste von Spalten auf eine bestimmte Liste von Begriffen oder Satzteilen geprüft werden. Die CONTAINS-Suchbedingung gibt TRUE oder FALSE zurück. Sie können bei der Suche nach mehreren Begriffen oder Satzteilen die gesuchten Elemente durch boolesche Operatoren verknüpfen. Siehe „CONTAINS-Suchbedingung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neue CONTAINS-Klausel in FROM-Klausel einer SELECT-Anweisung** Die CONTAINS-Klausel wird in der FROM-Klausel einer SELECT-Anweisung angegeben. Ihre Funktionsweise ist ähnlich der CONTAINS-Suchbedingung, doch die Klausel gibt auch eine Anzahl für jede übereinstimmende Spalte sowie eine Gesamtzahl für jede übereinstimmende Zeile zurück. Siehe „FROM-Klausel“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **CREATE TEXT CONFIGURATION-Anweisung** Diese Anweisung erstellt ein Textkonfigurationsobjekt. Ein Textkonfigurationsobjekt ist eine Gruppe von Konfigurationseinstellungen, die Merkmale eines Textindexes steuern. Siehe „CREATE TEXT CONFIGURATION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **ALTER TEXT CONFIGURATION-Anweisung** Diese Anweisung ändert ein Textkonfigurationsobjekt. Siehe „ALTER TEXT CONFIGURATION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **DROP TEXT CONFIGURATION-Anweisung** Diese Anweisung löscht ein Textkonfigurationsobjekt. Siehe „DROP TEXT CONFIGURATION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **CREATE TEXT INDEX-Anweisung** Diese Anweisung erstellt einen Textindex. In einem Textindex werden für jede Instanz jedes Begriffs in jeder indizierten Spalte vollständige Positionsinformationen gespeichert. Siehe „CREATE TEXT INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **ALTER TEXT INDEX-Anweisung** Diese Anweisung ändert einen Textindex. Siehe „ALTER TEXT INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **DROP TEXT INDEX-Anweisung** Diese Anweisung entfernt einen Textindex aus der Datenbank. Siehe „DROP TEXT INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **REFRESH TEXT INDEX-Anweisung** Diese Anweisung aktualisiert einen Textindex. Siehe „REFRESH TEXT INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **TRUNCATE TEXT INDEX-Anweisung** Diese Anweisung kürzt die Daten aus einem Textindex. Siehe „TRUNCATE TEXT INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der ALTER EVENT-Anweisung** Sie können nun die Definition für einen Event-Handler mithilfe der Anweisung ALTER EVENT...SET HIDDEN ausblenden. Das Ergebnis dieser Anweisung ist die Verschleierung der Event-Handler-Definition, die in der Aktionsspalte der Systemtabelle ISYSEVENT gespeichert ist. Siehe „ALTER EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **BEGIN SNAPSHOT-Anweisung** Mit der Anweisung BEGIN SNAPSHOT können Sie steuern, wann ein Snapshot für die Snapshot-Isolation startet. Siehe „[BEGIN SNAPSHOT-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen von CASE-Anweisung und CASE-Ausdruck** Für eine verbesserte Kompatibilität können CASE-Anweisungen und CASE-Ausdrücke nun mit END oder END CASE enden. Siehe „[CASE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[CASE-Ausdrücke](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der COMMENT-Anweisung** Sie können nun der Login-Richtlinien-Tabelle sowie einem DBSpace Kommentare hinzufügen. Siehe:
 - „[COMMENT-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[Login-Richtlinien](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „[Zusätzliche Hinweise zu DBSpaces](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- **Erweiterung der CREATE MATERIALIZED VIEW-Anweisung** Sie können nun mit der IMMEDIATE REFRESH-Klausel der Anweisung CREATE MATERIALIZED VIEW eine materialisierte Ansicht erstellen, die jedes Mal aktualisiert wird, wenn sich die zugrunde liegenden Daten ändern. Siehe „[CREATE MATERIALIZED VIEW-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterung der DESCRIBE-Anweisung** Sie können mit der Interactive SQL-Anweisung DESCRIBE nun Informationen über die mit Interactive SQL verbundene Datenbank bzw. den verbundenen Datenbankserver abrufen. Siehe „[DESCRIBE-Anweisung \[Interactive SQL\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen von IF-Anweisung und IF-Ausdruck** Für eine verbesserte Kompatibilität können IF-Anweisungen und IF-Ausdrücke nun mit ENDIF oder END IF beendet werden. Siehe „[IF-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[IF-Ausdrücke](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der LOAD TABLE-Anweisung** Sie können bei der Verwendung der LOAD TABLE-Anweisung mithilfe der Klauseln COMPRESSED und ENCRYPTED nun festlegen, ob die Daten in der Eingabedatei komprimiert und/oder verschlüsselt werden. Siehe „[LOAD TABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Erweiterungen der SELECT-Anweisung**
 - **Erweiterungen der INDEX-Klausel** Sie können bei der Festlegung eines Index-Hints mit der INDEX-Klausel nun bis zu vier Indizes festlegen, die der Datenbankserver verwenden muss. Siehe „[FROM-Klausel](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Neue INDEX ONLY-Klausel** Sie können nun beim Festlegen eines Index-Hints mit der INDEX-Klausel optional die INDEX ONLY-Klausel angeben, um zu steuern, ob der Datenbankserver einen reinen Indexabruf ausführt (d. h. es werden nur Indexdaten für die Ausführung der Abfrage verwendet). Siehe „[FROM-Klausel](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **Neue Klauseln CROSS APPLY und OUTER APPLY** Die SELECT-Anweisung wurde erweitert und unterstützt nun APPLY-Ausdrücke (Klauseln CROSS APPLY und OUTER APPLY) in der FROM-Klausel. Ein APPLY-Ausdruck bietet eine einfache Möglichkeit, Joins anzugeben, bei denen die rechte Seite von der linken Seite abhängig ist. Sie können zum Beispiel einen APPLY-Ausdruck verwenden, um eine Prozedur oder eine abgeleitete Tabelle einmal für jede Zeile in einem Tabellenausdruck auszuwerten. Siehe „Joins, die sich aus APPLY-Ausdrücken ergeben“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)] und „FROM-Klausel“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Neue OPENSTRING-Klausel** Mit der neuen OPENSTRING-Klausel können Sie nun Daten in einer Datei mithilfe einer SELECT-Anweisung abfragen. Siehe „FROM-Klausel“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Beim Erstellen, Ändern, Löschen oder Kommentieren von Ereignissen einen Eigentümer festlegen** Mit den Anweisungen CREATE EVENT, ALTER EVENT, DROP EVENT und COMMENT ON EVENT können Sie nun den Eigentümer angeben. Siehe:
 - „CREATE EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „ALTER EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DROP EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „COMMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - **Erweiterungen der UNLOAD-Anweisung** Sie können nun bei der Verwendung der UNLOAD-Anweisung mit den Klauseln COMPRESSED und ENCRYPTED festlegen, ob die zu entladenden Daten komprimiert bzw. verschlüsselt werden sollen. Siehe „UNLOAD-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- Dateien, die mit diesen Klauseln komprimiert oder verschlüsselt wurden, können nur von Datenbankservern der Version SQL Anywhere 11.0.0 geladen werden (z.B. mit LOAD TABLE). Dateien, die mit anderen Tools komprimiert oder verschlüsselt wurden, können von SQL Anywhere nicht verwendet werden.
- **Erweiterung der UPDATE-Anweisung** Sie können nun sowohl für Suchaktualisierungen als auch für positionierte Aktualisierungen mithilfe der SET-Klausel den Spaltenwert auf seinen Standardwert festlegen. Siehe „UPDATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „UPDATE-Anweisung (positionsbasiert) [ESQL] [SP]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
 - **Erweiterung der OPTION-Klausel** Die OPTION-Klausel für die Anweisungen INSERT, UPDATE, DELETE, SELECT, UNION, EXCEPT und INTERSECT kann nun die Einstellung der Datenbankoption user_estimates aufheben. Siehe:
 - „INSERT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „UPDATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DELETE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „SELECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „UNION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „EXCEPT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „INTERSECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Programmierschnittstellen

Die folgende Liste enthält die Erweiterungen der in SQL Anywhere Version 11.0.0 eingeführten Programmierschnittstellen.

- **Neue SQL Anywhere C-API** Die SQL Anywhere C-Anwendungs-Programmierschnittstelle (API, Application Programming Interface) vereinfacht die Erstellung von C- und C++-Wrapper-Treibern für mehrere interpretierte Programmiersprachen, einschließlich PHP, Perl, Python und Ruby. Die SQL Anywhere C-API setzt auf der DBLIB-Bibliothek auf und wurde mit Embedded SQL realisiert.

Auch wenn sie kein Ersatz für DBLIB ist, vereinfacht die SQL Anywhere C-API die Erstellung von Anwendungen mit C und C++. Sie benötigen keine umfassenden Kenntnisse von Embedded SQL, um die SQL Anywhere C-API zu verwenden. Siehe „[SQL Anywhere-C-API-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **Neue Python-Datenbank-API (sqlanydb)** Die neue Python-Datenbank-API bietet Zugriff auf SQL Anywhere-Datenbanken über Skripten, die in Python geschrieben sind. Das sqlanydb-Modul implementiert die Python-Datenbank-API-Spezifikation Version 2.0 mit Erweiterungen. Siehe „[Python-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Externe Umgebungen** SQL Anywhere bietet nun Unterstützung für sechs externe Laufzeitumgebungen: Java, Perl, PHP, CLR, Embedded SQL und ODBC. SQL Anywhere besaß die Fähigkeit, kompilierte native Funktionen, die in C oder C++ geschrieben waren, aufzurufen. Wenn diese Prozeduren vom Server ausgeführt wurden, wurde die dynamische Verknüpfungsbibliothek (DLL) oder das Shared Object immer vom Datenbankserver geladen und die Aufrufe der nativen Funktion wurden immer vom Datenbankserver durchgeführt. Dadurch bestand das Risiko, dass der Datenbankserver abstürzte, wenn die native Funktion einen Fehler verursachte. Durch die Ausführung kompilierter nativer Funktionen außerhalb des Datenbankservers in einer externen Umgebung wird dieses Risiko nun vermieden. Siehe „[Unterstützung für externe Umgebungen in SQL Anywhere](#)“ [[SQL Anywhere Server - Programmierung](#)].

Zur Nutzung dieser neuen Funktion ist ein Datenbank-Upgrade erforderlich. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

- **Unterstützung der externen PHP-Umgebung** SQL Anywhere 11.0.0 verfügt über eine Reihe integrierter Binärdateien für verschiedene PHP-Versionen, einschließlich 5.1.1 bis 5.1.6 und 5.2.0 bis 5.2.6. Wenn eine dieser Versionen auf Ihrem Serversystem installiert ist, sollten Sie die integrierten SQL Anywhere-Binärdateien verwenden, anstatt die externe PHP-Umgebung zu erstellen. Beachten Sie, dass für Linux und Solaris sowohl 32-Bit- als auch 64-Bit-Versionen dieser Binärdateien bereitgestellt werden. Für Windows und andere Betriebssysteme werden nur 32-Bit-Versionen bereitgestellt.

Wenn Sie eine andere, oben nicht genannte PHP-Version verwenden, müssen Sie die Software selbst erstellen. Sie können Ihre PHP-Version auch durch eine andere Version ersetzen, die zu einer in SQL Anywhere vorkompilierten Version passt. Weitere Hinweise zum Erstellen des PHP-Moduls in SQL Anywhere finden Sie unter „[SQL Anywhere-PHP-Unterstützung](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **Unterstützung der externen Perl-Umgebung** Vor der Verwendung der externen Perl-Umgebung muss unbedingt die Version des Perl DBD-Treibers in SQL Anywhere aktualisiert werden. Ohne die Aktualisierung des Perl DBD-Treibers ist Perl serverseitig nicht funktionsfähig.

Im Unterschied zu PHP stellt SQL Anywhere keine integrierten Binärdateien für verschiedene Versionen von Perl bereitgestellt. Der Quellcode für den Perl DBD-Treiber von SQL Anywhere befindet sich unter `%SQLANY11%\SDK\perl`. Weitere Hinweise zum Erstellen des Perl DBD-Treibers von SQL Anywhere finden Sie unter „Perl/DBI-Unterstützung“ [[SQL Anywhere Server - Programmierung](#)].

- **Webserver-Unterstützung für UTF-8-URLs** Bislang übersetzte der Webserver %-kodierte Daten innerhalb des Anforderungs-URL (oder Daten im Format "application/x-www-form-urlencoded" im Hauptteil der Anforderung) in den Zeichensatz der Datenbank. Nun wird der Inhalt von %-kodierten Daten auf UTF-8-Sequenzen geprüft und basierend darauf soweit wie möglich in den Zeichensatz der Datenbank konvertiert. Alle kodierten Daten, die nicht im UTF-8-Format sind, werden dekodiert und so behandelt, als wären sie bereits im Zeichensatz der Datenbank.

Client-HTTPAnwendungen sollten ausschließlich %-kodierte UTF-8-Daten senden. Beachten Sie, dass ASCII-Zeichen in UTF-8 in ihrer ursprünglichen Form dargestellt werden. Ein Leerzeichen wird beispielsweise als %20 kodiert.

- **Neue Client-Callback-API** Zur Unterstützung der neuen Funktionen zum clientseitigen Laden und Entladen von Daten wurde eine neue Client-Callback-API hinzugefügt. Weitere Hinweise zu Embedded SQL finden Sie unter DB_CALLBACK_VALIDATE_FILE_TRANSFER in „db_register_a_callback-Funktion“ [[SQL Anywhere Server - Programmierung](#)]. Weitere Hinweise zu ODBC finden Sie unter SA_REGISTER_VALIDATE_FILE_TRANSFER_CALLBACK in „Erweiterte Verbindungsattribute für SQLSetConnectAttr“ [[SQL Anywhere Server - Programmierung](#)].

- **SQL_ATTR_CONNECTION_DEAD erkennt nun sofort ungültige Verbindungen** Bei der Verwendung des ODBC-Aufrufs SQLGetConnectAttr zum Abrufen des Attributs SQL_ATTR_CONNECTION_DEAD wird nun der Wert SQL_CD_TRUE zurückgegeben, wenn die Verbindung unterbrochen wurde, auch wenn seit der Unterbrechung der Verbindung keine Anforderung an den Server gesendet wurde. Es wird ermittelt, ob die Verbindung unterbrochen wurde, ohne eine Anforderung an den Server zu senden. Die Unterbrechung der Verbindung wird innerhalb weniger Sekunden erkannt. Die Verbindung kann aus mehreren Gründen unterbrochen werden, z.B. aufgrund eines Inaktivitäts-Timeouts. Vor dieser Änderung erhielt SQL_ATTR_CONNECTION_DEAD nur den Wert SQL_CD_TRUE, wenn die Verbindung getrennt wurde oder wenn der ODBC-Treiber eine Anforderung an den Server gesendet hat (z. B. durch den Aufruf von SQLExecDirect), nachdem die Verbindung unterbrochen wurde. Siehe „Verbindungsattribute abrufen“ [[SQL Anywhere Server - Programmierung](#)].

- **JDBC-Treiber unterstützt nun ResultSet.getBlob().getBinaryStream()** Der JDBC-Treiber von iAnywhere unterstützt derzeit die Methode ResultSet.getBlob(). Diese Methode ist in der JDBC-Spezifikation jedoch optional. Die optionale Methode ResultSet.getBlob().getBinaryStream() wird nun ebenfalls unterstützt. Siehe „JDBC 4.0-API-Unterstützung“ [[SQL Anywhere Server - Programmierung](#)].

- **Der iAnywhere JDBC-Treiber akzeptiert nun zusätzlich zu jdbc:odbc: auch jdbc:iAnywhere: als URL-Header** Wenn Anwendungen bislang den URL-Header jdbc:odbc verwendet haben,

konnte davon ausgegangen werden, dass der JDBC-Treibermanager für das Herstellen von Verbindungen mit diesem URL den iAnywhere JDBC-Treiber verwendete. In neueren Versionen der Java VM wird die Sun JDBC-ODBC Bridge als JDBC-Treiber registriert. Da die Sun JDBC-ODBC Bridge auch URLs akzeptiert, die mit jdbc:odbc beginnen, ist die Wahrscheinlichkeit, dass eine Anwendung die Sun JDBC-ODBC Bridge anstelle des iAnywhere JDBC-Treibers erhält, relativ groß. Um sicherzustellen, dass der JDBC-Treibermanager den iAnywhere-Treiber und nicht die Sun JDBC-ODBC Bridge verwendet, sollte die Anwendung stattdessen den URL-Header jdbc:ianywhere verwenden. Siehe „[Verbindungen aus einer JDBC-Clientanwendung](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **ODBC-Treibermanager akzeptiert nun driver=iAnywhere Solutions 11 - Oracle** Der Unix ODBC-Treibermanager akzeptiert nun driver=iAnywhere Solutions 11 - Oracle. Er lädt den iAnywhere ODBC-Treiber für Oracle mit Threading, wenn die Anwendung Threading verwendet. Er lädt den Treiber nicht, wenn die Anwendung kein Threading verwendet, da der iAnywhere ODBC-Treiber für Oracle ohne Threading nicht unterstützt wird. Siehe „[SQL Anywhere 16 - Oracle-ODBC-Treiber](#)“ [[MobiLink - Serveradministration](#)].
- **ODBC-Treibermanager akzeptiert nun driver=UltraLite 11** Der UNIX ODBC-Treibermanager akzeptiert bereits die Anweisung driver=SQL Anywhere 10 und lädt den SQL Anywhere ODBC-Treiber einwandfrei (entweder mit oder ohne Threading, abhängig von der Anwendung). Der Unix ODBC-Treibermanager akzeptiert nun driver=SQL Anywhere 11 und driver=UltraLite 11. Beim UltraLite-Treiber lädt der Treibermanager *nur* die Version des UltraLite-ODBC-Treibers mit Threading, da nur die Version mit Threading verfügbar ist.
- **Erweiterungen der TDS-Verbindungen** Der SQL Anywhere-Datenbankserver gestattet nun TDS-Verbindungen mit der Standarddatenbank, selbst wenn der Open Client-Login-Servername nicht mit dem Namen der Standarddatenbank übereinstimmt. Hierbei darf die Verbindungszeichenfolge keine Datenbank starten (d.h. DBF= . . . ist nicht vorhanden) und auf dem Datenbankserver darf nur eine Datenbank laufen.
- **Einfacheres erneutes Deployment der Startprogramme für die Administrationstools** Das erneute Deployment der Startprogramme für die Datenbanktools (Sybase Central, DBISQL, DBConsole, MobiLink-Monitor) wurde vereinfacht. Registrierungseinträge und eine feste Verzeichnisstruktur für den Speicherort der JAR-Dateien sind nicht mehr erforderlich. Für jede ausführbare Datei muss im gleichen Verzeichnis eine *.ini*-Datei vorhanden sein (mit demselben Namen wie die EXE-Datei), die Informationen darüber enthält, wie das Tool geladen wird. Siehe „[Deployment von Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **SQL Anywhere .NET-Datenprovider unterstützt nun die Eintragung verteilter Transaktionen** In .NET Framework 2.0 wurde der neue Namespace System.Transactions eingeführt, der Klassen zur Erstellung transaktionaler Anwendungen enthält. Clientanwendungen können verteilte Transaktionen mit einem oder mehreren Teilnehmern erstellen und an diesen beteiligt sein. Clientanwendungen können mit der Klasse TransactionScope implizit Transaktionen erstellen. Das Verbindungsobjekt kann feststellen, ob eine von TransactionScope erstellte Ambient-Transaktion vorhanden ist und diese automatisch eintragen. Clientanwendungen können auch mit CommittableTransaction eine festschreibbare Transaktion erstellen und die Methode EnlistTransaction aufrufen, um sie einzutragen.

Diese Funktion wird vom SQL Anywhere .NET-Datenprovider unterstützt. Verteilte Transaktionen haben einen erheblichen Performance-Overhead. Es wird empfohlen, für nicht verteilte Transaktionen Datenbanktransaktionen zu verwenden. Siehe „[Transaktionsverarbeitung](#)“ [*SQL Anywhere Server - Programmierung*].

- **SQL Anywhere NET.-Datenprovider unterstützt nun benannte Parameter** Der SQL Anywhere-Provider unterstützt nun benannte Parameter in SACommand. Wenn der Benutzer alle Parameternamen angibt, ordnet der Provider bei der Ausführung des Befehls Parameterwerte zu. Bei der Verwendung benannter Parameter muss die Reihenfolge der Parameter nicht mit der Reihenfolge der Hostvariablen übereinstimmen.

```
SACommand cmd = new SACommand(
    "UPDATE MyTable SET name = :name WHERE id = :id", conn );

SAParameter p1 = new SAParameter(
    "id", SADBType.Integer );
p1.Direction = ParameterDirection.Input;
p1.Value = 1;
cmd.Parameters.Add( p1 );

SAParameter p2 = new SAParameter(
    "name", SADBType.Char, 40 );
p2.Direction = ParameterDirection.Input;
p2.Value = "asdasd";
cmd.Parameters.Add( p2 );

cmd.ExecuteNonQuery();
```

- **Erweiterungen der Webdienste** Den Webdiensten wurden in dieser Version folgende Erweiterungen hinzugefügt:
 - **Erweiterung der Webclient-Dienstprozeduren vom Typ HTTP:POST für einen benutzerdefinierten Hauptteil** Die TYPE-Klausel der Anweisungen CREATE PROCEDURE und CREATE FUNCTION wurde so erweitert, dass sie nun die Spezifikation eines Mime-Typs erlaubt. Siehe „[CREATE FUNCTION-Anweisung \[Webdienst\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*] oder „[CREATE PROCEDURE-Anweisung \[Webdienste\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
 - **Erweiterung der Webdienst-Clientprozeduren zur Unterstützung der Methoden PUT, DELETE und HEAD HTTP** Webdienst-Clientprozeduren und -funktionen unterstützen nun die Methoden PUT, DELETE und HEAD HTTP. Die TYPE-Klausel der Anweisungen CREATE PROCEDURE und CREATE FUNCTION wurde erweitert und unterstützt nun diese Methoden. Ähnlich der POST-Methode erfordert PUT eine Content-Type-Erweiterung innerhalb der TYPE-Klausel. Nur ein nicht ersetzbarer Parameter ist zulässig. Siehe „[CREATE SERVICE-Anweisung \[HTTP-Webdienst\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*], „[CREATE SERVICE-Anweisung \[SOAP-Webdienst\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*], „[CREATE FUNCTION-Anweisung \[Webdienst\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*] und „[CREATE PROCEDURE-Anweisung \[Webdienste\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
 - **Systemprozeduren sa_http_php_page und sa_http_php_page_interpreted** Die neuen Webdienst-Systemprozeduren sa_http_php_page und sa_http_php_page_interpreted geben das Ergebnis eines Durchlaufs eines PHP-Skripts durch einen PHP-Interpreter zurück. Siehe

„sa_http_php_page-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „sa_http_php_page_interpreted-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].

- **HTTP_BODY-Systemfunktion** Die neue Webdienst-Funktion HTTP_BODY wurde hinzugefügt. Die Funktion HTTP_BODY gibt den Hauptteil der HTTP-Anforderung in Binärform zurück. Siehe „[HTTP_BODY-Funktion \[Webdienst\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **WSDLC-Unterstützung zum Erzeugen von SOAP-Prozeduren des Webdienstclients** WSDLC unterstützt nun auch die Generierung von SQL SOAP-Clientprozeduren (Webdienst) für SQL Anywhere. WSDLC liest einen WSDL1.1-kompatiblen URL oder eine Datei und erzeugt Prozeduren (oder Funktionen) mit geeigneten Parametern und Klauseln, die in der WSDL aufgelisteten SOAP-Vorgängen zugeordnet sind. Die erzeugten SQL-Anweisungen werden in eine SQL-Datei geschrieben.
- **HTTP SOAP-Dienste, die mit einer FORMAT 'CONCRETE'-Klausel definiert wurden, können mit EXPLICIT OFF bzw. ON genauer qualifiziert werden.** Bei der Erstellung eines HTTP SOAP-Dienstes ist der Standardwert der FORMAT-Klausel EXPLICIT ON. Das bedeutet, dass die von einem DISH-Dienst generierte WSDL für jede in einer Ergebnismenge zurückgegebene Spalte explizite Namen und Datentypen angibt. Somit können SOAP-Client-Toolkits automatisch clientseitige Objekte und Schnittstellen generieren, die die Ergebnismenge repräsentieren und den Zugriff des Systems auf die Spaltenwerte ermöglichen. Bevor diese Funktion verfügbar war, konnte auf Spaltenwerte nur in Form von abstrakten XML-Datenelementen zugegriffen werden. Dieses Verhalten ist immer noch verfügbar, wenn EXPLICIT OFF angegeben wird.

Weitere Hinweise zum Definieren eines EXPLICIT-Antwortobjekts oder des generischen SimpleDataset finden Sie unter „[CREATE SERVICE-Anweisung \[SOAP-Webdienst\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)] und „[Praktische Einführung: Verwenden von JAX-WS für den Zugriff auf einen SOAP/DISH-Webdienst](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **Unterstützung der JSON-Webdienste** SQL Anywhere unterstützt nun Webdienste, die JSON-formatierte Antworten zurückgeben. Siehe „[CREATE SERVICE-Anweisung \[HTTP-Webdienst\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Protokollierung von Webdienst-Clients** Der Datenbankserver unterstützt nun die Protokollierung von Webdienst-Clientverbindungen in einer Ausgabedatei. Sie können die Serveroption -zoc angeben oder die Eigenschaften WebClientLogFile und WebClientLogging zusammen mit der Systemprozedur sa_server_option verwenden, um die Protokollierung zu steuern und den Speicherort der Webdienst-Client-Logdatei festzulegen. Sie können die Verwendung dieser Funktion mit der Serveroption -sf auch deaktivieren. Siehe:
 - „[Datenbankserveroption -zoc](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „[Datenbankserveroption -sf](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „[sa_server_option-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Erweiterungen für Windows Mobile

Die folgende Liste enthält alle Windows Mobile-Erweiterungen in SQL Anywhere Version 11.0.0.

- **Unterstützte Serveroption -gss** Unter Windows CE 4 (Pocket PC 2003) und höher können Sie mit der Serveroption -gss die Standard-Stack-Größe für interne Ausführungs-Threads festlegen. Siehe „Datenbankserveroption -gss“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Prüfsummen standardmäßig aktiviert** Wenn eine Datenbank unter Windows Mobile ausgeführt wird, aktiviert der Datenbankserver automatisch Prüfsummen, unabhängig davon, ob die Prüfsummen für die Datenbank aktiviert waren. Zur Verwendung dieser Funktion müssen Sie ein Upgrade der vorhandenen Datenbanken vornehmen oder eine neue Datenbank erstellen. Siehe „Erkennung von Beschädigungen mithilfe von Prüfsummen“ [[SQL Anywhere Server - Datenbankadministration](#)].

Unix- bzw. Linux-Erweiterungen

Die folgende Liste enthält die Erweiterungen für Unix und Linux in SQL Anywhere Version 11.0.0.

- **Berechtigungen für temporäre Dateien steuern** In früheren Versionen galten für die vom Datenbankserver und -client erstellten temporären Dateien globale Lese-, Schreib- und Ausführungsberechtigungen. Es ist nicht möglich, die Berechtigungen für temporäre Dateien zu steuern, indem die SATMP-Umgebungsvariable auf ein Verzeichnis mit den gewünschten Berechtigungen festgelegt wird. Siehe „SATMP-Umgebungsvariable“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung für SELinux** SELinux-Richtlinien steuern den Zugriff einer Anwendung auf Systemressourcen. Sie können die Standardrichtlinie von SQL Anywhere auf Red Hat Enterprise Linux 5 verwenden, doch SQL Anywhere ist nicht sicher, wenn es auf diese Weise ausgeführt wird. SQL Anywhere verfügt nun über eine Richtlinie, die die Anwendung unter Red Hat Enterprise Linux 5 sichert. Sie müssen die Richtlinie kompilieren und installieren, um sie zu verwenden. Der Richtlinien Quellcode wird als Teil der SQL Anywhere-Installation bereitgestellt.

Weitere Informationen zum Kompilieren und Installieren der SQL Anywhere SELinux-Richtlinie finden Sie in `$SQLANY11/selinux/readme`.

- **Elemente des Anwendungsmenüs unter Linux** Bei der Installation von SQL Anywhere 11 unter Linux können Sie auswählen, Optionen im **Anwendungsmenü** unter Linux zu erstellen.

Mac OS X

- **Verschlüsselung unter Mac OS X nun unterstützt** Die RSA-Kommunikationsverschlüsselung wird nun unter Mac OS X sowohl vom Datenbankserver als auch von den Clients unterstützt. Weitere Hinweise zur Verwendung der starken Verschlüsselung finden Sie unter „Transportschichtssicherheit“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **HTTPS nun unter Mac OS X unterstützt** Die HTTPS-Kommunikation wird nun unter Mac OS X vom Datenbankserver unterstützt. Weitere Hinweise zur Verwendung von HTTPS finden Sie unter „Datenbankserveroption -xs“ [[SQL Anywhere Server - Datenbankadministration](#)].

Verschiedenes

Die folgende Liste enthält verschiedene Erweiterungen in SQL Anywhere Version 11.0.0.

- **Ungültige Ansichten ändern** Bislang konnte eine normale Ansicht mit dem Status INVALID (ungültig) nicht geändert werden und Sie mussten die Ansicht löschen und neu erstellen. Nun können Sie eine ungültige Ansicht ändern und damit automatisch die Definition ändern, sodass die Ansicht nicht mehr ungültig ist.
- **Neues Synonym für die LENGTH-Funktion** Sie können nun LEN als Synonym für LENGTH verwenden, wenn Sie die LENGTH-Funktion ausführen. Siehe „[LENGTH-Funktion \[Zeichenfolge\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Unterstützung für Big-Endian und Little-Endian UTF-16-Kodierung** SQL Anywhere unterstützt nun sowohl Big-Endian als auch Little-Endian UTF-16-Kodierung auf allen Plattformen, und zwar unabhängig von der Endianness der Plattform. Sie können die UTF-16-Kodierung in den Anweisungen LOAD TABLE und INLOAD sowie mit der Funktion CSCONVERT verwenden. Es ist dagegen nicht möglich, die UTF-16-Kodierung als Kodierung für eine Verbindung oder Datenbank zu verwenden. Siehe „[LOAD TABLE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*] und „[UNLOAD-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].

Zur Nutzung dieser Funktion ist ein Datenbank-Upgrade erforderlich. Siehe „[SQL Anywhere-Server-Upgrades](#)“ auf Seite 295.

- **Verbesserte Index-Performance** Die Index-Performance wurde gesteigert, vor allem bei der Arbeit mit einem vollen Cache. Um die Vorteile durch die verbesserte Index-Performance zu nutzen, müssen Sie Ihre Indizes neu erstellen. Die einfachste Möglichkeit hierzu ist, die Datenbank neu zu erstellen. Nach der Neuerstellung ist die Datenbankdatei möglicherweise wesentlich kleiner. Dies ist normal und kein Grund zur Beunruhigung.
- **Einstellungen INLINE und PREFIX nun für komprimierte Spalten berücksichtigt** Bislang wurden die Einstellungen von INLINE und PREFIX für eine Spalte ignoriert und als 0 angenommen, wenn die Spalte komprimiert war. Nun werden die Einstellungen für Spalten auch dann berücksichtigt, wenn die Spalte komprimiert ist. Siehe „[Hinweise zu BLOBs](#)“ [*SQL Anywhere Server - Datenbankadministration*] und „[CREATE TABLE-Anweisung](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Hostvariablen nun in Batches erlaubt** Referenzen auf Hostvariablen sind nun - mit einigen Einschränkungen - in Batches zulässig. Siehe „[Batchanweisungen](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
- **Erweiterungen des InList-Algorithmus** Bislang wurde der InList-Algorithmus vom Optimierer nur verwendet, wenn alle Elemente der IN-Liste entweder konstante Werte waren oder zum Zeitpunkt der Optimierung als konstante Werte ausgewertet werden konnten. Nun kann das IN-Listenprädikat Werte enthalten, die erst zum Zeitpunkt der Öffnung ausgewertet werden (CURRENT DATE, CURRENT TIMESTAMP oder nicht-deterministische System- und benutzerdefinierte Funktionen), ebenso wie Werte, die innerhalb einer Ausführung eines Abfrageblocks (äußere Referenzen) konstant sind.

- **Plan-Caching für einfache DML-Anweisungen** Das Plan-Caching wurde auf SELECT-Anweisungen erweitert, die für das Umgehen der Abfrage in Frage kommen (einfache Anweisungen). Siehe „Plan-Caching“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Größe neuer Datenbanken verringert** Die folgenden Systemtabellenspalten sind nun komprimiert, um die Größe neuer (leerer) Datenbanken um etwa 200 kB zu verringern. Dies ist bei der Erstellung von Datenbanken für die Verwendung unter Windows Mobile nützlich.
 - ISYSEVENT.action
 - ISYSJARCOMPONENT.contents
 - ISYSPROCEDURE.proc_defn
 - ISYSSOURCE.source
 - ISYTEXTCONFIG.char_stoplist
 - ISYTEXTCONFIG.nchar_stoplist
 - ISYSTRIGGER.trigger_defn
 - ISYSVIEW.view_def
- **Erhöhte Standard- und Mindestgröße für Pakete** Die Standardpaketgröße wurde auf allen Betriebssystemen ausgenommen Windows Mobile auf 7300 Byte erhöht. Auf Windows Mobile beträgt die Standardgröße weiterhin 1460 Byte. Die Mindestpaketgröße wurde auf 500 Byte erhöht. Siehe „Verbindungsparameter CommBufferSize (CBSIZE)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Datenbankserveroption -p“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue ODBC-Klassen für Ferndatenzugriff unterstützt** Unterstützung für folgende ODBC-Klassen wurde hinzugefügt:
 - MSACCESSODBC
 - MYSQLODBC
 - ULODBC
 - ADSODBC

Weitere Hinweise finden Sie unter „Serverklassen für den Ferndatenzugriff“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Hinweis

Während Sie bislang das SQL Anywhere-Dienstprogramm für die Migration von MS Access (Upsize-Tool) verwendet haben, um Microsoft Access-Datenbanken nach SQL Anywhere zu migrieren, ist hierfür nun die MSACCESSODBC-Klasse verfügbar.

- **Erweiterungen von Datenbankservernachrichten** Den Nachrichten vom Datenbankserver ist nun eine Kategorie und ein Schweregrad zugewiesen. Sie können auf diese Informationen mithilfe der Systemprozedur `sa_server_messages` zugreifen und die Anzahl der verwalteten Nachrichten mit der Eigenschaft `MessageCategoryLimit` konfigurieren. Siehe „`sa_server_messages`-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Neuer VALIDATE_COMPLETE-Parameter für a_validate_type-Enumeration** Die `a_validate_type`-Enumeration hat den neuen Parameter `VALIDATE_COMPLETE` zur Durchführung

aller möglicher Validierungen der Datenbank. Siehe [a_validate_db-Struktur \[Datenbanktools\]](#) [*SQL Anywhere Server - Programmierung*].

- **Erweiterung des externen Entladevorgangs** Wenn Sie einen externen Entladevorgang einer Datenbank ausführen, enthält der Anfang der generierten Datei *reload.sql* nun eine kommentierte CREATE DATABASE-Anweisung. Diese Anweisung kann verwendet werden, um eine Datenbank zu erstellen, die mit der entladenen Datenbank übereinstimmt.

Wenn die entladene Datenbank mit Version 9 oder einer früheren Version von SQL Anywhere erstellt wurde und eine benutzerdefinierte Kollation hatte, wird die COLLATION-Klausel wie folgt angezeigt:

```
COLLATION collation-label DEFINITION collation-definition
```

Dabei gilt: *collation-definition* ist eine Zeichenfolge, die die benutzerdefinierte Kollation angibt.

Wenn die entladene Datenbank mit starker Verschlüsselung erstellt wurde, wird der Wert des KEY-Schlüssels in der CREATE DATABASE-Anweisung in Form von drei Fragezeichen angezeigt (???).

Weitere Hinweise finden Sie unter [Vergleich von externem Entladen und Neuladen \[SQL Anywhere Server - Datenbankadministration\]](#).

- **Neue OIDs des SQL Anywhere-Erweiterungsagenten** Folgende OIDs wurden in dieser Version hinzugefügt:
 - saAgent.saRestart
 - saAgent.saInifile

Weitere Hinweise finden Sie unter „SQL Anywhere MIB-Referenzhandbuch“ [*SQL Anywhere Server - Datenbankadministration*].

- **Deadlock-Systemereignis** Das Deadlock-Systemereignis wird ausgelöst, wenn ein Deadlock auftritt. Der Event-Handler kann mit der Prozedur sa_report_deadlock Informationen über die Bedingungen erhalten, die zum Deadlock geführt haben. Zur Verwendung des Deadlock-Systemereignisses müssen Sie ein Upgrade der vorhandenen Datenbanken durchführen. Siehe „Systemereignisse“ [*SQL Anywhere Server - Datenbankadministration*].
- **Höhere Datenbankbegrenzungen** Einige SQL Anywhere-Datenbankgrenzen wurden erhöht. Siehe „SQL Anywhere-Einschränkungen von Größe und Anzahl“ [*SQL Anywhere Server - Datenbankadministration*].
- **Änderungen von Ausführungsplänen** Vom Optimierer generierte lange Pläne zeigen nun die folgenden Einträge an, die sich auf den Gesamtplan beziehen:
 - **Beste Pläne mit Kosten** Anzahl unterschiedlicher bester Zugriffspläne, die vom Optimierer gefunden wurden.
 - **Pläne mit Kosten** Zahl der verschiedenen vom Optimierer berücksichtigten Zugriffspläne.
 - **Optimierungszeit** Die für die Optimierung der Abfrage aufgewendete Zeit.

Siehe „Komponenten von Ausführungsplänen“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Grafische Pläne zeigen nun folgende Einträge an:

- **Beste Pläne mit Kosten** Dieser Eintrag befindet sich im Abschnitt **Statistiken des Optimierers** des Stammknotens. Er gibt die Anzahl der verschiedenen vom Optimierer gefundenen besten Zugriffspläne an.
- **Pläne mit Kosten** Dieser Eintrag befindet sich im Abschnitt **Statistiken des Optimierers** des Stammknotens. Er gibt die Anzahl der verschiedenen vom Optimierer berücksichtigten Zugriffspläne an.
- **Optimierungszeit** Dieser Eintrag befindet sich im Abschnitt **Statistiken des Optimierers** des Stammknotens. Er gibt an, wie viel Zeit für die Abfrage aufgewendet wurde.
- **FirstRowRunTime** Dieser Eintrag befindet sich im Abschnitt **Knotenstatistiken**. Er gibt an, wie viel Zeit zum Abruf der ersten Zeile aufgewendet wurde.
- **Berücksichtigte Joins** Dieser Eintrag befindet sich im Abschnitt **Erweiterte Details** jedes Join-Operators. Er listet alle Operatoren auf, die vom Optimierer während des Optimierungsprozesses für die Unterstruktur rechts vom Join-Operator berücksichtigt wurden.
- **Vorfilter-Prädikate** Dieser Eintrag befindet sich in einem neuen Scan-Knoten-Abschnitt im Fensterausschnitt **Details**. Er listet alle Prädikate auf, die ausgewertet werden, bevor der Scan gestartet wird.
- **Scan-Prädikate** Dieser Eintrag befindet sich in einem Scan-Knoten-Abschnitt im Fensterausschnitt **Details**. Er listet die Prädikate auf, die beim Abrufen von Spalten aus der Zeile ausgewertet werden. Wenn ein Scan-Prädikat eine Zeile ablehnt, werden keine weiteren Spalten gelesen. Scan-Prädikate sind einfache, einzelne Spaltenprädikate, z. B. $T.x \leq 3$ oder $T.x \text{ IS NULL}$.
- **Nachscan-Prädikate** Dieser Eintrag befindet sich in einem neuen Scan-Knoten-Abschnitt im Fensterausschnitt **Details**. Er listet die Prädikate auf, die unmittelbar nach dem Lesen einer Zeile der Tabellenseite ausgewertet werden. Nachscan-Prädikate können sich auf mehrere Spalten beziehen und Funktionen oder arithmetische Vorgänge verwenden.
- **Verbleibende Prädikate** Dieser Eintrag befindet sich in einem neuen Scan-Knoten-Abschnitt im Fensterausschnitt **Details**. Er listet Prädikate auf, die nach dem Abruf einer Gruppe von Zeilen in den Speicher ausgewertet werden. Verbleibende Prädikate enthalten normalerweise komplexe Vorgänge, z.B. Unterabfragen oder benutzerdefinierte Funktionen, und können nicht als Scan-Prädikate oder Nachscan-Prädikate ausgewertet werden.
- **Berücksichtigte Indizes** Dieser Eintrag befindet sich im Fensterausschnitt **Erweiterte Details**. Dieser Eintrag listet alle Index- oder Tabellen-Scans auf, die vom Optimierer während des Optimierungsprozesses für die von diesem Scan-Operator referenzierte Tabelle berücksichtigt werden. Das Format der einzelnen Einträge in der Liste ist ähnlich dem der Details, die für einen im Zugriffsplan verwendeten Scan-Operator im Fensterausschnitt **Details** aufgelistet werden.

- **Primärschlüsseltabelle** Dieser Eintrag befindet sich im Abschnitt **Index** eines Indexscan-Operators. Er gibt den Namen der Primärschlüsseltabelle an.
- **Geschätzte Zeilen in Primärschlüsseltabelle** Dieser Eintrag befindet sich im Abschnitt **Index** eines Indexscan-Operators. Er gibt die Anzahl der Zeilen in der Primärschlüsseltabelle an.
- **Primärschlüsselspalte** Dieser Eintrag befindet sich im Abschnitt **Index** eines Indexscan-Operators. Er gibt die Namen der Primärschlüsselspalten an.
- **Sequenzielle Transitionen** Dieser Eintrag befindet sich im Abschnitt **Index** eines Indexscan-Operators. Er enthält die Statistiken für alle physischen Indizes und gibt an, wie stark geclustert der Index ist.
- **Zufällige Transitionen** Dieser Eintrag befindet sich im Abschnitt **Index** eines Indexscan-Operators. Er enthält die Statistiken für alle physischen Indizes und gibt an, wie stark geclustert der Index ist.
- **Schlüsselwerte** Dieser Eintrag befindet sich im Abschnitt **Index** eines Indexscan-Operators. Er gibt die Anzahl der eindeutigen Einträge im Index an.

Verhaltensänderungen von SQL Anywhere

Im Folgenden finden Sie eine nach Kategorien sortierte Liste der Änderungen von SQL Anywhere in Version 11.0.0.

- **Katalogänderungen** Folgende Tabelle enthält die Katalogänderungen in Version 11.0.0.

Um diese Änderungen zu erhalten, müssen Sie ein Upgrade Ihrer Datenbank durchführen. Siehe „[SQL Anywhere-Server-Ugrades](#)“ auf Seite 295.

Tabellenname bzw. Ansichtsname	Beschreibung der Änderung
ISYSTAB/SYSTAB	<ul style="list-style-type: none">○ Die neue Spalte dbspace_id wurde hinzugefügt. Sie soll die derzeitige Spalte file_id ersetzen.○ Die Spalte file_id wird nicht mehr empfohlen. Verwenden Sie stattdessen die Spalte dbspace_id. Für globale temporäre Tabellen zeigt SYSTAB.file_id nun auf den temporären DBSpace anstatt auf den DBSpace SYSTEM.○ Die neue Spalte last_modified_tsn speichert eine sequenzielle Zahl für die Transaktion, die die Tabelle geändert hat.

Tabellenname bzw. Ansichtsname	Beschreibung der Änderung
ISYSIDX/SYSIDX	<ul style="list-style-type: none"> Die neue Spalte dbspace_id wurde hinzugefügt. Sie soll die derzeitige Spalte file_id ersetzen. Die Spalte file_id wird nicht mehr empfohlen. Verwenden Sie stattdessen die Spalte dbspace_id.
ISYSFILE	Diese Systemtabelle wird nicht mehr empfohlen. Alle Spalten, ausgenommen lob_map, befinden sich nun in der (neuen) Systemtabelle ISYSDBSPACE. Die Spalte lob_map befindet sich nun in der (neuen) Systemtabelle ISYSDBFILE.
ISYSDBFILE/SYSDBFILE	Neue Tabelle für Informationen über DBSpaces.
ISYSDBSPACE/SYSDBSPACE	Neue Tabelle für Informationen über DBSpaces.
SYSDBSpacePERM/ISYSDBSPACEPERM	Neue Tabelle für DBSpace-Berechtigungen.
ISYSOBJECT/SYSOBJECT	Die Spalte file_id wurde in dbspace_id umbenannt. Die Spalte object_type kann zwei neue Werte enthalten: 17 (Textkonfiguration) und 18 (DBSpace).
SYSINDEXES	Das Feld indextype identifiziert nun Fremdschlüssel- und Primärschlüsselindizes als Primärschlüssel bzw. Fremdschlüssel, um sie von anderen Indizes zu unterscheiden.
ISYSCAPABILITYNAME	Diese Tabelle ist im Katalog nicht mehr vorhanden. Die zugehörige Systemansicht SYSCAPABILITYNAME ist weiterhin verfügbar, doch sie wird mithilfe von Servereigenschaften generiert.
ISYSEVENTTYPE	Diese Tabelle ist im Katalog nicht mehr vorhanden. Die zugehörige Systemansicht SYSEVENTTYPE ist weiterhin verfügbar, doch sie wird mithilfe von Servereigenschaften generiert.
ISYSVIEW	Die neue Spalte mv_last_refreshed_tsn speichert eine sequenzielle Zahl für die Transaktion, die die materialisierte Ansicht aktualisiert hat.
ISYSLOGINMAP/SYSLOGINMAP	Diese neue Tabelle speichert Informationen über Login-Richtlinien.
ISYSLOGINPOLICY/SYSLOGINPOLICY	Diese neue Tabelle speichert Informationen über Login-Richtlinien.

Tabellenname bzw. Ansichtsname	Beschreibung der Änderung
ISYSLOGINPOLICY- OPTION/SYSLOGIN- POLICYOPTION	Diese neue Tabelle speichert Informationen über Login-Richtlinien.
ISYTEXTCONFIG/ SYSTEXTCONFIG	Diese neue Tabelle speichert Informationen über Textkonfigurationsobjekte.
ISYTEXTIDX/ SYSTEXTIDX	Diese neue Tabelle speichert Informationen über Textindizes.
ISYTEXTIDXTAB/ SYSTEXTIDXTAB	Diese neue Tabelle speichert Informationen über Textindizes.

- **Änderungen der PHP-Funktionsnamen** Alle PHP-Funktionen wurden umbenannt und haben nun das Präfix sasql_ anstelle von sqlanywhere_. Das Präfix sqlanywhere_ kann beim Aufruf einer Funktion weiterhin verwendet werden, doch es wird nicht mehr empfohlen. Sie sollten Ihre Anwendung so ändern, dass sie das neue Präfix verwendet.
- **Anweisung INSERT...ON EXISTING UPDATE löst nun Trigger aus** Wenn Sie bislang eine INSERT...ON EXISTING UPDATE-Anweisung ausgeführt haben, wurden beim Aktualisieren von Daten keine Trigger ausgelöst. Nun löst der Datenbankserver für die Aktualisierungen Trigger auf Anweisungsebene aus.
- **REFRESH MATERIALIZED VIEW-Anweisung** Es ist nicht mehr möglich, die Anweisungen STATEMENT SNAPSHOT und READONLY STATEMENT SNAPSHOT als Isolationsstufe für die Aktualisierung zu verwenden, da die Wirkung dieser Optionen der Angabe von SNAPSHOT für die Isolationsstufe entspricht. Siehe „REFRESH MATERIALIZED VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **REORGANIZE TABLE-Anweisung** Der Versuch, gleichzeitig mehrere REORGANIZE TABLE-Anweisungen für eine Tabelle auszuführen, führt nun zu einem Fehler.
- **sa_validate-Systemprozedur** Die Argumente check_type, express und checksum für sa_validate werden nicht mehr empfohlen. Ihre Verwendung ist nun ohne Wirkung. Die Prüfsummenvalidierung wird nun standardmäßig durchgeführt. Wenn die Systemprozedur sa_validate ohne Argumente aufgerufen wird, überprüft der Datenbankserver neben allen Tabellen, materialisierten Ansichten und Indizes auch die Datenbank selbst, einschließlich der Prüfsummen. Siehe „sa_validate-Systemprozedur“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **Serveroption -gss** Die Serveroption -gss wird nun von Windows XP und höher unterstützt. In früheren Versionen wurde diese Option von Windows-Betriebssystemen nicht unterstützt. Siehe „Datenbankserveroption -gss“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Serveroption -gx nicht mehr unterstützt** Die Unterstützung der Serveroption -gx wurde in dieser Version entfernt. Die Verwendung der Option -gx beim Start eines SQL Anywhere-Datenbankservers führt zu einem Fehler.

- Standardeinstellung des LazyClose-Verbindungsparameters ist nun AUTO** In früheren Versionen war ein Roundtrip zum Datenbankserver erforderlich, wenn eine Anwendung einen Cursor geschlossen hat, außer der LazyClose-Verbindungsparameter war auf NO gesetzt. Nun werden Schließenanforderungen vieler Cursor in eine Warteschlange gestellt, wodurch ein Roundtrip vermieden und die Performance verbessert wird. Der Verbindungsparameter LazyClose akzeptiert nun drei Werte: YES, NO und AUTO (Standardwert). YES war in früheren Versionen die Standardeinstellung. Siehe „[Verbindungsparameter LazyClose \(LCLOSE\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - Änderungen bei der Embedded SQL-Importbibliothek** Die Watcom- und Borland-Versionen der DBLIB-Importbibliotheken sind nicht mehr enthalten. Hierbei handelt es sich um *dblibtw.lib* bzw. *dblibtb.lib*. Als Ersatz für diese Importbibliotheken wird eine Importdefinitionsdatei (Datei `%SQLANY11%\SDK\Lib\Def\dblib.def`) bereitgestellt.
 - Änderungen bei Datenbanktools-Importbibliotheken** Die Watcom- und Borland-Versionen der Datenbanktools-Importbibliotheken sind nicht mehr enthalten. Hierbei handelte es sich um *dbtlstw.lib* bzw. *dbtlstb.lib*. Als Ersatz für diese Importbibliotheken wird eine Importdefinitionsdatei (`%SQLANY11%\SDK\Lib\Def\dbtool.def`) bereitgestellt.
 - Verhalten des DBLIB-Indikators definiert für den Fall, dass keine Zeilen empfangen werden** Wenn eine Fetch- oder Execute-Anweisung keine Zeilen vom Datenbankserver erhält (bei einem Fehler oder wenn das Ende der Ergebnismenge erreicht ist), bleiben die Indikatorwerte nun unverändert. Siehe „[Indikatorvariablen](#)“ [[SQL Anywhere Server - Programmierung](#)].
 - ODBC SQLGetConnectAttr** Bei der Verwendung des ODBC-Aufrufs SQLGetConnectAttr zum Abrufen des Attributs SQL_ATTR_CONNECTION_DEAD wird nun der Wert SQL_CD_TRUE zurückgegeben, wenn die Verbindung unterbrochen wurde, auch wenn seit der Unterbrechung der Verbindung keine Anforderungen an den Server gesendet wurden. Es wird ermittelt, ob die Verbindung unterbrochen wurde, ohne eine Anforderung an den Server zu senden. Die Unterbrechung der Verbindung wird innerhalb weniger Sekunden erkannt. Die Verbindung kann aus mehreren Gründen unterbrochen werden, z.B. aufgrund eines Inaktivitäts-Timeouts.
- In früheren Versionen erhielt SQL_ATTR_CONNECTION_DEAD nur den Wert SQL_CD_TRUE, wenn die Verbindung getrennt wurde oder wenn der ODBC-Treiber eine Anforderung an den Server gesendet hat (z. B. durch den Aufruf von SQLExecDirect), nachdem die Verbindung unterbrochen wurde.
- Datenbanken mit dem Namen utility_db können nicht erstellt oder gestartet werden** Der Name utility_db ist nun für die SQL Anywhere Server-Dienstprogrammdatenbank reserviert. Wenn Sie versuchen, eine neue Datenbank mit dem Namen *utility_db.db* zu erstellen oder eine vorhandene gleichnamige Datei zu starten, wird ein Fehler zurückgegeben. Wenn Sie eine Datenbank mit dem Namen utility_db haben, können Sie diese mit einem anderen Namen starten. Siehe „[Die Dienstprogrammdatenbank](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - Berechnete Spaltenabhängigkeiten** Bislang konnte eine Anwendung einen Aktualisierungs- oder Einfügevorgang ohne Fehler fortführen, indem Trigger verwendet wurden, um Spalten, die als NOT NULL deklariert waren, Nicht-Nullwerte zuzuweisen. Dies hatte Auswirkungen auf berechnete Spalten, die von der Spalte abhängig waren, da sich ein berechneter Wert ergeben konnte, der nicht der beabsichtigten Berechnung entsprach. Nun schlägt der Versuch fehl, einer NOT NULL-Spalte, die

von einer berechneten Spalte abhängt, einen Nullwert zuzuweisen. Es wird eine Fehlermeldung ausgegeben, und es werden keine Trigger ausgelöst. Siehe „[Einfügungen in und Aktualisierungen von berechneten Spalten](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

- **DBSpace-Namen mit einem Punkt generieren einen Fehler** Wenn in früheren Versionen ein DBSpace-Name, der nicht in Anführungszeichen stand, einen Punkt enthielt, wurde der Teil des DBSpace-Namens vor dem Punkt vom Server stillschweigend ignoriert. Nun generiert der Datenbankserver einen Fehler für diese Namen.
- **SQL Anywhere-Webserver unterstützt SSL-Version 2.0 nicht mehr** Der SQL Anywhere-Webserver unterstützt ab dieser Version nur Verbindungen mit SSL-Version 3.0 und TLS-Version 1.0. Verbindungen der SSL-Version 2.0 werden nicht unterstützt.
- **DATATYPE-Standardwert für CREATE SERVICE-Option wurde geändert** Der Standardwert der DATATYPE-Klausel wurde von OFF auf ON geändert. Wenn Sie das alte Verhalten beibehalten wollen, müssen Sie DATATYPE OFF explizit in die CREATE SERVICE-Definition einbeziehen. Siehe „[CREATE SERVICE-Anweisung \[SOAP-Webdienst\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Prüfsummen-Verhaltensänderungen** Bei Datenbanken, die mit Version 11 erstellt wurden oder für die ein Upgrade auf Version 11 durchgeführt wurde, aktiviert der Datenbankserver automatisch die Prüfsummen für Datenbanken, die auf anderen Speichermedien ausgeführt werden, wie z. B. auf Netzlaufwerken oder Wechseldatenträgern. Prüfsummen bleiben aktiviert, so lange die Datenbank sich auf einem solchen Gerät befindet. Siehe „[Erkennung von Beschädigungen mithilfe von Prüfsummen](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- **HTTP-Verbindungen bewirken keinen automatischen Stopp von Datenbanken** In früheren Versionen wurde eine für den automatischen Stopp konfigurierte Datenbank gestoppt, wenn eine HTTP-Verbindung getrennt wurde und keine anderen Verbindungen mit der Datenbank vorhanden waren. Nun werden Datenbanken nur dann automatisch gestoppt, wenn die letzte Befehlssequenz eine Trennung ausführt oder die TDS-Verbindung getrennt wird.

Wenn die einzige Verbindung mit einer Datenbank eine HTTP-Verbindung ist und die Datenbank so konfiguriert wurde, dass sie automatisch gestoppt wird, wird bei der Trennung der HTTP-Verbindung die Datenbank nicht automatisch gestoppt. Wenn eine für den automatischen Stopp konfigurierte Datenbank eine HTTP-Verbindung und eine Befehlssequenz oder eine TDS-Verbindung hat und die letzte Befehlssequenz eine Trennung bewirkt oder die TDS-Verbindung getrennt wird, dann wird die Datenbank gestoppt und alle HTTP-Verbindungen werden getrennt. Siehe „[Datenbankserveroption - ga](#)“ [*SQL Anywhere Server - Datenbankadministration*] und „[Verbindungsparameter AutoStop \(ASTOP\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- **Verhaltensänderung der Datenbankspiegelung** Wenn in früheren Versionen die in der Option -xp angegebenen Verbindungsparameter für den Primär- oder Spiegelserver ungültig waren, hat der Datenbankserver wiederholt versucht, eine Verbindung herzustellen, doch die Verbindungserstellung konnte nie hergestellt werden. Wenn in dieser Version Verbindungsparameter in der Option -xp ungültig sind und mehrere Datenbanken auf dem Server ausgeführt werden, kann die gespiegelte Datenbank nicht gestartet werden und sie versucht nicht, eine erneute Verbindung herzustellen. Wenn die gespiegelte Datenbank die einzige laufende Datenbank auf dem Datenbankserver ist, wird der Datenbankserver nicht gestartet.

- Standardverhalten für Aktualisierung materialisierter Ansichten** Bislang war das Standardverhalten bei der Aktualisierung materialisierter Ansichten WITH EXCLUSIVE MODE. Nun hängt das Standardverhalten davon ab, ob die materialisierte Ansicht als IMMEDIATE REFRESH definiert ist und ob die Snapshot-Isolationsstufe für die Datenbank aktiviert ist. Siehe „REFRESH MATERIALIZED VIEW-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- Verhaltensänderung der post_login_procedure-Datenbankoption** Die Standardeinstellung der Datenbankoption post_login_procedure ist nun die Systemprozedur sa_post_login_procedure. Siehe „post_login_procedure-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- non_keywords-Datenbankoption** In früheren Versionen konnten Sie individuelle Schlüsselwörter angeben und darüber hinaus alle Schlüsselwörter ab einer bestimmten Version deaktivieren, indem Sie einen der folgenden Spezialwerte in der Liste der Schlüsselwörter verwendeten:


```
keywords_4_0_d, keywords_4_0_c, keywords_4_0_b, keywords_4_0_a,
keywords_4_0,
keywords_5_0_01, keywords_5_0
```

Diese Spezialwerte werden nicht mehr unterstützt. Sie können weiterhin einzelne Schlüsselwörter deaktivieren. Siehe „non_keywords-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- cooperative_commit_timeout-Datenbankoption** Diese Optionseinstellung wird nun ignoriert, da das Festschreibeverhalten automatisch abgestimmt wird.
- cooperative_commits-Datenbankoption** Diese Optionseinstellung wird nun ignoriert, da das Festschreibeverhalten automatisch abgestimmt wird.
- Einstellung der quoted_identifier-Datenbankoption für Ferndatenzugriff berücksichtigt** Wenn Ferndatenzugriff verwendet wird, steuert die lokale Einstellung der Option quoted_identifier nun die Verwendung von Bezeichnern in Anführungszeichen für Adaptive Server Enterprise und Microsoft SQL Server. Wenn Sie z.B. die quoted_identifier-Option lokal auf "Off" festlegen, werden Bezeichner in Anführungszeichen für Adaptive Server Enterprise deaktiviert. Siehe:
 - „Serverklasse ASEODBC“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
 - „Serverklasse MSSODBC“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
- Änderungen des Bereichs der Datenbankoptionen scale und precision** In früheren Versionen konnten Sie die Optionen precision und scale für einzelne Benutzer oder temporär festlegen. Diese Einstellungen können jedoch die Wiederherstellbarkeit einer Datenbank beeinträchtigen. Wenn sich die temporären Einstellungen oder die Einstellungen auf Benutzerebene bei der Ausführung von DDL-Anweisungen, die Tabellen oder Domänen erstellen oder ändern, von den entsprechenden Einstellungen für PUBLIC-Benutzer unterscheiden, kommt es möglicherweise bei der Neuerstellung der Datenbank zu Problemen. Das folgende Verhalten gilt nun für die Optionen precision und scale:

Datenbankserverversion	Datenbank der Version 10 oder früher	Datenbank der Version 11	Datenbank mit Upgrade auf Version 11	Entladen einer Datenbank der Version 10 oder früher
11	PUBLIC-Einstellungen zulässig Benutzereinstellungen zulässig Temporäre Einstellungen <i>nicht</i> zulässig	PUBLIC-Einstellungen zulässig Benutzereinstellungen <i>nicht</i> zulässig Temporäre Einstellungen <i>nicht</i> zulässig	PUBLIC-Einstellungen zulässig Benutzereinstellungen <i>nicht</i> zulässig Temporäre Einstellungen <i>nicht</i> zulässig	PUBLIC-Einstellungen entladen Benutzereinstellungen beim Entladen verworfen
10 oder früher	PUBLIC-Einstellungen zulässig Benutzereinstellungen zulässig Temporäre Einstellungen zulässig	k.A.	k.A.	PUBLIC-Einstellungen entladen Benutzereinstellungen entladen

Datenbankserver der Version 10 und früher gestatten es weiterhin, die Optionen scale und precision temporär sowie für einzelne Benutzer festzulegen.

Vorsicht

Es wird nicht empfohlen, die temporären Einstellungen bzw. die Einstellungen auf Benutzerebene für die Datenbankoptionen scale und precision zu verwenden, da diese bei der Neuerstellung der Datenbank möglicherweise zu Problemen führen können. Außerdem ist das Verhalten des Datenbankservers u.U. nicht vorhersehbar.

Siehe:

- „precision-Option“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „scale-Option“ [[SQL Anywhere Server - Datenbankadministration](#)]

- **Verhaltensänderung der OPTION-Klausel** Die OPTION-Klausel für die Anweisungen INSERT, UPDATE, DELETE, SELECT, UNION, EXCEPT und INTERSECT gibt nun einen Fehler zurück, wenn Sie eine Datenbankoption festlegen, die von der Klausel nicht unterstützt wird. Siehe:
 - „INSERT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „UPDATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „DELETE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „SELECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „UNION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „EXCEPT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „INTERSECT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **Verhaltensänderung des Rollback-Logs für schreibgeschützte Datenbanken** In früheren Versionen wurden Vorgänge in schreibgeschützten Datenbanken, an denen transaktionale temporäre Objekte beteiligt waren, nicht als transaktional behandelt. Es wurden keine Rollback-Log-Informationen für sie beibehalten. In dieser Version verfügen transaktionale temporäre Objekte in schreibgeschützten Datenbanken über eine vollständige transaktionale Semantik. Sie unterliegen Festschreibungen, Rollbacks und Rollbacks auf Savepoints.
- **Änderungen der unterstützten Itanium 64-Bit-Plattformen** In früheren Versionen war eine vollständige 64-Bit-Version der Software für Windows Server 2003 auf Itanium II-Chips verfügbar. Für 64-Bit-Linux- und HP-UX-Betriebssysteme war eine Deployment-Version verfügbar.

In dieser Version ist nur die Deployment-Version für 64-Bit HP-UX verfügbar.

- **Verhaltensänderungen des Dienstprogramms Entladen (dbunload)** In früheren Versionen mussten die dbunload-Optionen -ea, -ek und -ep mit der Option -an oder -ar angegeben werden, um die Verschlüsselung für die neue Datenbank zu steuern. Wenn Sie nun eine Datenbank oder einen Teil daraus entladen und nicht neu laden, steuern die Optionen -ea, -ek und -ep die Verschlüsselung der erstellten Tabellendatendateien. Wenn Sie diese Dateien verwenden, um eine Datenbank aus Interactive SQL neu zu laden, müssen Sie den Chiffrierschlüssel als Parameter der READ-Anweisung angeben. Siehe „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

In früheren Versionen musste die dbunload-Version, die zum Extrahieren einer Datenbank verwendet wurde, nicht mit der Version des Datenbankservers übereinstimmen, auf dem die Datenbank ausgeführt wurde. Wenn Sie nun dbunload mit einer Datenbank der Version 10.0.0 oder höher verwenden, muss die Version von dbunload mit der Version des Datenbankservers übereinstimmen, der für den Zugriff auf die Datenbank verwendet wird. Wenn eine ältere Version von dbunload mit einem neueren Datenbankserver verwendet wird oder umgekehrt, wird ein Fehler gemeldet.

- **Verhaltensänderung des Extraktionsdienstprogramms (dbxtract)** In früheren Versionen musste die dbxtract-Version, die zum Extrahieren einer Datenbank verwendet wurde, nicht mit der Version des Datenbankservers übereinstimmen, auf dem die Datenbank ausgeführt wurde. Wenn Sie nun dbxtract mit einer Datenbank der Version 10.0.0 oder höher verwenden, muss die Version von dbxtract mit der Version des Datenbankservers übereinstimmen, der für den Zugriff auf die Datenbank verwendet wird. Wenn eine ältere Version von dbxtract mit einem neueren Datenbankserver verwendet wird oder umgekehrt, wird ein Fehler gemeldet.

- Verändertes Sperrverhalten** In früheren Versionen konnte eine UPDATE- oder DELETE-Anweisung, die auf Isolationsstufe 0 ausgeführt wurde, eine Zeilensperre für eine Zeile setzen, die nicht von der Anweisung betroffen war. Es ist nun weniger wahrscheinlich, dass eine UPDATE- oder DELETE-Anweisung eine Absichts- oder Exklusivsperr für eine Zeile setzt, die nicht von der Anweisung betroffen ist. Bei der Entwicklung von Anwendungen sollten Sie die Isolationsstufe 0 oder 1 mit den Anweisungen UPDATE und DELETE vorsichtig verwenden und sicherstellen, dass das Verhalten für Ihre Anwendung akzeptabel ist. Siehe „[Sperren während einer Aktualisierung](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*] und „[Sperren während des Löschs](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
- Änderungen von Eigenschaftsnamen** Folgende Datenbankeigenschaften wurden in dieser Version umbenannt:

Alter Name	Neuer Name
CacheHitsEng	CacheHits
CacheReadEng	CacheRead
DiskReadEng	DiskRead
ReadHint	DiskReadHint
ReadHintScatter	DiskReadHintPages
ReadHintScatterLimit	DiskReadHintScatterLimit

Weitere Hinweise finden Sie unter „[Verbindungs-, Datenbank- und Datenbankserveigenschaften](#)“ [*SQL Anywhere Server - Datenbankadministration*].

- Dienstprogramm für die Sprachauswahl (dblang)** In früheren Versionen wurde dieses Dienstprogramm nur installiert, wenn das IRDK (International Resources Development Kit) bei der Installation ausgewählt wurde. In dieser Version werden alle internationalen Ressourcen und das Dienstprogramm für die Sprachauswahl immer installiert.
- Standard-DBSpace für temporäre Tabellen und Indizes** Temporäre Tabellen können nur im DBSpace TEMPORARY erstellt werden. Wenn Sie den DBSpace SYSTEM in der IN-Klausel der CREATE TABLE-Anweisung angeben, wird die IN-Klausel ignoriert und die temporäre Tabelle wird im temporären DBSpace erstellt. Wenn Sie in der IN-Klausel der CREATE TABLE-Anweisung einen benutzerdefinierten DBSpace angeben, wird ein Fehler zurückgegeben. Außerdem wird die Option default_dbspace beim Erstellen temporärer Objekte ignoriert.
- Daten in temporäre Tabellen laden** Beim Laden in temporäre Tabellen ist es nun nicht mehr möglich, eine lokale temporäre Tabelle zu laden, für die ON COMMIT DELETE festgelegt wurde. In früheren Versionen konnten Sie Daten in eine lokale temporäre Tabelle laden, für die ON COMMIT DELETE ROWS festgelegt war.

Jetzt wird automatisch ein Autocommit ausgeführt, wenn Sie eine LOAD TABLE-Anweisung ausführen. In früheren Versionen wurde Autocommit nicht immer ausgeführt.

- **Datenbankserveroptionen** Die Serveroptionen -uc und -ui werden nun unter Mac OS X unterstützt. Bislang wurden sie nur unter Linux unterstützt. Unter Linux öffnet die Serveroption -ui das Fenster **Startoptionen für den Server**, zeigt das Datenbankserver-Meldungsfenster an und startet den Datenbankserver, unabhängig davon, ob der X Window-Server startet. Unter Mac OS X zeigt -ui die Datenbankservermeldungen in einem neuen Fenster an und startet den Datenbankserver im Shell-Modus, wenn keine nutzbare Anzeige verfügbar ist. Die Serveroption -uc startet den Datenbankserver im Shell-Modus. Siehe „Datenbankserveroption -uc“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Datenbankserveroption -um“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Ferndatenzugriff nur noch mit ODBC-Treibern, die UNICODE-Aufrufe unterstützen** Ferndatenzugriff ist nur noch mit ODBC-Treibern möglich, die UNICODE-Aufrufe unterstützen. Aus diesem Grund wird bei der Verwendung von Nicht-UNICODE ODBC-Treibern beim Ferndatenzugriff keine Zeichensatzkonvertierung für Daten ausgeführt, die vom ODBC-Treiber stammen.
- **SYSFILE-Systemansicht** In der SYSFILE-Kompatibilitätsansicht ist nun eine Zeile für die temporäre Datei enthalten.

Nicht mehr empfohlene und nicht mehr weiterentwickelte SQL Anywhere-Funktionen

- **FORMAT ASCII-Klausel für Anweisungen LOAD TABLE, UNLOAD TABLE, INPUT und OUTPUT nicht mehr empfohlen** Die FORMAT ASCII-Klausel für die Anweisungen LOAD TABLE, UNLOAD TABLE, INPUT und OUTPUT wird nicht mehr empfohlen und wurde durch FORMAT TEXT ersetzt. Dienstprogramme, z. B. dbunload, generieren nun Reload-Skripten mit FORMAT TEXT anstelle von FORMAT ASCII.

Für die OUTPUT-Anweisung schreibt die FORMAT TEXT-Klausel nun die Daten in demselben Dateiformat wie FORMAT ASCII in früheren Versionen. Die bislang von FORMAT TEXT erstellte Ausgabe ist nicht mehr verfügbar.

- **Datenbankeigenschaften** Folgende Datenbankeigenschaften werden nicht unterstützt:
 - MapPages
 - PreserveSource
 - UniqueIdentifier
- **Servereigenschaften** Folgende Servereigenschaften werden in dieser Version nicht mehr empfohlen:
 - MaxMessage
 - Message
 - MessageTime
 - MessageText
 - MessageWindowSize

- **SPX-Protokoll nicht unterstützt** Die Unterstützung für das SPX-Protokoll wurde in dieser Version entfernt. Aus diesem Grund wurden folgende Protokolloptionen entfernt:
 - ExtendedName-Protokolloption [ENAME]
 - RegisterBindery-Protokolloption [REGBIN]
 - SearchBindery-Protokolloption [BINSEARCH]

Folgende Funktionen des SQL Anywhere .NET-Datenproviders werden nicht mehr weiterentwickelt:

- SACommLinksOptionsBuilder-Klasse: SpxOptionsBuilder-Eigenschaft
 - SACommLinksOptionsBuilder-Klasse: SpxOptionsString-Eigenschaft
 - SASpxOptionsBuilder-Klasse
- **dbinit-Option -e nicht unterstützt** Die Option -e von dbinit, mit der bei der Datenbankerstellung die einfache Verschlüsselung festgelegt wurde, wird nicht mehr unterstützt. Verwenden Sie die Option "-ea simple", um die einfache Verschlüsselung festzulegen. Siehe „[Dienstprogramm Initialisierung \(dbinit\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Nicht mehr unterstützte Datenbankoptionen** Die Unterstützung der folgenden Datenbankoptionen und ihrer zugehörigen Datenbankeigenschaften wurde aus dieser Version entfernt.

Optionen	Verhalten in dieser Version
ansi_integer_overflow	Ein Überlauf führt nun immer zu einem SQLSTATE = 22003 - overflow-Fehler. Beim Entladen oder Verbinden mit älteren Datenbanken mit materialisierten Ansichten wird die Einstellung dieser Option ignoriert.
ansi_substring	Das Verhalten der Funktion SUBSTRING entspricht nun dem ANSI/ISO SQL/2003-Verhalten. Siehe „ SUBSTRING-Funktion [Zeichenfolge] “ [SQL Anywhere Server - SQL-Referenzhandbuch].
automatic_timestamp	Neue Spalten mit dem Datentyp TIMESTAMP, für die kein expliziter Standardwert festgelegt ist, erhalten nie einen Standardwert vom Transact-SQL-Zeitstempel (timestamp).
divide_by_zero_error	Ein Nullteilungsfehler führt nun zu einem Fehler mit SQLSTATE 22012. Beim Entladen oder Verbinden mit älteren Datenbanken mit materialisierten Ansichten wird die Einstellung dieser Option ignoriert.

Optionen	Verhalten in dieser Version
float_as_double	<p>In SQL Anywhere verhält sich das FLOAT-Schlüsselwort nie wie das FLOAT-Schlüsselwort von Adaptive Server Enterprise, wenn keine Gesamtstellenzahl festgelegt ist. SQL Anywhere behandelt FLOAT-Werte nicht auf dieselbe Weise wie DOUBLE-Werte.</p> <p>Bei Open Client- und jConnect-Verbindungen unterscheidet sich dieses Verhalten vom Standardverhalten in früheren Versionen.</p> <p>Beim Entladen oder Verbinden mit älteren Datenbanken mit materialisierten Ansichten wird die Einstellung dieser Option ignoriert.</p>
optimistic_wait_for_commit	Diese Option wird nicht mehr unterstützt.
query_plan_on_open	Es wird kein Plan mehr zurückgegeben, wenn eine OPEN-Anweisung für einen Cursor ausgeführt wird. Eine vollständigere Beschreibung erhalten Sie mithilfe der EXPLAIN-Anweisung oder der PLAN-Funktion. Siehe „EXPLAIN-Anweisung [ESQL]“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>] und „PLAN-Funktion [Verschiedene]“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>].
ri_trigger_time	Aktionen zur referenziellen Integrität werden nun nach den Anweisungen UPDATE oder DELETE ausgeführt.
truncate_with_auto_commit	Es wird nun sowohl vor als auch nach einer TRUNCATE TABLE-Anweisung eine COMMIT-Anweisung ausgeführt.
tsql_hex_constant	Hexadezimale Konstante werden nun wie binäre Konstante behandelt.
uuid_has_hyphens	<p>UUID-Zeichenfolgen enthalten nun vier Bindestriche.</p> <p>Beim Entladen oder Verbinden mit älteren Datenbanken mit materialisierten Ansichten wird die Einstellung dieser Option ignoriert.</p>
percent_as_comment	In früheren Versionen konnte das Prozentzeichen (%) abhängig von der Einstellung der Datenbankoption percent_as_comment als Kommentarmarkierung verwendet werden. Nun behandelt SQL Anywhere das Prozentzeichen als Modulo-Operator. Siehe „MOD-Funktion [Numerisch]“ [<i>SQL Anywhere Server - SQL-Referenzhandbuch</i>].

- SQLANYSH10-Umgebungsvariable nicht unterstützt** In früheren Versionen wurde ein Teil der SQL Anywhere-Software in einem freigegebenen Verzeichnis installiert. Dieser Speicherort konnte über die Umgebungsvariable SQLANYSH10 festgelegt werden. Es wird keine Software mehr in einem freigegebenen Verzeichnis installiert und die Umgebungsvariable SQLANYSH10 wird nicht mehr verwendet.

Bei der Verwendung einer dialogfreien Installation müssen Sie den Speicherort nicht mehr mit der Umgebungsvariablen SHARED_DIR festlegen. Siehe „[Dialogfreie Installationen mit dem SQL Anywhere-Installationsprogramm](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **sa_get_server_messages-Systemprozedur nicht mehr unterstützt** In früheren Versionen konnten Sie die Systemprozedur sa_get_server_messages verwenden, um Konstante aus dem Meldungsfenster des Datenbankservers als Ergebnismenge zurückzugeben. Nun können Sie diese Informationen mit der Systemprozedur sa_server_messages abfragen. Siehe „[sa_server_messages-Systemprozedur](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)].
- **background_priority-Option nicht mehr empfohlen** Die background_priority-Option wird nicht mehr weiterentwickelt. Verwenden Sie stattdessen die Option priority. Siehe „[priority-Option](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **encrypt_aes_random_iv-Option nicht mehr empfohlen** Die Unterstützung der Datenbankoption encrypt_aes_random_iv wurde in dieser Version entfernt. Nun wird immer ein zufälliger IV (Initialisierungsvektor) verwendet.
- **DLL-Protokolloption nicht unterstützt** Die Unterstützung für die DLL-Protokolloption wurde entfernt. Windows-Datenbankserver und -Clients verwenden Winsock 2.2. Windows Mobile-Clients verwenden Winsock 1.1. Siehe „[TCP/IP-Protokoll](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **SQL Anywhere Broadcast Repeater-Dienstprogramm umbenannt** In Version 10 wurde das SQL Anywhere Broadcast Repeater-Dienstprogramm mit dem Befehl dbns10 ausgeführt. In dieser Version heißt der Befehl dbns11. Siehe „[Broadcast Repeater-Dienstprogramm \(dbns16\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Umgebungsvariablen SQLANY10 und SQLANYSAMP10 umbenannt** Die Umgebungsvariablen SQLANY10 und SQLANYSAMP10 wurden in SQLANY11 bzw. SQLANYSAMP11 umbenannt. Siehe:
 - „[SQLANY16-Umgebungsvariable](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „[SQLANYSAMP16-Umgebungsvariable](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- **Einige gesicherte Funktionen umbenannt** Die folgenden gesicherten Funktionen wurden für diese Version umbenannt:

Nicht mehr empfohlener Name	Neuer Name
xp_read_file	read_file
xp_write_file	write_file
unload_table	write_file
load_table	read_file

Weitere Hinweise finden Sie unter „Datenbankserveroption -sf“ [[SQL Anywhere Server - Datenbankadministration](#)].

MobiLink

In den folgenden Abschnitten finden Sie Beschreibungen der neuen Funktionen und Verhaltensänderungen in MobiLink für Version 11.0.0 und Informationen dazu, welche Funktionen nicht mehr empfohlen werden.

Neue MobiLink-Funktionen

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in MobiLink in Version 11.0.0 hinzugefügt wurden.

Konsolidierte Datenbanken

- **DB2-Mainframe nun als konsolidierte Datenbank unterstützt** MobiLink unterstützt bereits seit langem DB2 LUW (Linux, Unix und Windows) als konsolidierte Datenbank. Nun wird auch DB2-Mainframe unterstützt.
- **MySQL nun als konsolidierte Datenbank unterstützt** MobiLink unterstützt nun MySQL als konsolidierte Datenbank.

Siehe „[Konsolidierte MySQL-Datenbank](#)“ [[MobiLink - Serveradministration](#)].

- **MobiLink-Systemdatenbank (MLSD)** Unterstützung für eine separate Datenbank zum Speichern von MobiLink-Systemdaten (MLSD, MobiLink-Systemdatenbank) wurde hinzugefügt. Diese Funktion muss zusammen mit Microsoft DTC (Distributed Transaction Coordinator) verwendet werden. Siehe „[mlsrv16-Option -cs](#)“ [[MobiLink - Serveradministration](#)].

Neue Systemobjekte

- **Neue Systemtabellen und neues Schema für MobiLink-Server** Im Folgenden sind die Änderungen der MobiLink-Systemtabellen aufgeführt:

- Es wurden mehrere neue MobiLink-Systemtabellen hinzugefügt:
 - ml_qa_delivery_archive
 - ml_qa_repository_archive
 - ml_qa_repository_props_archive
 - ml_qa_status_history_archive
 - ml_server
 - ml_active_remote_id
 - ml_passthrough
 - ml_passthrough_repair
 - ml_passthrough_script
 - ml_passthrough_status
- Es wurden mehrere neue MobiLink-Systemprozeduren hinzugefügt. Siehe:
 - „ml_add_passthrough-Systemprozedur“ [[MobiLink - Serveradministration](#)]
 - „ml_add_passthrough_repair-Systemprozedur“ [[MobiLink - Serveradministration](#)]
 - „ml_add_passthrough_script-Systemprozedur“ [[MobiLink - Serveradministration](#)]
 - „ml_delete_passthrough-Systemprozedur“ [[MobiLink - Serveradministration](#)]
 - „ml_delete_passthrough_repair-Systemprozedur“ [[MobiLink - Serveradministration](#)]
 - „ml_delete_passthrough_script-Systemprozedur“ [[MobiLink - Serveradministration](#)]

iAnywhere Solutions Oracle-Treiber

- **Oracle DSN kann verschlüsseltes Kennwort speichern** Beim Erstellen einer Oracle ODBC-Datenquelle im Windows ODBC-Administrator können Sie nun auswählen, das in der ODBC-Datenquelle gespeicherte Kennwort zu verschlüsseln. Siehe „[SQL Anywhere 16 - Oracle-ODBC-Treiber](#)“ [[MobiLink - Serveradministration](#)].
- **Oracle ODBC-Treiber unterstützt verteilte Microsoft-Transaktionen** Der Oracle ODBC-Treiber unterstützt nun verteilte Microsoft-Transaktionen. Wählen Sie im Windows ODBC-Administrator "Verteilte Microsoft-Transaktionen aktivieren" aus, und stellen Sie sicher, dass die geeignete DLL auf dem Oracle-Client installiert ist. Siehe „[SQL Anywhere 16 - Oracle-ODBC-Treiber](#)“ [[MobiLink - Serveradministration](#)].

MobiLink-Server

- **Relay Server** Beim Relay Server handelt es sich um eine Gruppe von Weberweiterungen, die eine sichere Kommunikation mit Lastenverteilung zwischen mobilen Geräten und MobiLink-, Afaria- und Onebridge-Servern ermöglichen, die über einen Webserver kommunizieren. Siehe „[Einführung in den Relay Server](#)“ [[Relay Server](#)].
- **Sybase Relay Server-Hostingdienst** Der Sybase Relay Server-Hostingdienst ist eine Farm von Relay Servern, die von Sybase gehostet wird. Dieser Dienst vereinfacht die Entwicklung mobiler Anwendungen, die die MobiLink-Datensynchronisation verwenden, sowie den Evaluierungsprozess für Entwickler, insbesondere für Anwendungen, bei denen Daten über öffentliche drahtlose Netzwerke gesendet werden. Siehe „[Sybase Hosted Relay Service](#)“ [[Relay Server](#)].
- **MobiLink-Serverfarm** MobiLink Server können nun in Serverfarmen von identischen Servern eingeteilt werden.

Redundante gleichzeitige Synchronisationen von derselben ID werden nun in der gesamten Farm automatisch erkannt. Dadurch ist es nicht mehr erforderlich, dass ein Lastverteiler dieselbe entfernte ID weiter an denselben MobiLink-Server sendet.

Sie können nun alle MobiLink-Server identisch konfigurieren.

Failover wird für den Notifier automatisch unterstützt. Die Farm wählt automatisch einen MobiLink-Server zur Ausführung dieser Programme aus. Wenn der erste gewählte Computer ausfällt, wählt die Farm einen neuen Server aus.

- **64-Bit-Plattformen** MobiLink-Server unterstützt nun die 64-Bit-Verarbeitung auf mehreren 64-Bit-Plattformen in vollem Umfang. Eine Liste der unterstützten 64-Bit-Plattformen finden Sie unter <http://www.sybase.com/detail?id=1002288>.
- **Neues Mitglied für Java-Schnittstelle DownloadTableData** `getLastDownloadTime`-Methode zum Zurückgeben der letzten Downloadzeit für eine Tabelle. Siehe [DownloadTableData.getLastDownloadTime-Methode \[MobiLink-Server Java\]](#) [*MobiLink - Serveradministration*].
- **SQL Passthrough** Die SQL-Passthrough-Funktion gestattet es, Skripten mit SQL-Anweisungen von einer konsolidierten Datenbank auf einen SQL Anywhere- oder UltraLite-Client herunterzuladen und diese SQL-Anweisungen auf dem Client zum geeigneten Zeitpunkt auszuführen.
- **Benachrichtigungen für informative Meldungen** Die Java- und .NET-API erlauben nun Benutzern die Registrierung, um jeweils Benachrichtigungen zu erhalten, wenn eine informative Zeile mit dem Präfix "I" in das Log geschrieben wird.

Siehe:

- [LogMessage.INFO-Variable \[MobiLink-Server Java\]](#) [*MobiLink - Serveradministration*]
- [ServerContext.addInfoListener-Methode \[MobiLink-Server Java\]](#) [*MobiLink - Serveradministration*]
- [ServerContext.removeInfoListener-Methode \[MobiLink-Server Java\]](#) [*MobiLink - Serveradministration*]
- [LogMessage.MessageType-Enumeration \[MobiLink-Server .NET\]](#) [*MobiLink - Serveradministration*]
- [ServerContext.InfoListener-Ereignis \[MobiLink-Server .NET\]](#) [*MobiLink - Serveradministration*]

Neue mlsrv11-Funktionen

- **Option -cs** MobiLink-Server-Systemobjekte, z. B. Systemtabellen, Prozeduren, Trigger und Ansichten, können nun in einer anderen als der konsolidierten Datenbank gespeichert werden. Die Datenbank, die die MobiLink-Systemobjekte speichert, heißt MobiLink-Systemdatenbank oder MLSD.

Mithilfe von `-cs` können Sie Verbindungsparameter für die MLSD festlegen. Siehe „[mlsrv16-Option -cs](#)“ [*MobiLink - Serveradministration*].

- **Option -lsc** Diese Option legt die Informationen für die lokale Serververbindung fest. Diese Informationen werden anderen Servern in der Serverfarm übergeben. Siehe „[mlsrv16-Option -lsc](#)“ [*MobiLink - Serveradministration*].

- **Option -ss** Diese Option aktiviert den MobiLink-Server, sodass er in einer Serverfarm ausgeführt wird.

Hinweis

Die Option -ss ist eine Funktion der High Availability-Option von MobiLink, für die eine separate Lizenz erforderlich ist. Siehe „[Getrennt lizenzierbare Komponenten](#)“ [[SQL Anywhere 16 - Einführung](#)].

- **Option -tc** Mit dieser Option können Sie den Countdown-Timer für die Ausführung von SQL-Skripten festlegen. Siehe „[mlsrv16-Option -tc](#)“ [[MobiLink - Serveradministration](#)].
- **Option -tf** Legt fest, dass die Ausführung des SQL-Skripts fehlschlägt, wenn der mit -tc festgelegte Countdown-Timer abläuft (nicht für Oracle). Siehe „[mlsrv16-Option -tf](#)“ [[MobiLink - Serveradministration](#)].

Neue MobiLink-Skriptfunktionen

- **Nicht blockierende Download-ACK-Skripten** Siehe „[nonblocking_download_ack](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)] und „[publication_nonblocking_download_ack](#) (Verbindungsereignis)“ [[MobiLink - Serveradministration](#)].

Erweiterungen des MobiLink-Redirectors

- **Redirector nicht mehr weiterentwickelt** Der Redirector wird nicht mehr weiterentwickelt. Er wurde durch den Relay Server ersetzt. Siehe „[Einführung in den Relay Server](#)“ [[Relay Server](#)].
- **Neuer Relay Server** Beim Relay Server handelt es sich um eine Gruppe von Weberweiterungen, die eine sichere Kommunikation mit Lastenverteilung zwischen mobilen Geräten und MobiLink-, Afaria- und Onebridge-Servern ermöglichen, die über einen Webserver kommunizieren. Siehe „[Einführung in den Relay Server](#)“ [[Relay Server](#)].

MobiLink-Clients

SQL Anywhere-Clients

- **dbmlsync-APIs für C++ und .NET** Die Dbmlsync-API stellt eine Programmierschnittstelle bereit, die es in C++ oder .NET geschriebenen MobiLink-Clientanwendungen gestattet, Synchronisationen zu starten und Rückmeldungen über den Fortschritt der angeforderten Synchronisationen zu empfangen. Siehe „[Dbmlsync-API](#)“ [[MobiLink - Clientadministration](#)].
- **Synchronisationsprofile** Sie können nun dbmlsync-Befehlszeilen in der Datenbank in Form von Synchronisationsprofilen speichern. Mit der dbmlsync-Option -sp können Sie Optionen aus einem Synchronisationsprofil zu den Synchronisationsoptionen in der Befehlszeile hinzufügen. Siehe „[dbmlsync-Option -sp](#)“ [[MobiLink - Clientadministration](#)].
- **LOAD TABLE protokolliert nun optional geladene Zeilen als Einfügungen** Die neue Klausel WITH ROW LOGGING bewirkt, dass entfernte SQL Anywhere-Datenbanken nun Daten mit der LOAD TABLE-Anweisung laden können. Bislang war LOAD TABLE in einer Synchronisationsumgebung nicht immer nützlich, da die Zeilen nicht protokolliert und daher von

dbmlsync ignoriert wurden. Siehe „Datenimport mit der LOAD TABLE-Anweisung“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

- **Log-Scan-Optimierung für dbmlsync** Log-Scans sind nun für nicht überlappende Publikationen optimiert.

Sicherheit

Folgende Sicherheitsfunktionen wurden in dieser Version hinzugefügt:

- **Ende-zu-Ende-Verschlüsselung** MobiLink-Synchronisationsdatenströme und -Clients unterstützen jetzt durchgehende Verschlüsselung auf Protokollebene. Siehe „Ende-zu-Ende-Verschlüsselung“ [*SQL Anywhere Server - Datenbankadministration*].
- **Schlüsselpaargenerator-Dienstprogramm (createkey)** Dieses Dienstprogramm erstellt RSA- und ECC-Schlüsselpaare, die bei der Ende-zu-Ende-Verschlüsselung von MobiLink verwendet werden. Siehe „Schlüsselpaargenerator-Dienstprogramm (createkey)“ [*SQL Anywhere Server - Datenbankadministration*].

Serverinitiierte Synchronisation

- **Unterstützung für MobiLink-Serverfarm** Die serverinitiierte Synchronisation wurde erweitert und ist nun besser auf die Verwendung in einer MobiLink-Serverfarmumgebung abgestimmt. Sie können nun auf jedem MobiLink-Server in der Farm einen Notifier ausführen. Die Notifier stellen gemeinsam sicher, dass keine redundanten Benachrichtigungen für einen Listener vorhanden sind. Siehe „mlsrv16-Option -lsc“ [*MobiLink - Serveradministration*].

Listener-Erweiterungen

- **Neue dblns-Optionen** Folgende Optionen wurden dem MobiLink-Listener für Windows hinzugefügt:
 - **-ni** Diese Option deaktiviert die IP-Protokollierung.
 - **-pc** Diese Option deaktiviert eine beständige Verbindung.
 - **-nu** Diese Option deaktiviert die standardmäßige UDP-Überwachung.

MobiLink-Verhaltensänderungen und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen, die in MobiLink in Version 11.0.0 vorgenommen wurden.

Änderungen des MobiLink-Servers

- **Änderung der standardmäßigen Downloadbestätigung** Der Standardwert für die Option `-nba` ist nun `-nba+`. Mit diesem Wert wird die Downloadbestätigung standardmäßig nicht blockiert. Wenn Sie die Blockierung der Downloadbestätigung in einem vorhandenen Deployment verwenden, müssen Sie eine der beiden folgenden Vorgehensweisen wählen:
 - Option `-nba-` verwenden
 - Synchronisationsskripten so einrichten, dass sie die Skripten `nonblocking_download_ack` bzw. `publication_nonblocking_download_ack` verwenden (empfohlen)
- **Protokolloptionen `certificate` und `certificate_password` umbenannt** Die TLS- und HTTPS-Protokolloptionen `certificate` und `certificate_password` wurden in `identity` bzw. `identity_password` umbenannt. Siehe „[mlsrv16-Option -x](#)“ [*MobiLink - Serveradministration*].

Änderungen des MobiLink-Clients

- **Dbmlsync-Integrationskomponente nicht mehr empfohlen** Die Dbmlsync-Integrationskomponente wird nicht mehr empfohlen. Sie wurde durch die dbmlsync-API ersetzt.

Siehe „[Dbmlsync-API](#)“ [*MobiLink - Clientadministration*].
- **Erweiterte dbmlsync-Option `StreamCompression` nicht mehr unterstützt** Diese Option wird nicht mehr unterstützt.
- **Erweiterte Option `-lt` hat nun den Standardwert `OFF`** Die erweiterte Option `LockTables (-lt)` für dbmlsync hatte bislang den Standardwert `ON`. Nun hat sie den Standardwert `OFF`, d. h. dbmlsync sperrt Synchronisationstabellen standardmäßig nicht. Siehe „[Erweiterte Option `LockTables \(-lt\)`](#)“ [*MobiLink - Clientadministration*].

Sonstige MobiLink-Verhaltensänderungen

Serverinitiierte Synchronisation

- **Sierra Wireless Aircards nicht mehr unterstützt** Die SMS-Listening-Bibliotheken für Sierra Wireless Aircards werden nicht mehr unterstützt.
- **Option `-g` nicht mehr empfohlen** Die Listener-Option `-g` wurde durch die Option `-ni` ersetzt.
- **Option `-ga` nicht mehr empfohlen** Die Listener-Option `-ga` wird nicht mehr unterstützt. Die asynchrone IP-Protokollierung wird nun implizit ausgeführt.
- **Standardwert für `-gi` geändert** Das Standardabrufintervall wurde von 10 Sekunden auf 60 Sekunden geändert.

SQL Remote

In den folgenden Abschnitten finden Sie Beschreibungen der neuen Funktionen und Verhaltensänderungen in SQL Remote für Version 11.0.0 und Informationen dazu, welche Funktionen nicht mehr empfohlen werden.

Neue Funktionen in SQL Remote

- **Erweiterungen der Option -ea des Extraktionsdienstprogramms (dbxtract)** Die Option -ea für dbxtract akzeptiert nun sowohl "none" als auch "simple" als Verschlüsselungstypen. Die Angabe von "none" führt zu keiner Verschlüsselung. Die Angabe von "simple" führt zu einer einfachen Verschlüsselung. Außerdem wurde der Standardverschlüsselungstyp geändert, abhängig davon, ob die Optionen -ek, -et oder -ep mit -ea angegeben werden. Siehe „[Extraktionsdienstprogramm \(dbxtract\)](#)“ [[SQL Remote](#)].
- **Optionen -ap, -er, -et und -nl dem Extraktionsdienstprogramm (dbxtract) hinzugefügt** Sie können nun die Optionen -ap, -er, -et und -nl zusammen mit dbxtract verwenden. Siehe „[Extraktionsdienstprogramm \(dbxtract\)](#)“ [[SQL Remote](#)].

Verhaltensänderungen von SQL Remote und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen, die in SQL Remote in Version 11.0.0 vorgenommen wurden.

- **LOAD TABLE protokolliert nun optional geladene Zeilen als Einfügungen** Die neue Klausel WITH ROW LOGGING bewirkt, dass entfernte SQL Anywhere-Datenbanken nun Daten mit der LOAD TABLE-Anweisung laden können. Bislang war LOAD TABLE in einer Synchronisations- oder Replikationsumgebung nicht immer nützlich, da die Zeilen nicht protokolliert und daher von dbremote oder dbltm ignoriert wurden. Siehe „[Datenimport mit der LOAD TABLE-Anweisung](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].
- **Nachrichtentypen VIM und MAPI nicht unterstützt** Die Unterstützung für die Nachrichtensysteme VIM und MAPI wurde in dieser Version entfernt. Wenn Sie ein Upgrade einer Datenbank, die VIM oder MAPI verwendet, auf SQL Anywhere Version 11.0.0 vornehmen, müssen Sie den Nachrichtentyp auf File, FTP oder SMTP ändern. *Dbremote.exe* startet nicht, wenn der Nachrichtentyp MAPI oder VIM ist.

Die einfachste Änderung ist der Wechsel zu SMTP/POP. Die Änderung des Nachrichtentyps kann eine Änderung des Mailservers erfordern, sodass dieser SMTP/POP unterstützt.

Weitere Hinweise zur Auswahl eines SQL Remote-Nachrichtentyps finden Sie unter „[SQL Remote-Nachrichtensysteme](#)“ [[SQL Remote](#)].

UltraLite

In den folgenden Abschnitten finden Sie Beschreibungen der neuen Funktionen und Verhaltensänderungen in UltraLite für Version 11.0.0 und Informationen dazu, welche Funktionen nicht mehr empfohlen werden.

Neue Funktionen von UltraLite

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in UltraLite in Version 11.0.0 hinzugefügt wurden.

Wichtigste Funktionen

- **Unterstützung für Synchronisationsprofile** UltraLite 11.0.0 und 11.0.1 unterstützt Synchronisationsprofile. Siehe „[ALTER SYNCHRONIZATION PROFILE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#), „[CREATE SYNCHRONIZATION PROFILE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#) und „[DROP SYNCHRONIZATION PROFILE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#).
- **UltraLite SELECT-Anweisung** Der Standardwert für UltraLite-SELECT-Anweisungen, die keine explizite FOR-Klausel enthalten, ist nun FOR READ ONLY. Durch diese Änderung kann UltraLite die bestmöglichen Pläne für Abfragen auswählen, wenn keine Aktualisierungen zulässig sind. Siehe „[SELECT-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#).
- **UltraLite SYNCHRONIZE-Anweisung** Dies ist eine neue Anweisung zur Synchronisation eines UltraLite-Synchronisationsprofils oder spezifischer Synchronisationsoptionen. Siehe „[SYNCHRONIZE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#).
- **UltraLite CREATE SYNCHRONIZATION PROFILE-Anweisung** Dies ist eine neue Anweisung zur Erstellung eines UltraLite-Synchronisationsprofils. Synchronisationsprofile legen fest, wie eine UltraLite-Datenbank mit dem MobiLink-Server synchronisiert wird. Siehe „[CREATE SYNCHRONIZATION PROFILE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#).
- **UltraLite ALTER SYNCHRONIZATION PROFILE-Anweisung** Dies ist eine neue Anweisung zur Änderung eines UltraLite-Synchronisationsprofils. Siehe „[ALTER SYNCHRONIZATION PROFILE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#).
- **UltraLite DROP SYNCHRONIZATION PROFILE-Anweisung** Dies ist eine neue Anweisung zur Löschung eines UltraLite-Synchronisationsprofils. Siehe „[DROP SYNCHRONIZATION PROFILE-Anweisung \[UltraLite\]](#)“ [\[UltraLite - Datenbankverwaltung\]](#).
- **Unterstützung für Passthrough-Skripten von SQL Anywhere** Diese Funktionalität wird in UltraLite 12 nicht unterstützt.

UltraLite-Dienstprogramme beziehen nun die Unterstützung von Passthrough-Skripten in SQL Anywhere ein. Die Änderungen beziehen sich auf folgende Dienstprogramme:

- ulcond11
- ulunload
- ulload
- ulinfo
- ulsync

Siehe:

- „UltraLite-Dienstprogramm zum Entladen von Datenbanken (ulunload)“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Dienstprogramm zum Laden von Daten aus XML-Dateien (ulload)“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Informationsdienstprogramm (ulinfo)“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Synchronisationsdienstprogramm (ulsync)“ [[UltraLite - Datenbankverwaltung](#)]
- **UltraLite-Datenbankvalidierung** Sie können nun das ulvalid-Dienstprogramm oder die ValidateDatabase-API verwenden, um eine UltraLite-Datenbank zu validieren. Die Validierung prüft die Datenbankdatei auf bestimmte Typen von Beschädigungen. Sie können die Ergebnisse mithilfe von Befehlszeilenparametern verfeinern. Siehe „UltraLite-Dienstprogramm zum Validieren von Datenbanken (ulvalid)“ [[UltraLite - Datenbankverwaltung](#)] und „Eine UltraLite-Datenbank validieren“ [[UltraLite - Datenbankverwaltung](#)].

UltraLite.NET unterstützt nun die Funktion ValidateDatabase. Sie können nun eine Datenbank oder bestimmte Tabellen mit oder ohne Verbindung validieren. Siehe [ULDatabaseManager-Klasse](#) [[UltraLite.NET](#)] [[UltraLite - .NET-Programmierung](#)] und [ULConnection-Klasse](#) [[UltraLite.NET](#)] [[UltraLite - .NET-Programmierung](#)].

Sie können jetzt den **Assistenten zum Validieren einer Datenbank** in Sybase Central verwenden, um eine UltraLite-Datenbank zu validieren. Die Menüoption **Datenbank validieren** ist über das Menü **Extras** verfügbar.

- **Unterstützung für Ereignisse und Benachrichtigungen** UltraLite unterstützt nun Ereignisse und Benachrichtigungen. Benachrichtigungen werden beim Eintreten von Ereignissen an registrierte Warteschlangen oder Verbindungen gesendet. Benutzerereignisse können auch von Anwendungen definiert und ausgelöst werden. APIs für Ereignisse und Benachrichtigungen werden in jeder unterstützten Sprache bereitgestellt. Außerdem wird eine SQL-Funktion für den Zugriff auf die API-Funktionalität bereitgestellt.
- **UltraLite-Unterstützung für Isolationsstufen** Verbindungen werden nun standardmäßig voneinander isoliert. Nicht festgeschriebene Änderungen von anderen Verbindungen und Downloads sind erst sichtbar, wenn sie festgeschrieben wurden.

Sie können nun die Isolationsstufe auf READ_COMMITTED oder READ_UNCOMMITTED festlegen. Siehe „Isolationsstufen“ [[UltraLite - Datenbankverwaltung](#)] und „UltraLite-Transaktionsverarbeitung“ [[UltraLite - Datenbankverwaltung](#)].

UltraLite.NET unterstützt nun die Isolationsstufe ReadUncommitted. Die Standardisolationsstufe einer Verbindung im Autocommit-Modus ist ReadCommitted. Siehe „[UltraLite-](#)

Transaktionsverarbeitung“ [[UltraLite - Datenbankverwaltung](#)] und „Isolationsstufen“ [[UltraLite - Datenbankverwaltung](#)].

- **UltraLite ALTER DATABASE SCHEMA FROM FILE-Anweisung** Sie können nun ein UltraLite-Schema mit der Anweisung ALTER DATABASE SCHEMA FROM FILE ändern. Die Anweisung ALTER DATABASE SCHEMA FROM FILE ersetzt die Schema-Upgrade-Funktion aus Version 9.0.2, die mit den nun entfernten Methoden UpgradeSchemaFromFile bzw. ApplyFile implementiert wurde. Verwenden Sie die Dienstprogramme ulinit oder ulunload, um sicherzustellen, dass die DDL-Anweisungen syntaktisch einwandfrei sind.

Siehe:
 - „ALTER DATABASE SCHEMA FROM FILE-Anweisung [UltraLite]“ [[UltraLite - Datenbankverwaltung](#)]
 - „Deployment von UltraLite-Datenbankschema-Upgrades“ [[UltraLite - Datenbankverwaltung](#)]
 - „UltraLite-Dienstprogramm zum Initialisieren einer Datenbank (ulinit)“ [[UltraLite - Datenbankverwaltung](#)]
 - „UltraLite-Dienstprogramm zum Entladen von Datenbanken (ulunload)“ [[UltraLite - Datenbankverwaltung](#)]
- **Assistent zum Extrahieren einer Datenbank - Verhaltensänderungen** Sie können nun Tabellen aus dem Extraktionsvorgang ausschließen. Der **Assistent zum Extrahieren einer Datenbank** bezieht keine Publikationen mit mehrfach vorhandenen Namen aus der Liste der verfügbaren Publikationen ein. Siehe „[UltraLite-Upgrades](#)“ auf Seite 337.
- **UltraLite-Clientversion und Build-Nummer den MobiLink-Logdateien hinzugefügt** Während der Synchronisation fügen UltraLite-Clients nun dem MobiLink-Serverlog ihre Versions- und Build-Nummer hinzu. Siehe:
 - „MobiLink-Serverlogs anzeigen“ [[MobiLink - Serveradministration](#)]
 - „UltraLite-Synchronisationsparameter“ [[UltraLite - Datenbankverwaltung](#)]
- **LOAD TABLE-Anweisung von UltraLite** Die LOAD TABLE-Anweisung kann nun auf PCs ausgeführt werden. Siehe "LOAD TABLE-Anweisung von UltraLite". Siehe „[LOAD TABLE-Anweisung \[UltraLite\]](#)“ [[UltraLite - Datenbankverwaltung](#)].
- **Unterstützung der Hintergrundsynchrisation** Sie können nun eine Synchronisation an jedem Punkt der Anwendung auf einem separaten Thread beginnen, und UltraLite lädt nur die Zeilen herauf, die zum Zeitpunkt des Upload-Beginns festgeschrieben waren. Sie können nun die Datenbank während des Uploads ändern und die Änderungen festschreiben, ohne den Upload zu beeinträchtigen. Alle Zeilen, die während des Uploads festgeschrieben werden, werden vom Upload ignoriert. Siehe „[UltraLite-Parallelität](#)“ [[UltraLite - Datenbankverwaltung](#)].
- **Erweiterte Unterstützung von GUID-Bezeichnern** In früheren Versionen von UltraLite hat die Laufzeitumgebung die Eingabe und Ausgabe von UUID- (Universally Unique Identifier) oder GUID-Bezeichnern (Globally Unique Identifier) als 16-Byte-Binärdatentyp oder als Zeichenfolgen zugelassen. Eine Endian-Konvertierung machte die Bezeichner kompatibel mit GUID-Strukturen. In UltraLite 11 können GUID-Strukturen explizit in der Laufzeitumgebung eingegeben bzw. ausgegeben werden und die Endian-Konvertierung ist nicht erforderlich.

- **ul_stream_error-Struktur** In UltraLite 11 wurden die Felder stream_id, stream_context und error_string_length entfernt. Außerdem wurde das Feld error_string von einem vom Benutzer bereitgestellten Datentyp char * in den statischen Datentyp char array geändert.

Plattformen und Geräte

- **Unterstützung des Deployments von nativen amd64/x64 ESQL- und C++-Anwendungen auf 64-Bit-Windows-Plattformen (64-Bit-Windows XP und höher)** UltraLite unterstützt nun das Deployment nativer amd64/x64 ESQL- und C++-Anwendungen auf 64-Bit-Windows-Plattformen (64-Bit-Windows XP und höher). Beachten Sie jedoch, dass Sie zur Entwicklung von UltraLite-Anwendungen auf einem 64-Bit-Computer die 32-Bit-Versionen der UltraLite-Dienstprogramme verwenden müssen. Zusätzlich müssen Sie die 32-Bit-Version von DBISQL und Sybase Central verwenden, wenn Sie eine Verbindung zu UltraLite auf einem 64-Bit-Computer herstellen.
- **UltraLite-Dienstprogramme auf Linux portiert (32-Bit)** Siehe [„Dienstprogramme“ auf Seite 273](#).

Sicherheit

- **Ende-zu-Ende-Verschlüsselung** UltraLite unterstützt nun die Ende-zu-Ende-Verschlüsselung auf Protokollebene. Daten werden mithilfe von 128-Bit-AES im CBC-Modus (Cipher Block Chaining) mit Schlüsselaustausch über RSA oder ECC verschlüsselt.

Dienstprogramme

- **UltraLite-Dienstprogramm zum Entladen von Datenbanken in XML (ulunload)** Sie können nun mit der Option -s das Schema entladen und Daten in einem SQL Anywhere-kompatiblen Format speichern. Siehe [„UltraLite-Dienstprogramm zum Entladen von Datenbanken \(ulunload\)“ \[UltraLite - Datenbankverwaltung\]](#).
- **UltraLite-Dienstprogramm zum Initialisieren einer Datenbank (ulinit)** Sie können nun mit der Option -d Daten aus einer SQL Anywhere-Datenbank in eine neue UltraLite-Datenbank kopieren. Siehe [„UltraLite-Dienstprogramm zum Initialisieren einer Datenbank \(ulinit\)“ \[UltraLite - Datenbankverwaltung\]](#).
- **ulerase** Dies ist ein neues Dienstprogramm zum Löschen einer UltraLite-Datenbank (einschließlich temporärer Dateien der Datenbank). Für dieses Dienstprogramm sind eine Benutzer-ID und ein Kennwort erforderlich, um den Zugriff auf die Datenbank zu bestätigen. Siehe [„UltraLite-Dienstprogramm zum Löschen von Datenbanken \(ulerase\)“ \[UltraLite - Datenbankverwaltung\]](#).
- **ulvalid** Dies ist ein neues Dienstprogramm zum Validieren einer UltraLite-Datenbank. Die Validierung prüft auf bestimmte Typen von Beschädigungen der Datenbankdatei. Sie ist anhand von Befehlszeilenparametern konfigurierbar. Siehe [„UltraLite-Dienstprogramm zum Validieren von Datenbanken \(ulvalid\)“ \[UltraLite - Datenbankverwaltung\]](#).
- **Folgende UltraLite-Dienstprogramme wurden auf Linux portiert (und sind nur in 32-Bit-Versionen verfügbar).**

- **ulcreate** Eine neue, leere UltraLite-Datenbank erstellen
- **ulerase** Eine UltraLite-Datenbank und eine zugehörige Checkpoint-Datei dauerhaft löschen
- **ulinfo** Information über eine vorhandene UltraLite-Datenbank anzeigen
- **ulinit** Eine neue UltraLite-Datenbank anhand des Schemas erstellen, das in einer SQL Anywhere-Referenzdatenbank verfügbar ist
- **ulload** Eine UltraLite-Datenbank aus den XML-Daten, die mithilfe von ulunload gespeichert wurden, erstellen und laden
- **ulsync** Eine UltraLite-Datenbank mit einer konsolidierten Datenbank synchronisieren, wobei MobiLink als Transfer-Agent verwendet wird
- **ulunload** Eine UltraLite-Datenbank in XML entladen
- **ulvalid** Gültigkeitsprüfungen für eine UltraLite-Datenbank ausführen

Das Dienstprogramm ulunloadold ist auf Linux nicht verfügbar.

SQL

- **Erweiterung der Anweisungen IF und CASE sowie des CASE-Ausdrucks** Für eine verbesserte Kompatibilität können IF-Ausdrücke nun mit ENDIF oder END IF abgeschlossen werden. CASE-Ausdrücke können nun mit END oder END CASE abgeschlossen werden. Siehe „IF-Ausdrücke“ [[UltraLite - Datenbankverwaltung](#)] und „CASE-Ausdrücke“ [[UltraLite - Datenbankverwaltung](#)].

Programmierschnittstellen

Allgemeine Verbesserungen

Publikationsmasken wurden durch Publikationslisten ersetzt. Mit dem Schlüsselwort Publications kann eine kommagetrennte Liste von Publikationsnamen angegeben werden.

UltraLite C/C++

Die neue Methode UltraLite_Table* OpenTableEx() ist nun Teil des UltraLite_Connection-Objekts. Diese Methode bietet Nicht-SQL-Anwendungen eine vielseitigere Möglichkeit, eine Tabelle zu öffnen und Zeilen direkt zu durchsuchen. Siehe „Direkte Page-Scans“ [[UltraLite - Datenbankverwaltung](#)].

Mit dieser Methode können Sie einen der folgenden Wege zum Öffnen einer Tabelle festlegen:

- Verwenden Sie ul_table_open_primary_key, um die Zeilen in der Reihenfolge des Primärschlüssels zurückzugeben
- Verwenden von ul_table_open_no_index, um die Zeilen in einer beliebigen Reihenfolge zurückzugeben

- Verwenden von `use ul_table_open_with_index`, um die Zeilen in der von einem Index festgelegten Reihenfolge zurückzugeben

UltraLite Embedded SQL

- Dies ist die Dokumentation für zwei Funktionen im Zusammenhang mit der Fehlerinterpretation. Siehe [ULGetErrorParameter-Methode \[UltraLite Embedded SQL\] \[UltraLite - C- und C++-Programmierung\]](#) und [ULGetErrorParameterCount-Methode \[UltraLite Embedded SQL\] \[UltraLite - C- und C++-Programmierung\]](#).
- Die Syntax der Funktionen `ULGetLastDownloadTime`, `ULResetLastDownloadTime` und `ULCountUploadRows` wurde geändert, um den Wechsel von Publikationsmasken zu Publikationslisten widerzuspiegeln.
- Die Funktion `ULGetPublicationMask` ist nicht mehr verfügbar.

UltraLite.NET

- Die Klasse `ULDataReader` umfasst die neue Methode `GetRowCount(threshold)` zum Abrufen einer Zeilenanzahl bis zu einer bestimmten Anzahl von Zeilen.
- Die Klasse `ULDataReader` implementiert nun die Schnittstelle `IListSource`.

UltraLite für M-Business Anywhere

- Neue Methoden in der Klasse `Connection` für die Ereignisverarbeitung und die Benachrichtigung: `cancelGetNotification`, `createNotificationQueue`, `declareEvent`, `destroyNotificationQueue`, `getNotification`, `getNotificationParameter`, `registerForEvent`, `sendNotification` und `triggerEvent`.

UltraLiteJ

UltraLiteJ ist eine Java-Implementierung von UltraLite, die Java SE- und Java ME-Umgebungen unterstützt, einschließlich BlackBerry-Smartphones. Siehe „[Erforderliche Systemausstattung und unterstützte Plattformen](#)“ [[UltraLite® – Java-Programmierung](#)].

UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen, die in UltraLite in Version 11.0.0 vorgenommen wurden.

Nicht mehr unterstützte Plattformen

- Die UltraLite C++-Schnittstelle unterstützt das Symbian-Betriebssystem nicht mehr. Entwickler von UltraLite-Anwendungen für Symbian sollten UltraLiteJ verwenden.
- Die UltraLite.NET-Schnittstelle unterstützt die .NET 1.0- Komponente nicht mehr. Die .NET 1.0-API-Referenz wurde aus der Dokumentation entfernt.

UltraLite für AppForge wird in Version 11 nicht unterstützt.

Datenbankeigenschaften

Folgende Datenbankeigenschaften werden in dieser Version nicht mehr unterstützt:

- CollationName

Verbindungsparameter

Die folgenden UltraLite-Verbindungsparameter werden in dieser Version nicht mehr unterstützt.

- ORDERED_TABLE_SCAN

Entfernte Dienstprogramme

Der **Assistent zum Migrieren von C++-Anwendungen** ist in Sybase Central nicht mehr verfügbar.

Entfernte, nicht mehr empfohlene und geänderte Funktionen

- Die Funktion ULGetPublicationMask der ESQ-LSchnittstelle wurde entfernt (Publikationsmasken wurden durch Publikationslisten ersetzt).
- Die Syntax der Funktionen ULGetLastDownloadTime, ULResetLastDownloadTime und ULCountUploadRows der ESQ-LSchnittstelle wurde geändert, um den Wechsel von Publikationsmasken zu Publikationslisten widerzuspiegeln.
- In der M-Business Anywhere-API werden folgende Komponenten nicht mehr empfohlen: DatabaseSchema.getTableCountInPublications, DatabaseSchema.SYNC_ALL_DB, DatabaseSchema.SYNC_ALL_PUBS, PublicationSchema.getMask, SyncParms.getPublicationMask und SyncParms.setPublicationMask.
- In der M-Business Anywhere- und der .NET API werden folgende Methoden verwendet, um eine Publikationsmaske als Parameter zu akzeptieren. Dieser Parameter wurde in eine Publikationsliste geändert (Zeichenfolge). Folgende Methoden in der Connection-Klasse sind davon betroffen: countUploadRows, getLastDownloadTime und resetLastDownloadTime.
- Die ul_sync_info-Struktur wurde geändert: Die Felder **disable_concurrency**, **checkpoint_store** und **table_order** wurden entfernt. Diese Optionen werden nun in einem neuen Feld namens **additional_parms** angegeben, das durch Semikola getrennte Schlüsselwort-Wert-Paare enthält.

Verschiedenes

- **Zweite Instanz von CustDB ausführen** In früheren Versionen führte der Aufruf einer zweiten Instanz einer CustDB-Anwendung zu einem Fehler. Wenn Sie nun eine zweite Instanz von CustDB starten, wird die erste Instanz in den Vordergrund gebracht und die zweite Instanz wird beendet.

Sybase Central und Interactive SQL

In den folgenden Abschnitten finden Sie Beschreibungen der neuen Funktionen und Verhaltensänderungen in Sybase Central und Interactive SQL für Version 11.0.0 und Informationen dazu, welche Funktionen nicht mehr empfohlen werden.

Neue Funktionen in Sybase Central und Interactive SQL

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Sybase Central und Interactive SQL in Version 11.0.0 hinzugefügt wurden.

- **Neue Schnelllader-Strategie** Bislang wurden bei der Anmeldung eines Benutzers die Interactive SQL- und Sybase Central-Schnelllader gestartet. Nun werden die Schnelllader nur gestartet, wenn Interactive SQL oder Sybase Central gestartet werden. Der Schnelllader wird noch bis zu 30 Minuten (konfigurierbar) lang ausgeführt, nachdem die Anwendung beendet wurde. Diese neue Strategie beschleunigt nachfolgende Aufrufe von Interactive SQL und Sybase Central innerhalb des 30-Minuten-Fensters. Siehe „Schnelllader-Option“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen und Änderungen des Fensters "Verbinden"** Folgende Erweiterungen wurden dem Fenster **Verbinden** in Sybase Central und Interactive SQL hinzugefügt:
 - **Neues Tool "Als ODBC-Datenquelle speichern"** Mit dem Tool **Als ODBC- Datenquelle speichern** können Sie eine ODBC-Datenquelle unter Verwendung der Verbindungsparameter generieren, die Sie im Fenster **Verbinden** angeben. Klicken Sie zur Verwendung dieses Tools im Fenster **Verbinden** auf die Schaltfläche **Extras**, klicken Sie auf **Als ODBC-Datenquelle speichern** und befolgen Sie dann die angezeigten Anweisungen. Siehe „[ODBC-Datenquellen erstellen \(Administrationstools\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - **Neues Tool "Verbindungszeichenfolge in Zwischenablage kopieren"** Mit dem Tool **Verbindungszeichenfolge in Zwischenablage kopieren** können Sie die Verbindungszeichenfolge in die Zwischenablage kopieren, um zu sehen, was an den Datenbankserver übergeben wird. Klicken Sie zur Verwendung dieses Tools im Fenster **Verbinden** auf die Schaltfläche **Extras** und klicken Sie dann auf **Verbindungszeichenfolge in Zwischenablage kopieren**. Fügen Sie die Verbindungszeichenfolge zur Anzeige in einen Texteditor ein.
 - **Verbindungsassistent** Der neue **Verbindungsassistent** ist nun in der rechten Hälfte des Fensters **Verbinden** verfügbar. Der **Verbindungsassistent** ist eine Assistentenfunktion mit deren Hilfe Sie eine Verbindung mit einer Datenbank herstellen können.
 - **Erweiterungen der Registerkarte "Erweitert"** Im Register **Erweitert** wird nun eine Tabelle mit Eigenschaften sowie eine kurze Beschreibung der Eigenschaften angezeigt.
 - **Neue Registerkarte "Netzwerk"** Auf der Registerkarte **Netzwerk** können Sie Optionen für die unterstützten Protokolle (Shared Memory und TCP/IP) festlegen. Diese neue Registerkarte ersetzt die Funktion der Option **Netzwerk nach Datenbankservern durchsuchen**, die bislang auf der Registerkarte **Datenbank** verfügbar war.
- **Tastenkürzel** Sie können nun im Code-Editor in Sybase Central und im Bereich **SQL-Anweisungen** in Interactive SQL mithilfe von Tastenkürzeln den Einzug des Codes und der Kommentare erhöhen oder verringern. Außerdem können Sie mithilfe von Tastenkürzeln sowohl den Doppel-Bindestrich als auch den Doppel-Schrägstrich als Kommentarindikator hinzufügen bzw. entfernen. Siehe „[Sybase Central-Tastenkürzel](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Neue Funktionen in Sybase Central

Im Folgenden finden Sie eine Liste der neuen Funktionen der Sybase Central-Plug-Ins in Version 11.0.0.

Neue Funktionen der SQL Anywhere-Plug-Ins

- **Register Übersicht** Sie können nun eine Überblicksansicht über den Status und die Statistiken des Datenbankservers und seiner Funktionen anzeigen. Siehe „Zustand und Statistik von Datenbanken“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbankdokumentation** Sie können nun mit dem **Assistenten zum Erstellen der Datenbankdokumentation** eine Dokumentation für Ihre Datenbank erstellen. Siehe „Datenbank dokumentieren“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue Assistenten** Das SQL Anywhere-Plug-In enthält nun folgende neue Assistenten:
 - **Assistent zum Erstellen einer Login-Richtlinie**
 - **Assistent zum Erstellen der Datenbankdokumentation**
 - **Assistent zum Erstellen von Textindizes**
 - **Assistent zum Erstellen eines Textkonfigurationsobjekts**
 - **Assistent zum Erstellen einer Datenbank**
- **Neue Eigenschaftsfenster** Das SQL Anywhere-Plug-In enthält nun Eigenschaftsfenster für folgende Funktionen:
 - Textindizes
 - Textkonfigurationsobjekte
 - Externe Umgebungen
 - Materialisierte Ansichten
 - Datenbank-Auditing
- **Auditing-Voreinstellungen in Sybase Central festlegen** Sie können nun mit Sybase Central Voreinstellungen für das Auditing von Datenbanken festlegen und Auditing-Daten anzeigen. Siehe „Auditing konfigurieren (Sybase Central)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Audit-Daten (Sybase Central)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung für Einstellungen zur Kollationsanpassung** Sie können im Fenster **Datenbankeigenschaften** im Register **Einstellungen** nun Optionen der Kollationsanpassung festlegen. Sie können auch beim Erstellen einer neuen Datenbank mithilfe des **Assistenten zum Erstellen einer Datenbank** Optionen der Kollationsanpassung festlegen.

Neue Funktionen des UltraLite-Plug-Ins

- **Neue Assistenten** Das UltraLite-Plug-In enthält nun folgende neue Assistenten:
 - **Assistent zum Validieren einer Datenbank**
 - **Assistent zum Erstellen eines Synchronisationsprofils**

- **Neue Eigenschaftsfenster** Das UltraLite-Plug-In enthält nun Eigenschaftsfenster für folgende Funktionen:
 - Synchronisationsprofile

Neue Funktionen des MobiLink-Plug-Ins

- **Unterstützung für Synchronisationsprofile** Sie können in Sybase Central Synchronisationsprofile erstellen und verwalten.
- **Neue Assistenten** Das MobiLink-Plug-In enthält nun folgende neue Assistenten:
 - Assistent zum Erstellen von Passthrough-Skripten
 - Assistent zum Erstellen eines Passthrough-Downloads
- **Neue Eigenschaftsfenster** Das MobiLink-Plug-In enthält nun Eigenschaftsfenster für folgende Funktionen:
 - Passthrough-Skripten
 - Passthrough-Downloads

Neue Funktionen von Interactive SQL

Im Folgenden finden Sie eine Liste der neuen Funktionen, die in Interactive SQL in Version 11.0.0 hinzugefügt wurden.

- **Datenbanksperren automatisch freigeben** Der Datenbankserver erstellt Schemasperren für Tabellen, die Sie in Interactive SQL anzeigen, auch wenn Sie die Tabelle nicht ändern.

Nun versucht Interactive SQL jedoch, die Datenbankschemasperren freizugeben, die bei der Anzeige Ihrer Ergebnismenge erstellt werden.

Nachdem Sie eine Anweisung mit darauf folgender Rückgabe einer Ergebnismenge ausgeführt haben, überprüft Interactive SQL, ob im Rahmen Ihrer Verbindung nicht festgeschriebene Änderungen in der Datenbank vorhanden sind. Wenn keine gefunden werden, gibt Interactive SQL Ihre Schemasperren frei; andernfalls bleiben die Schemasperren erhalten. Das heißt, Interactive SQL gibt Ihre Schemasperren nicht frei, wenn nicht festgeschriebene Änderungen in der Datenbank vorhanden sind.

- **Unterstützung für schreibgeschützte Ergebnismengen** Sie können nun für Ergebnismengen in Interactive SQL den Schreibschutz festlegen. Hierzu klicken Sie auf **Optionen** und klicken Sie dann auf die Option **Bearbeitung deaktivieren**. Diese Einstellung gilt für alle nachfolgend abgerufenen Ergebnisse. Siehe „[Zeilen in Interactive SQL-Ergebnismengen bearbeiten](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Beim Deployment von Interactive SQL können Sie verhindern, dass Benutzer diese Einstellung ändern, indem Sie dem Eintrag lockedPreferences in der Datei *OEM.ini* die Option disableResultSetEditing hinzufügen. Siehe „[Konfiguration der Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **SQL-Anweisungen einzeln nacheinander ausführen** Wenn Sie bislang SQL-Anweisungen einzeln ausführen wollten, mussten Sie die SQL-Anweisung wiederholt auswählen und auf **Markierte Anweisungen ausführen** klicken. Nun können Sie auf **Einzelschritt** klicken, um die betreffende Anweisung auszuführen und die nachfolgende Anweisung zur Ausführung auszuwählen. Ähnlich wie **Markierte Anweisungen ausführen** ist **Einzelschritt** im Menü **SQL** in Interactive SQL verfügbar. Siehe „[SQL-Anweisungen in Interactive SQL](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Favoriten für SQL-Datei und Datenbankverbindung** Sie können nun über das Menü **Favoriten** in Interactive SQL eine Liste von bevorzugten Datenbankverbindungen und eine Liste von bevorzugten SQL-Skriptdateien erstellen und verwalten. Siehe „[SQL-Skriptdateien, SQL-Anweisungen und Verbindungen zu den Favoriten hinzufügen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Tabellenbearbeitung von Ergebnismengen deaktivieren** Sie können nun beim Deployment von Interactive SQL die Tabellenbearbeitung von SQL Anywhere- und UltraLite-Ergebnismengen in Interactive SQL deaktivieren. Siehe „[Konfiguration der Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Neue Tastenkürzel** Interactive SQL wurden neue Tastenkürzel hinzugefügt. Siehe „[Tastenkürzel für Interactive SQL](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Unterstützung zum Verhindern von Optionsänderungen in Clientanwendungen** Sie können nun verhindern, dass Benutzer einige der Optionseinstellungen von Interactive SQL ändern, indem Sie die Einstellungen in der Datei *OEM.ini* sperren. Siehe „[Konfiguration der Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].
- **Neue Option -version für dbisql** Geben Sie an der Eingabeaufforderung `dbisql -version` ein, um die Versionsnummer von Interactive SQL anzuzeigen. Siehe „[Interactive SQL-Dienstprogramm \(dbisql\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[Interactive SQL-Dienstprogramm für UltraLite \(dbisql\)](#)“ [[UltraLite - Datenbankverwaltung](#)].

Verhaltensänderungen und nicht mehr empfohlene Funktionen von Sybase Central und Interactive SQL

Im Folgenden finden Sie eine Liste der Änderungen, die in Sybase Central und Interactive SQL in Version 11.0.0 vorgenommen wurden.

- **Einfacheres erneutes Deployment der Datenbanktool-Ladeprogramme** Das erneute Deployment der Ladeprogramme für Sybase Central, Interactive SQL und das Konsolendienstprogramm sowie des MobiLink-Monitors wurde vereinfacht. Registrierungseinträge und eine feste Verzeichnisstruktur für den Speicherort der JAR-Dateien sind nicht mehr erforderlich. Für jedes Programm muss eine entsprechende *Dateiname.INI*-Datei im gleichen (gleichnamigen) Verzeichnis wie die *Dateiname.exe*-Datei vorhanden sein. Die *.INI*-Datei enthält die Details zum Laden des Tools. Siehe „[Deployment von Administrationstools](#)“ [[SQL Anywhere Server - Programmierung](#)].

- **OEM.ini-Abschnitt [help] nicht mehr unterstützt** Der Abschnitt [help] in der Datei *OEM.ini* wird nicht mehr unterstützt. Weitere Hinweise finden Sie unter „[Konfiguration der Administrationstools](#)“ [*SQL Anywhere Server - Programmierung*].

Verhaltensänderungen von Sybase Central und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Sybase Central in Version 11.0.0.

- **Konfigurationsdatei von Sybase Central wurde umbenannt** Die Datei *.screpository* heißt nun *.screpository600*.

Geänderte Funktionen des SQL Anywhere-Plug-Ins

- **Erweiterungen der Eigenschaftsfenster** Folgende Eigenschaftsfenster wurden geändert:
 - Eigenschaftsfenster **Webdienst**
 - Eigenschaftsfenster **Benutzer**
 - Eigenschaftsfenster **Ansicht**
 - Eigenschaftsfenster **Materialisierte Ansicht**
- **Fehlersuche für spezifische Benutzer** Wenn Sie den **Debug-Modus**-Modus im SQL Anywhere-Plug-In starten, müssen Sie angeben, für welche Benutzer Sie die Fehlersuche durchführen möchten. Siehe „[Praktische Einführung: Erste Schritte mit dem Debugger](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
- **Geänderte Assistenten** Folgende Assistenten wurden aktualisiert:
 - Assistent zum Erstellen von Benutzern
 - Assistent zum Erstellen von Webdiensten
 - Deployment-Assistent

Geänderte Funktionen des MobiLink-Plug-Ins

- **Geänderte Assistenten** Folgende Assistenten wurden aktualisiert:
 - Assistent für das Deployment eines Synchronisationsmodells

Nicht mehr empfohlene Funktionen des SQL Anywhere-Plug-Ins

- **Entfernte Eigenschaftsregister** Folgende Eigenschaftsfenster wurden geändert:
 - Eigenschaftsfenster **Tabelle**
 - Eigenschaftsfenster **UltraLite-Projekt**
 - Eigenschaftsfenster **UltraLite-Anweisung**

Verhaltensänderungen von Interactive SQL und nicht mehr empfohlene Funktionen

Im Folgenden finden Sie eine Liste der Änderungen von Interactive SQL in Version 11.0.0.

- **Grafische Pläne nun in Plananzeige einsehbar** Sie können nun in Interactive SQL grafische Pläne für SQL Anywhere-Datenbanken in einem separaten Fenster namens "Plananzeige" anzeigen. Die Größe des Fensters kann verändert werden. Diese Änderung vereinfacht die Ansicht und den Vergleich von Plänen, da mehrere Plananzeige-Fenster gleichzeitig geöffnet werden können. Um auf die Plananzeige zuzugreifen, klicken Sie auf **Extras** → **Plananzeige**. Textpläne für UltraLite-Datenbanken werden ebenfalls in der Plananzeige angezeigt. Siehe „[Grafische Pläne mit detaillierten und Knotenstatistiken erstellen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Die Interactive SQL-Option isql_plan wird nicht mehr unterstützt.

- **Unterstützung der Anzeige von ausführlichen und kurzen Plänen wurde entfernt** Es ist nicht mehr möglich, Textpläne für SQL Anywhere-Datenbanken in Interactive SQL anzuzeigen. Sie können sie jedoch weiterhin mit den Funktionen EXPLANATION und PLAN abfragen. Sie können weiterhin Textpläne für UltraLite-Datenbanken in der Plananzeige in Interactive SQL anzeigen.
- **Ausführungspläne und Ergebnismengen drucken** Sie können nun den Inhalt des Fensterausschnitts **SQL-Anweisungen** und die Ergebnismengen drucken, indem Sie STRG+P drücken oder im Menü **Datei** auf die Option **Drucken** klicken. Bislang konnten Sie nur den Inhalt des Fensterausschnitts **SQL-Anweisungen** drucken. Sie können in der Plananzeige drucken, indem Sie auf die Schaltfläche **Drucken** klicken.
- **Fenster SQL-Anweisungen wurden Zeilenzahlen hinzugefügt** Auf der linken Seite des Fensterausschnitts **SQL-Anweisungen** werden nun Zeilenzahlen angezeigt. Diese Zeilenzahlen helfen Ihnen dabei, die Position von Syntaxfehlern zu identifizieren.
- **Erweiterung der Schaltfläche SQL-Anweisungen ausführen in der Symbolleiste** Bislang wurden beim Klick auf die Schaltfläche **SQL-Anweisungen ausführen** alle SQL-Anweisungen ausgeführt. Nun können Sie festlegen, ob alle Anweisungen oder nur die ausgewählten Anweisungen ausgeführt werden sollen.

Zum Festlegen des Verhaltens der Schaltfläche **SQL-Anweisungen ausführen** klicken Sie im Menü **Extras** auf **Optionen** → **Symbolleiste**. Siehe „[SQL-Anweisungen in Interactive SQL](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Erweiterung zur Ausführung von Batch-Anweisungen**
 - Interactive SQL stellt beim Ausführen von Batch-Anweisungen verbesserte Rückmeldungen bereit. Wenn SQL-Anweisungen über den Fensterausschnitt **SQL-Anweisungen** ausgeführt werden, wird die ausgeführte Anweisung nun ausgewählt und in die Ansicht abgerollt. Wenn Skriptdateien durch Klicken auf **Datei** → **Skript ausführen** ausgeführt werden, wird in einem Statusfenster der Verarbeitungsfortschritt des Skripts angezeigt. Siehe „[SQL-Anweisungen ausführen \(Interactive SQL\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Erweiterungen des Fensterausschnitts Ergebnisse**
 - Sie können nun im Fenster **Ergebnisse** alle Ergebnisse mit STRG+A auswählen. Sie können auch anstelle der aktuell abgerufenen Ergebnisse die gesamte Ergebnismenge auswählen. Wenn der Fensterausschnitt **Ergebnisse** nicht die gesamte Ergebnismenge enthält, werden Sie aufgefordert, die übrigen Ergebnisse abzurufen. Andernfalls werden nur die aktuell abgerufenen Ergebnisse ausgewählt.

- Wenn Sie nun Zellen aus dem Register **Ergebnisse** kopieren, werden die kopierten Daten basierend auf den folgenden Interactive SQL-Optionen formatiert: `isql_field_separator`, `isql_quote` und `isql_escape_character`. Sie können auch ausgewählte Werte, Zeilen und Spalten aus der Ergebnismenge in die Zwischenablage kopieren. Siehe „[Spalten, Zeilen und Zellen aus Interactive SQL-Ergebnismengen kopieren](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- Wenn Sie nun im Register **Ergebnisse** auf einen Spaltentitel klicken, werden die Ergebnisse anhand dieser Spalte sortiert. Wenn der Fensterausschnitt **Ergebnisse** nicht die gesamte Ergebnismenge enthält, werden Sie aufgefordert, die übrigen Ergebnisse abzurufen. Andernfalls werden nur die aktuell abgerufenen Ergebnisse sortiert.
- Sie können nun auf den ausgewählten Zeilen in der Ergebnismenge basierende INSERT-, DELETE- und UPDATE-Anweisungen generieren und in die Zwischenablage kopieren. Siehe „[SQL-Anweisungen aus Interactive SQL Ergebnismengen generieren](#)“ [*SQL Anywhere Server - Datenbankadministration*].
- Der Fensterausschnitt **Ergebnisse** in Interactive SQL wurde erweitert und verfügt nun im Kontextmenü über folgende Funktionen:

- **Kopieren&rarrZelle kopieren** Kopiert den Inhalt der ausgewählten Zelle.
- **Kopieren&rarrSpalte kopieren** Kopiert Zellwerte aus einer Spalte in die ausgewählte Zelle.
- **Generieren&rarrINSERT-Anweisung** Generiert für jede ausgewählte Zeile eine INSERT-Anweisung und kopiert die INSERT-Anweisungen in die Zwischenablage.
- **Generieren&rarrDELETE-Anweisung** Generiert für jede ausgewählte Zeile eine DELETE-Anweisung und kopiert die DELETE-Anweisungen in die Zwischenablage.
- **Generieren&rarrUPDATE-Anweisung** Generiert für jede ausgewählte Zeile eine UPDATE-Anweisung und kopiert die UPDATE-Anweisungen in die Zwischenablage. Die generierten Anweisungen setzen die Spaltenwerte auf ihre aktuellen Werte. Daher werden die Spaltenwerte durch die Ausführung der Anweisungen nicht verändert. Mit dieser Funktion können Sie eine Vorlage für eine UPDATE-Anweisung bereitstellen, die Sie vor ihrer Ausführung bearbeiten können.

Siehe „[Spalten, Zeilen und Zellen aus Interactive SQL-Ergebnismengen kopieren](#)“ [*SQL Anywhere Server - Datenbankadministration*] und „[SQL-Anweisungen aus Interactive SQL Ergebnismengen generieren](#)“ [*SQL Anywhere Server - Datenbankadministration*].

● Erweiterungen von Interactive SQL-Anweisungen

- **Erweiterungen der DESCRIBE-Anweisung** Die DESCRIBE-Anweisung gibt nun Informationen über die Datenbank oder den Datenbankserver zurück, der mit Interactive SQL verbunden ist. Siehe „[DESCRIBE-Anweisung \[Interactive SQL\]](#)“ [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- **Erweiterungen der Anweisungen INPUT und READ** Die Anweisungen INPUT und READ versuchen nun, relative Pfade auf zwei Weisen aufzulösen. Siehe „[INPUT-Anweisung \[Interactive](#)

SQL]“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und „**READ**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

○ **Erweiterungen der Anweisungen INPUT und OUTPUT**

- **Neue Unterstützung für Import aus und Export in ODBC-Quellen** Sie können nun beim Importieren in die Datenbank und beim Exportieren aus der Datenbank mit den Anweisungen INPUT bzw. OUTPUT eine ODBC-Datenquelle festlegen. Hierzu steht die neue USING-Klausel zur Verfügung. Siehe „**INPUT**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und „**OUTPUT**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

Sie können auch eine ODBC-Datenquelle festlegen, wenn Sie den **Import-Assistenten** und den **Export-Assistenten** verwenden. Siehe „**Importieren von Daten mit dem Import-Assistenten**“ [\[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#) und „**Exportieren von Daten mit dem Exportassistenten**“ [\[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#).

- **Neue Unterstützung der Bytereihenfolge-Markierung (BOM)** Sie können nun steuern, ob eine Bytereihenfolge-Markierung (Byte Order Mark, BOM) in Daten verarbeitet wird. Hierzu verwenden Sie die neue BYTE ORDER MARK-Klausel. Siehe „**INPUT**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#) und „**OUTPUT**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Unterstützte Formate für INPUT-Anweisung wurden geändert** Die INPUT-Anweisung unterstützt die Dateiformate dBase, Lotus, Excel und FoxPro nicht mehr. TEXT und FIXED werden weiterhin unterstützt. Wenn Sie diese Dateiformate weiterhin verwenden wollen, müssen Sie hierzu einen ODBC-Treiber verwenden. Siehe „**INPUT**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **Unterstützte Formate für OUTPUT-Anweisung wurden geändert** Die OUTPUT-Anweisung unterstützt die Dateiformate dBase, Lotus, Excel und FoxPro nicht mehr. TEXT, FIXED, HTML, SQL und XML werden weiterhin unterstützt. Siehe „**OUTPUT**-Anweisung [\[Interactive SQL\]“](#) [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- **ASCII-Format für das Dienstprogramm dbisqlc in INPUT- und OUTPUT-Anweisungen in TEXT umbenannt** Folgende Formate wurden in dieser Version bei der Verwendung der Anweisungen INPUT und OUTPUT mit dem Dienstprogramm dbisqlc umbenannt:

Formatname in früheren Versionen von SQL Anywhere	Neuer Formatname
ASCII	TEXT
TEXT	COLUMNS

- **Änderungen des Import-Assistenten und des Export-Assistenten** Wenn der **Import-Assistent** oder der **Export-Assistent** ausgeführt wurde, wird die vom Assistenten generierte SQL-Anweisung im Befehlsverlauf gespeichert. Zur Anzeige der generierten SQL-Anweisung klicken Sie auf **SQL&trarrVerlauf**.

- **Interactive SQL-Optionen**

- **isql_allow_read_client_file und isql_allow_write_client_file** Diese beiden Optionen beschreiben, wie Interactive SQL auf Anforderungen zum Lesen und Schreiben clientseitiger Dateien reagiert. Siehe „[isql_allow_read_client_file-Option \[Interactive SQL\]](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[isql_allow_write_client_file-Option \[Interactive SQL\]](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Option -codepage nicht mehr empfohlen** Wenn Sie wollen, dass Interactive SQL eine Datei mit einer bestimmten Codepage liest, verwenden Sie die ENCODING-Klausel der INPUT-, OUTPUT- oder READ-Anweisung. Siehe:
 - „[Interactive SQL-Dienstprogramm \(dbisql\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
 - „[Interactive SQL-Dienstprogramm für UltraLite \(dbisql\)](#)“ [[UltraLite - Datenbankverwaltung](#)]
 - „[INPUT-Anweisung \[Interactive SQL\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[OUTPUT-Anweisung \[Interactive SQL\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
 - „[READ-Anweisung \[Interactive SQL\]](#)“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- **isql_plan-Option nicht unterstützt** Die Interactive SQL-Option isql_plan wird nicht mehr unterstützt. Der Versuch, sie zu setzen, wird aus Gründen der Abwärtskompatibilität stillschweigend ignoriert. Siehe „[Grafische Pläne mit detaillierten und Knotenstatistiken erstellen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Schlüsselwort PUBLIC der SET OPTION-Anweisung wurde entfernt** Die Unterstützung des Schlüsselworts PUBLIC wurde für die Verwendung der Interactive SQL-Optionen mit der Anweisung SET OPTION entfernt. Siehe „[Interactive SQL-Optionen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Änderungen des Interactive SQL-Startprogramms** Das Programm für die Windows-Version des Interactive SQL-Startprogramms wurde von *dbisqlg.exe* in *dbisql.exe* geändert.

Das Programm für die Befehlszeilenversion des Interactive SQL-Startprogramms wurde von *dbisql.exe* in *dbisql.com* geändert. Batch-Skripten sollten *dbisql* oder *dbisql.com* anstelle von *dbisql.exe* aufrufen.

Verhaltensänderungen des SQL Anywhere-Konsolendienstprogramms

Die folgende Liste enthält die Änderungen des SQL Anywhere- Konsolendienstprogramms in Version 11.0.0.

- **Neue Konsolenoptionen** Sie können im Fenster **Optionen** nun ein Datum und eine Uhrzeit zum Herunterfahren des Datenbankservers festlegen. Klicken Sie im Menü **Datei** auf **Optionen** & Konsole.

Verhaltensänderungen des MobiLink-Monitors

Die folgende Liste enthält die Änderungen des MobiLink-Monitors in Version 11.0.0.

Der MobiLink-Monitor kann keine Monitordateien lesen, die mit Version 9 des MobiLink-Servers oder einer früheren Version erstellt wurden. Der MobiLink-Monitor sollte nur mit MobiLink-Servern der gleichen Version verwendet werden. Außerdem wurde die Worker-Spalte entfernt und die folgenden MobiLink-Eigenschaften wurden umbenannt:

Alter Eigenschaftsname	Neuer Eigenschaftsname
preload_upload	sync_request
verify_upload	authenticate_user

Erweiterungen der Dokumentation

- **Erweiterungen des Dokumentationsverzeichnisses** Bislang befand sich die Dokumentation unter `%SQLANY11%\docs`. Nun befindet sie sich unter `%SQLANY11%\documentation`. Die Dateinamen für die einzelnen HTML-Hilfdateien und die PDF-Dateien wurden ebenfalls aktualisiert.

Die neuen Dateinamen für die HTML-Hilfe lauten folgendermaßen:

- *dbadmin_de11.chm*
- *dbprogramming_de11.chm*
- *dbreference_de11.chm*
- *dbusage_de11.chm*
- *mlclient_de11.chm*
- *mlserver_de11.chm*
- *mlsisync_de11.chm*
- *mlstart_de11.chm*
- *sachanges_de11.chm*
- *saerrors_de11.chm*
- *saintro_de11.chm*
- *scplugin_de11.chm*
- *sqlanywhere_de11.chm*
- *sqlremote_de11.chm*
- *uladmin_de11.chm*
- *ulc_de11.chm*
- *uldotnet_de11.chm*
- *ulj_de11.chm*
- *ulmbus_de11.chm*

- **Auf unterstützte Plattformseiten wird nun über die Website zugegriffen** Bislang wurden die unterstützten Plattformseiten mit der Software installiert. Jetzt sind alle unterstützten Plattforminformationen auf der Sybase-Website <http://www.sybase.com/detail?id=1002288> verfügbar.

- **Suchen-Schaltfläche** Sie können nun im Browser der HTML-Hilfe mit der Schaltfläche **Suchen** feststellen, wo sich die aktuelle Hilfeseite im Inhaltsverzeichnis befindet.

Produktweite Funktionen

In den folgenden Abschnitten finden Sie Beschreibungen der neuen Funktionen und Verhaltensänderungen sowie nicht mehr empfohlenen Funktionen, die Auswirkungen auf alle Komponenten von SQL Anywhere Version 11.0.0 haben.

Produktweite neue Funktionen

Im Folgenden finden Sie eine Liste der produktweiten neuen Funktionen, die in Version 11.0.0 hinzugefügt wurden.

- **Erweiterungen der Fehler-Berichterstellung** Wenn unter Windows, Windows Mobile und Linux ein Fehlerbericht generiert wird, wird ein Fenster angezeigt, in dem Sie den Inhalt des Fehlerberichts einsehen können. Anschließend können Sie wählen, ob Sie den Fehler an iAnywhere senden wollen. Siehe „Fehlerbehandlung: Fehler in SQL Anywhere melden“ [[SQL Anywhere Server - Datenbankadministration](#)].

Sie können nun das Dienstprogramm für Supportanfragen (dbsupport) mit der Option -ce konfigurieren. Diese Option bewirkt, dass eine E-Mail gesendet wird, wenn dbsupport eine Anwendung überwacht und diese Anwendung abstürzt. Siehe „Dienstprogramm für Supportanfragen (dbsupport)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Zusätzliche Unterstützung für die Windows Mobile-Plattform** SQL Anywhere unterstützt nun Windows Mobile 5 für Smartphone und Windows Mobile 6. Weitere Informationen über die Ausführung von SQL Anywhere Server unter Windows Mobile finden Sie unter „SQL Anywhere für Windows Mobile“ [[SQL Anywhere Server - Datenbankadministration](#)] und „Hinweise zur Installation: Einschränkungen von Windows Mobile 5 und 6 für Smartphones“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Neue Tabelle in der SQL Anywhere-Beispieldatenbank (demo.db)** Der SQL Anywhere-Beispieldatenbank wurde die neue Tabelle "MarketinInformation" hinzugefügt. Jede Zeile in dieser Tabelle enthält eine HTML-Seite, die ein Produkt in der der Tabelle "Products" beschreibt. Diese Tabelle wurde hinzugefügt, um umfangreichere Zeichendaten für die Abfrage bereitzustellen, wenn Funktionen getestet werden. Siehe „SQL Anywhere-Beispieldatenbank“ [[SQL Anywhere 16 - Einführung](#)].

Produktweite Verhaltensänderungen

Im Folgenden finden Sie eine Liste der produktweiten Änderungen in Version 11.0.0.

- **Windows CE wurde in Windows Mobile geändert** Der Name Windows CE wurde in der Dokumentation und der Software in Windows Mobile geändert, außer an Stellen, an denen es präziser ist, weiterhin Windows CE zu verwenden.

- **Readcert, gencert und reqtool entfernt** Die Dienstprogramme readcert, gencert und reqtool wurden entfernt. Sie wurden bereits zuvor nicht mehr empfohlen. Stattdessen können Sie createcert und viewcert verwenden. Siehe „Zertifikat-Dienstprogramme“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Dienstprogramme createcert und viewcert unter Mac OS X unterstützt** Die Zertifikat-Dienstprogramme createcert und viewcert werden nun unter Mac OS X unterstützt. Siehe „Zertifikat-Dienstprogramme“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Protokolloptionen certificate und certificate_password umbenannt** Die TLS- und HTTPS-Protokolloptionen certificate und certificate_password wurden in identity bzw. identity_password umbenannt. Siehe:
 - SQL Anywhere-Datenbankserver. Siehe „Datenbankserveroption -ec “ [[SQL Anywhere Server - Datenbankadministration](#)].
 - SQL Anywhere-Webserver. Siehe „Datenbankserveroption -xs “ [[SQL Anywhere Server - Datenbankadministration](#)].
 - SQL Anywhere-Protokolloptionen. Siehe „Identity-Protokolloption“ [[SQL Anywhere Server - Datenbankadministration](#)] und „identity_password-Protokolloption“ [[SQL Anywhere Server - Datenbankadministration](#)].
 - MobiLink-Server. Siehe „mlsrv16-Option -x“ [[MobiLink - Serveradministration](#)].
- **Änderungen der Beispiel-Identitätsdatei** Die Identitätsdateien mit den Beispielzertifikaten und entsprechenden privaten Schlüsseln für TLS wurden in dieser Version umbenannt. Die Datei *rsaserver.crt* wurde in *rsaserver.id* umbenannt, und die Datei *sample.crt* wurde in *eccserver.id* umbenannt. Das Kennwort für beide Identitätsdateien wurden von **tJ1#m6+W** in **test** geändert.
- **Änderungen der Installationsverzeichnisse** Die 32-Bit-Software wird nun im Verzeichnis *bin32* anstatt im Verzeichnis *win32* installiert. Die 64-Bit-Software wird im Verzeichnis *bin64* anstatt im Verzeichnis *X64* installiert. Software, die in früheren Versionen unter *C:\Programme\SQL Anywhere 11\win32* installiert wurde, wird somit nun unter *C:\Programme\SQL Anywhere 11\bin32* installiert.
- **Änderungen der ODBC-Datenquellen für die Beispieldatenbanken** In früheren Versionen handelte es sich bei den ODBC-Datenquellen für die Beispieldatenbanken, die zusammen mit der Software installiert wurden, um Benutzerdatenquellen. Die Datenquellen SQL Anywhere 11 und SQL Anywhere 11 CustDB sind nun Systemdatenquellen.
- **.NET 1.0 nicht unterstützt** SQL Anywhere 11 unterstützt Microsoft Visual Studio .NET 2002 und Visual Studio .NET 2003 nicht. Microsoft Visual Studio 2005 (.NET 2.0) und Visual Studio 2008 (.NET 3.x) werden dagegen unterstützt.

Upgrade auf SQL Anywhere 16

Unterschiede zwischen Hauptversionen, Minor Releases und Supportpaketen

- **Supportpaket** Ein Supportpaket ist eine Teilmenge der Software mit einer oder mehreren Fehlerbehebungen. Die Fehlerbehebungen werden in den Versionshinweisen für das Update aufgelistet. Updates zu Fehlerbehebungen können nur in eine installierte Software mit derselben Versionsnummer übernommen werden. Obwohl einige Tests der Software vorgenommen wurden, sollten Sie diese Dateien nicht mit Ihrer Anwendung verteilen, es sei denn, Sie haben Ihre Anwendung gründlich mit der Software getestet.
- **Minor Releases** Eine Minor Release ist eine vollständige Software, mit der ein Upgrade der installierten Software von einer älteren Version mit derselben Hauptversionsnummer durchgeführt wird. (Das Versionsnummernformat lautet *major.minor.build*). Fehlerbehebungen und andere Änderungen werden in den Versionshinweisen für das Upgrade aufgelistet.
- **Hauptversionen** Eine Hauptversion ist eine vollständige Software, die eine eigene Versionsnummer hat.

Kompatibilität mit vorhandenen Datenbanken, Servern und Administrationstools

Hinweise zu folgenden Aspekten finden Sie auf der Seite mit den [Komponenten nach Plattform](#):

- Unterstützung für Datenbank- und Serververbindungen mit Datenbanken, Servern und Clientanwendungen, die Software aus früheren Versionen verwenden
- Administrationstool-Unterstützung für Verbindungen mit Datenbanken Server und Clientanwendungen, die Software aus früheren Versionen verwenden

Siehe auch

- „Dienstprogramm zum Entladen (dbunload)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind“ auf Seite 298

Funktionen, die einen Neuaufbau (Entladen/Neuladen), ein Datenbank-Upgrade oder aktualisierte Clientbibliotheken erfordern

Obwohl viele Funktionen zum Ausführen einer älteren Datenbank unter der neuesten Version des Datenbankservers verfügbar sind, müssen Sie für den Zugriff auf einige Funktionen Ihre Datenbank neu aufbauen (entladen und neu laden), ein Upgrade der Datenbankdatei durchführen oder Ihre Clientbibliotheken aktualisieren.

Hinweis

Für Funktionen, die nicht in den Kategorien unten aufgelistet sind, ist nur ein Datenbankserver der Version SQL Anywhere 16.0 erforderlich. In diesem Fall müssen Sie weder ein Upgrade vorhandener Datenbanken noch eine Aktualisierung der Clientbibliotheken durchführen. Eine vollständige Liste der neuen Funktionen in SQL Anywhere 16 finden Sie unter [„Neue Funktionen in Version 16.0“ auf Seite 1](#).

Unterschiede zwischen einer neu aufgebauten (Entladen/Neuladen) Datenbank und einer Datenbank nach Upgrade

- **Datenbanken nach Upgrade** Eine Datenbank nach Upgrade ist eine Datenbank, in der einige Funktionen aktualisiert wurden. Im Rahmen eines Upgrades werden beispielsweise die Systemtabellen und -ansichten aktualisiert, neue Datenbankoptionen hinzugefügt, alle gespeicherten Systemprozeduren neu erstellt, die jConnect-Unterstützung installiert und die Unterstützung für Java in der Datenbank geändert. Sie können ein Upgrade einer Datenbank entweder mit dem Dienstprogramm zum Upgrade (dbupgrad) durchführen oder mit dem **Assistenten zum Upgrade einer Datenbank** in Sybase Central.
- **Neu aufgebaute Datenbanken** Eine neu aufgebaute Datenbank ist eine neue Datenbank, die die Daten aus einer älteren Datenbank enthält. Für den Neuaufbau einer Datenbank können Sie entweder die UNLOAD-Anweisung oder das Dienstprogramm zum Entladen (dbunload) mit der Option -an verwenden.

Funktionen, die einen Neuaufbau (Entladen/Neuladen) der Datenbank erfordern

- Die UltraLite-Version 16 kann keine Verbindungen mit Datenbanken herstellen, die mit früheren Versionen von UltraLite erstellt wurden. Deshalb ist ein Entladen/Neuladen erforderlich. Clients der UltraLite-Version 11 oder höher können jedoch mit MobiLink-Version 16 synchronisiert werden.
- Die ADD COLUMN-Klausel der ALTER TABLE-Anweisung (Datenbanken der Version 10 und früher). Siehe [„ALTER TABLE-Anweisung“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- CESU-8-Zeichensatzunterstützung. Weitere Hinweise zu unterstützten Kollationen finden Sie unter [„Alternative Kollationen“ \[SQL Anywhere Server - Datenbankadministration\]](#).

Funktionen, die nur ein Upgrade erfordern (dbupgrad-Dienstprogramm oder UPGRADE DATABASE-Anweisung)

Funktionen, die nur ein Datenbank-Upgrade erfordern, können auch ausgeführt werden, wenn Sie die Datenbank entladen/neu laden.

- Neue Fehlercodes und geänderte Rückgabecodes für SMTP und MAPI. Siehe [Rückgabecodes für MAPI- und SMTP-Systemprozeduren \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- ADD COLUMN-Klausel der ALTER TABLE-Anweisung (Datenbanken der Version 11 und höher). Siehe [„ALTER TABLE-Anweisung“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).
- CREATE INDEX-Anweisung. Siehe [„CREATE INDEX-Anweisung“ \[SQL Anywhere Server - SQL-Referenzhandbuch\]](#).

- Rollen und Privilegien, einschließlich der Änderungen an Sybase Central zur Unterstützung des neuen rollenbasierten Sicherheitsmodells. Siehe [Neues Sicherheitsmodell: RBAC \(Role-Based Access Control - rollenbasierte Zugriffssteuerung\)](#) auf Seite 1.
- Einige Systemprozeduren werden nun mit den Privilegien des Aufrufers ausgeführt statt mit denjenigen des Definierers (Eigentümers). Siehe [„Prozeduren und Funktionen zum Ausführen mit Eigentümer- oder Aufruferprivilegien einrichten“](#) [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
- Unterstützung der LDAP-Benutzerauthentifizierung. Siehe [„LDAP-Benutzerauthentifizierung“](#) [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
- Unterstützung der Datentypen ROW und ARRAY. Siehe [„Zusammengesetzte Datentypen“](#) [*SQL Anywhere Server - SQL-Referenzhandbuch*].
- OData-Unterstützung. Siehe [„OData-Unterstützung“](#) [*SQL Anywhere Server - Programmierung*].
- Reservierte Wörter:
 - array
 - json
 - row
 - rowtype
 - unnest
 - varray
- Login-Richtlinienoptionen:
 - ldap_primary_server
 - ldap_secondary_server
 - ldap_auto_failback_period
 - ldap_failover_to_std
 - ldap_refresh_dn
 - change_password_dual_control

Siehe [„Root-Login-Richtlinie“](#) [*SQL Anywhere Server - Datenbankadministration*].

- Datenbankoptionen:
 - „auto_commit_on_create_local_temp_index-Option“ [*SQL Anywhere Server - Datenbankadministration*]
 - „disk_sandbox-Option“ [*SQL Anywhere Server - Datenbankadministration*]
 - „Datenbankoption -sbx “ [*SQL Anywhere Server - Datenbankadministration*]
 - „min_role_admins-Option“ [*SQL Anywhere Server - Datenbankadministration*]
 - „db_publisher-Option“ [*SQL Anywhere Server - Datenbankadministration*]
 - „login_mode-Option“ [*SQL Anywhere Server - Datenbankadministration*]
 - „trusted_certificates_file-Option“ [*SQL Anywhere Server - Datenbankadministration*]
 - „Datenbankoption -al “ [*SQL Anywhere Server - Datenbankadministration*]

- Datenbankserveroptionen:
 - „Datenbankserveroption -gu “ [\[SQL Anywhere Server - Datenbankadministration\]](#)
 - „Datenbankserveroption -gk “ [\[SQL Anywhere Server - Datenbankadministration\]](#)
 - „Datenbankserveroption -gd “ [\[SQL Anywhere Server - Datenbankadministration\]](#)
 - „Datenbankserveroption -gl “ [\[SQL Anywhere Server - Datenbankadministration\]](#)
 - „Datenbankserveroption -al “ [\[SQL Anywhere Server - Datenbankadministration\]](#)
- Verbindungseigenschaften:
 - `auto_commit_on_create_local_temp_index`-Verbindungseigenschaft [\[SQL Anywhere Server - Datenbankadministration\]](#)
 - `extern_login_credentials`-Verbindungseigenschaft [\[SQL Anywhere Server - Datenbankadministration\]](#)
- Systemansichten:
 - „SYSGROUP-Kompatibilitätsansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSGROUPS-Kompatibilitätsansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSLDAPSERVER-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSROLEGRANT-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „Konsolidierte Ansicht SYSROLEGRANTS“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSROLEGRANTTEXT-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSUSERAUTHORITY-Kompatibilitätsansicht (nicht mehr empfohlen)“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSTABLEPERM-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „Konsolidierte Ansicht SYSTABAUTH“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSCERTIFICATE-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSTABCOL-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „SYSUSER-Systemansicht“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
- Systemtabellen:
 - „ISYSCERTIFICATE-Systemtabelle“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „ISYSTABCOL-Systemtabelle“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „ISYSTAB-Systemtabelle“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
 - „ISYSUSER-Systemtabelle“ [\[SQL Anywhere Server - SQL-Referenzhandbuch\]](#)
- Zeitstempel werden nun in Systemtabellen als UTC-basiertes Äquivalent (Coordinated Universal Time - koordinierte Weltzeit) gespeichert, aber in Systemansichten werden sowohl UTC-Zeitstempel als auch

Ortszeitstempel bereitgestellt. Anwendungen, die auf der Anzahl der Spalten in den folgenden Systemansichten bzw. deren Basistabellen basieren, müssen aktualisiert werden:

- „SYSTAB-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSUSER-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSVIEW-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSOBJECT-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSTEXTIDX-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSEXTERNENVOBJECT-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSJAR-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSJAVACLASS-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSHISTORY-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSREMOTEUSER-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „Konsolidierte Ansicht SYSREMOTEUSERS“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „SYSCOLSTAT-Systemansicht“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „Konsolidierte Ansicht SYSCOLSTATS“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- Funktionen:
 - „ARRAY-Konstruktor [zusammengesetzt]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „ROW-Konstruktor [zusammengesetzt]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „ENCRYPT-Funktion [Zeichenfolge]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - „DECRYPT-Funktion [Zeichenfolge]“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
 - Operatoren:
 - „Array-Operator UNNEST“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]

- Systemprozeduren:

- „sa_certificate_info-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_cpu_topology-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_db_option-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_get_ldapsrvr_status-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_get_user_status-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_parse_json-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_server_option-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_copy_directory-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_copy_file-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_create_directory-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_delete_directory-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_delete_file-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_move_directory-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_move_file-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_create_secure_feature_key-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_alter_secure_feature_key-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_drop_secure_feature_key-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_list_secure_feature_keys-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_use_secure_feature_key-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_get_mail_error_code-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_get_mail_error_text-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_getenv-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_startmail-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_startsmtp-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_sendmail-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_stopmail-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „xp_stopsmtp-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_objectpermission-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_displayroles-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_has_role-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_proc_priv-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_auth_sys_role_info-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sp_sys_priv_role_info-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_get_ldapsrvr_status-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]
- „sa_get_user_status-Systemprozedur“ [*SQL Anywhere Server - SQL-Referenzhandbuch*]

- Neue oder erweiterte SQL-Anweisungen:

- „GRANT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „REVOKE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE ROLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER ROLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP ROLE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „GRANT ROLE SYS_REPLICATION_ADMIN_ROLE-Anweisung [MobiLink] [SQL Remote]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „GRANT ROLE SYS_RUN_REPLICATION_ROLE-Anweisung [MobiLink] [SQL Remote]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „GRANT PUBLISH-Anweisung [SQL Remote]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „REVOKE PUBLISH-Anweisung [SQL Remote]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „VALIDATE LDAP SERVER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER USER-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER LOGIN POLICY-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „COMMENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „SET MIRROR OPTION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE CERTIFICATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP CERTIFICATE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE FUNCTION-Anweisung [Webdienst]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE PROCEDURE-Anweisung [Webdienste]“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE TEXT INDEX-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „CREATE TEMPORARY TRACE EVENT SESSION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „ALTER TRACE EVENT SESSION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP TRACE EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „DROP TRACE EVENT SESSION-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „NOTIFY TRACE EVENT-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]

Siehe auch

- „Neue Funktionen in Version 16.0“ auf Seite 1

SQL Anywhere-Server-Ugrades

Bevor Sie vorhandene Anwendungen mit dieser Version der Software verwenden, sollten Sie die Liste der Verhaltensänderungen überprüfen, um festzustellen, ob Auswirkungen auf Ihre Anwendung zu erwarten sind. Siehe [SQL Anywhere 16 - Änderungen und Upgrades auf Seite 1](#).

Upgrade von Datenbanken der Version 10 und höher

Falls Sie ein Upgrade von Version 10 oder höher durchführen, können Sie ein Datenbank-Upgrade durchführen bzw. die Datenbank neu aufbauen. Das Upgrade bzw. der Neuaufbau ist ein optionaler Schritt, weil die Software der Version 16 zusammen mit einer Datenbank der Version 10 oder höher verwendet werden kann. Wenn Sie jedoch alle neuen Funktionen in Version 16 nutzen möchten, müssen Sie die Datenbank neu aufbauen. Siehe [„Upgradevorgang für Datenbanken der Version 10 und später“ auf Seite 310](#) und [„Neuaufbauprozess für Datenbanken der Version 10 und höher“ auf Seite 300](#).

Hinweis

Es empfiehlt sich, die materialisierten Ansichten in der Datenbank zu aktualisieren, nachdem Sie ein Upgrade des Datenbankservers durchgeführt oder die Datenbank neu aufgebaut bzw. umgestellt haben, um mit einem Datenbankserver zu arbeiten, für den ein Upgrade durchgeführt wurde. Siehe [„Materialisierten Ansichten manuell aktualisieren“ \[SQL Anywhere Server - SQL-Benutzerhandbuch\]](#).

Upgrade von Datenbanken bis Version 9

Wenn Sie von Version 9 oder früher auf Version 16 umstellen, müssen Sie die Datenbank neu aufbauen. Hierzu muss die alte Datenbank entladen und in eine neue Datenbank der Version 16 geladen werden. Der Versuch, Datenbanken der Version 9 oder früher zu starten, führt zu einem Fehler. Es gibt mehrere Möglichkeiten, vorhandene Datenbanken neu aufzubauen:

- Sie verwenden das Dienstprogramm Entladen (dbunload) der Version 16 mit der Option -an (Erstellen einer neuen Datenbank) oder der Option -ar (Ersetzen der alten Datenbank). Siehe [Datenbanken der Version 9 oder früher mit dem Dienstprogramm zum Entladen neu aufbauen \(Befehlszeile\) auf Seite 309](#).

Hinweis

Das Dienstprogramm Entladen (dbunload) hat in allen Versionen von SQL Anywhere denselben Dateinamen. Vergewissern Sie sich, dass Sie die richtige Version verwenden. Führen Sie den Befehl `dbunload -?` aus, um zu ermitteln, welche Version des Dienstprogramms Entladen Sie verwenden. Siehe [„Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind“ auf Seite 298](#).

- Sie entladen die Datenbank mit dem dbunload-Dienstprogramm der Version 16 und laden die Datenbank dann neu mit der Datei *reload.sql* auf dem Datenbankserver der Version 16.

Wenn Sie Schemaänderungen durchführen müssen, wird dieser Weg für das Upgrade empfohlen. Nachdem Sie die Schemaänderungen vorgenommen haben, können Sie eine Datenbank erstellen und anschließend das Reload-Skript darauf anwenden.

- Verwenden Sie den **Assistenten zum Entladen einer Datenbank** in Sybase Central. Sie können wählen, eine neue Datenbank zu erstellen, eine vorhandene Datenbank durch die neue Datenbank zu ersetzen oder die Datenbank in eine Datei zu entladen. Siehe [Datenbanken der Version 9 oder früher neu aufbauen \(Sybase Central\) auf Seite 308](#).
- Sie entladen die Datenbank mit einer älteren Version von dbunload und laden die Datenbank dann neu mit der Datei *reload.sql* und dem Datenbankserver der Version 16. Wählen Sie diese Vorgehensweise nur dann, wenn die anderen Methoden fehlschlagen, weil möglicherweise nicht mehr empfohlene oder nicht unterstützte Datenbankoptionseinstellungen, Objekte oder SQL-Syntax in die Datei *reload.sql*

entladen werden. Falls beim Neuladen Probleme auftreten, bearbeiten Sie die Datei manuell. Die internen Neuladefunktionen von Version 16 beseitigen viele dieser Probleme.

Hinweis

SQL Anywhere 9.0.2 für Mac OS X wurde auf PPC unterstützt, während SQL Anywhere 10 und höher für Mac OS X auf Intel unterstützt wird. Wenn Sie eine Datenbank der Version 9.0.2 oder früher unter Mac OS X betreiben, gibt es zwei Möglichkeiten zum Entladen der Datenbank:

- Sie entladen die Datenbank mit der Version 9.0.2.
- Sie kopieren die Datenbank auf eine andere Plattform, auf der SQL Anywhere 16 installiert ist, und entladen die Datenbank dann mithilfe der Version 16.

Wenn die Datenbank entladen wurde, können Sie sie mit der Version 16 auf Mac OS X neu laden.

Das Verfahren ist komplizierter, wenn Sie die Merkmale der Datenbank während des Entladens und Neuladens ändern möchten (z.B. eine Datenbank so einstellen, dass sie nicht mehr die Groß- und Kleinschreibung berücksichtigt). Siehe „[Neuaufbau von Datenbanken](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].

Vorsichtsmaßnahmen für Upgrade und Neuaufbau

Es gibt mehrere Vorsichtsmaßnahmen, die Sie vor einem Upgrade von SQL Anywhere ergreifen sollten:

- **Überprüfung der Verhaltensänderungen** Überprüfen Sie, ob die dokumentierten Verhaltensänderungen Ihre Anwendung betreffen. Wenn dies der Fall ist, aktualisieren Sie Ihre Anwendung. Siehe „[Neue Funktionen in Version 16.0](#)“ auf Seite 1.
- **Testen der Anwendung** Testen Sie die Anwendung gründlich in einer SQL Anywhere 16-Umgebung, bevor Sie das Upgrade bei Anwendungen in einer Produktionsumgebung durchführen.
- **Verwenden der richtigen Version der Dienstprogramme** Stellen Sie sicher, dass Sie für die neue Datenbank die richtige Version der Datenbank-Dienstprogramme verwenden. Siehe „[Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind](#)“ auf Seite 298.
- **Validieren und Sichern der Datenbank** Bevor Sie ein Upgrade beginnen, validieren Sie Ihre Datenbank und sichern Sie Ihre Software und Datenbank. Um die zukünftige Wiederherstellbarkeit sicherzustellen, führen Sie eine Sicherung der Datenbank durch, wenn Sie mit dem Upgrade fertig sind.
- **Synchronisieren vor dem Upgrade** Für Datenbanken, die in eine Synchronisationsumgebung eingebunden sind, z.B. UltraLite-Datenbanken oder entfernte SQL Anywhere-Datenbanken in MobiLink-Installationen, müssen Sie vor dem Upgrade eine erfolgreiche Synchronisation durchführen.
- **Testen der Upgrade-Prozedur** Testen Sie die Upgrade-Prozedur sorgfältig, bevor Sie sie in einem Produktionssystem ausführen.

SQL Anywhere wird in zahlreichen verschiedenen Konfigurationen verwendet. Es ist daher nicht möglich, für alle Fälle zutreffende Upgrade-Richtlinien zu geben.

Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind

Wenn mehrere Versionen von SAP Sybase SQL Anywhere auf Ihrem Windows-Computer installiert sind, müssen Sie bei der Verwendung von Dienstprogrammen auf den Systempfad achten. Da bei der Installation das Verzeichnis mit der Programmdatei der zuletzt installierten Version am Ende Ihres Systempfads hinzugefügt wird, ist es möglich, eine neue Version der Software zu installieren und dennoch unbeabsichtigt mit der früheren Version zu arbeiten.

Wenn beispielsweise ein Verzeichnis mit der Programmdatei von Version 8 im Pfad vor dem Verzeichnis mit der Programmdatei von Version 16 steht und Sie den dbinit-Befehl verwenden, arbeiten Sie mit dem Dienstprogramm der Version 8 und erstellen daher eine Datenbank der Version 8.

Es gibt unter anderem folgende Möglichkeiten sicherzustellen, dass Sie die Dienstprogramme der Version 16 verwenden:

- Ändern Sie Ihren Systempfad so, dass das Verzeichnis mit der Programmdatei von SQL Anywhere 16 vor allen Verzeichnissen mit Programmdateien früherer Versionen steht.
- Wechseln Sie in das Verzeichnis mit der Programmdatei von SQL Anywhere 16, bevor Sie den betreffenden Befehl ausführen.
- Sie geben einen voll qualifizierten Pfadnamen für das Dienstprogramm an, der den genauen Speicherort des Dienstprogramms enthält, das ausgeführt werden soll.
- Sie erstellen Skripten, um die Umgebung so zu ändern, dass sie die richtige Version der Dienstprogramme verwendet.
- Sie deinstallieren die alte Software.

Datenbanken neu aufbauen (entladen/neu laden)

Für Benutzer von früheren Versionen der Software fasst die folgende Aufgabe den Prozess zum Neuaufbau der Datenbank in Version 16 zusammen.

Voraussetzungen

Für Datenbanken der Version 16 müssen Sie die folgenden Privilegien haben:

- BACKUP DATABASE

- VALIDATE ANY OBJECT
- SERVER OPERATOR
- SELECT ANY TABLE

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Sichern Sie die Datenbank. Beispiel:

```
dbbackup -c "DBF=mydb.db;UID=DBA;PWD=sql" old-db-backup-dir
```

Datenbanken neu aufbauen (Befehlszeile)

1. Defragmentieren Sie, wenn möglich, das Laufwerk, auf dem die neue Datenbank gespeichert werden soll, da ein fragmentiertes Laufwerk die Performance der Datenbank beeinträchtigen kann.
2. Fahren Sie alle SQL Anywhere-Datenbankserver herunter, da das dbunload-Dienstprogramm der Version 16 nicht für eine Datenbank verwendet werden kann, die auf einer früheren Version des Datenbankservers läuft. Beispiel:

```
dbstop -c "DBF=mydb.db;UID=DBA;PWD=sql"
```

3. Entladen Sie die alte Datenbank und laden Sie sie (Neuaufbau) in eine neue Datenbank der Version 16. Beispiel:

```
dbunload -c "DBF=mydb.db;UID=DBA;PWD=sql" -an mydb16.db
```

4. Sichern Sie die neue Datenbank, bevor Sie sie verwenden. Beispiel:

```
dbbackup -c "DBF=mydb16.db;UID=DBA;PWD=sql" new-db-backup-dir
```

5. Validieren Sie die neue Datenbank vor ihrer Verwendung. Beispiel:

```
dbvalid -c "DBF=mydb16.db;UID=DBA;PWD=sql"
```

Ergebnisse

Die Datenbank wird in der neuesten Version neu aufgebaut. Standardmäßig wird die Datenbank gestoppt und neu gestartet.

Nächste Schritte

Sie können die neu aufgebaute Datenbank mit Ihrer Anwendung testen.

Siehe auch

- [Datenbanken der Version 9 oder früher mit dem Dienstprogramm zum Entladen neu aufbauen \(Befehlszeile\) auf Seite 309](#)
- [Datenbanken der Version 9 oder früher neu aufbauen \(Sybase Central\) auf Seite 308](#)
- [„Upgrade von authentifizierten Datenbanken“ \[SQL Anywhere Server - Datenbankadministration\]](#)

Neuaufbauprozess für Datenbanken der Version 10 und höher

Um eine Datenbank neu aufzubauen, muss sie entladen und neu geladen werden, damit ihr Dateiformat aktualisiert wird. Wenn Sie das Dateiformat aktualisieren, wird das Format geändert, in dem Daten auf die Festplatte geschrieben und von ihr gelesen werden. Dadurch können Sie alle neuen Funktionen und Performance-Erweiterungen der letzten Softwareversion verwenden.

Vorsicht

Das Entladen und Neuladen einer großen Datenbank kann zeitaufwändig sein und viel Plattenspeicher benötigen. Der für den Vorgang erforderliche Plattenspeicher entspricht etwa der doppelten Datenbankgröße, um die entladenen und neu geladenen Daten zu speichern.

Aufgrund von Indexänderungen in SQL Anywhere kann es vorkommen, dass eine Datenbank nach dem Neuaufbau durch Entladen und Neuladen kleiner ist als die ursprüngliche Datenbank. Die geringere Datenbankgröße ist kein Hinweis auf ein Problem oder einen Datenverlust.

Hinweis

Es wird empfohlen, die Datenbank vor dem Neuaufbau zu sichern.

Neuladen von Tabellen mit AUTOINCREMENT-Spalten

Sie können den nächsten verfügbaren Wert für AUTOINCREMENT-Spalten in der neu aufgebauten Datenbank abfragen, indem Sie die dbunload-Option -l angeben. Diese Option fügt dem generierten Skript *reload.sql* Aufrufe der sa_reset_identity-Systemprozedur für jede Tabelle hinzu, die einen AUTOINCREMENT-Wert enthält, sodass der aktuelle Wert von SYSTABCOL.max_identity erhalten bleibt.

Datenbanken neu aufbauen (Sybase Central)

Sie können den **Assistenten zum Entladen einer Datenbank** verwenden, um ein Upgrade einer SQL Anywhere-Datenbank der Version 10 oder höher auf die neueste Version durchzuführen.

Voraussetzungen

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe [„Vorsichtsmaßnahmen für Upgrade und Neuaufbau“ auf Seite 297](#).

Sie müssen die Systemprivilegien SELECT ANY TABLE und SERVER OPERATOR haben.

Achten Sie darauf, die Datenbank zu sichern, bevor Sie sie entladen und neu laden. Beispiel:

```
dbbackup -c "DBF=mydb.db;UID=DBA;PWD=sql" old-db-backup-dir
```

Datenbanken neu aufbauen (Sybase Central)

1. Klicken Sie auf **Start→Programme→SQL Anywhere 16→Administrationtools→Sybase Central**.

2. Starten Sie einen Datenbankserver der Version 16 mit der Datenbank, für die Sie das Upgrade durchführen wollen, und stellen Sie dann von Sybase Central aus eine Verbindung mit der Datenbank her.
3. Klicken Sie auf **Extras** → **SQL Anywhere 16** → **Datenbank entladen**.
4. Lesen Sie die erste Seite des **Assistenten zum Entladen einer Datenbank** und klicken Sie dann auf **Weiter**.
5. Klicken Sie auf **Datenbank entladen, die auf einer aktuellen Version des Servers läuft**, und wählen Sie dann die Datenbank aus der Liste. Klicken Sie auf **Weiter**.
6. Wählen Sie, die Datenbank zu entladen und in eine neue Datenbank zu laden. Klicken Sie auf **Weiter**.
7. Geben Sie einen neuen Dateinamen für die Datenbank an.
8. Sie können auch die Seitengröße für die neue Datenbank angeben, doch die angegebene Seitengröße darf nicht größer als die Seitengröße des Datenbankservers sein. Die Standard-Seitengröße ist 4096 Byte. Sie können die Datenbankdatei verschlüsseln. Wenn Sie die starke Verschlüsselung wählen, müssen Sie bei jedem Start der Datenbank den Chiffrierschlüssel verwenden. Klicken Sie auf **Weiter**.
9. Klicken Sie auf **Struktur und Daten entladen**. Sie können auch beliebige weitere Optionen für Ihre Datenbank auswählen. Klicken Sie auf **Weiter**.
10. Klicken Sie auf **Alle Datenbankobjekte entladen**. Klicken Sie auf **Weiter**.
11. Legen Sie fest, ob nach dem Laden bzw. Entladen eine Verbindung mit der neuen Datenbank hergestellt werden soll.
12. Klicken Sie auf **Fertig stellen**, um den Vorgang zu starten.

Ergebnisse

Das Upgrade der Datenbank auf die neueste Version wird durchgeführt. Standardmäßig wird die Datenbank gestoppt und neu gestartet.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass der Neuaufbau einwandfrei abgeschlossen wurde, und testen Sie die Datenbank nach dem Upgrade mit Ihrer Anwendung.

Siehe auch

- „Sichern von Datenbanken“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Tipps zum Exportieren von Daten mit dem Assistenten zum Entladen einer Datenbank“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
- „Neuaufbau von Datenbanken“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]
- „Datenbankverschlüsselung und -entschlüsselung“ [[SQL Anywhere Server - Datenbankadministration](#)]

Datenbanken neu aufbauen (Befehlszeile)

Sie können das Dienstprogramm dbunload verwenden, um ein Upgrade einer SQL Anywhere-Datenbank der Version 10 oder höher auf die neueste Version durchzuführen.

Voraussetzungen

Der in *connection-string* angegebene Datenbankbenutzer muss die Systemprivilegien SELECT ANY TABLE und SERVER OPERATOR haben.

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Stellen Sie sicher, dass Sie exklusiven Zugriff auf die Datenbank haben, für die Sie das Upgrade durchführen möchten. Vergewissern Sie sich außerdem, dass der Pfad für die Dienstprogramme der Version 16 im Systempfad vor den Pfaden der anderen Dienstprogramme steht. Siehe „[Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind](#)“ auf Seite 298.

Achten Sie darauf, die Datenbank zu sichern, bevor Sie sie entladen und neu laden. Beispiel:

```
dbbackup -c "DBF=mydb.db;UID=DBA;PWD=sql" old-db-backup-dir
```

Kontext und Bemerkungen

Wenn Sie dbunload mit einer Datenbank der Version 10 oder höher verwenden, muss die Version von dbunload mit der Version des Datenbankservers übereinstimmen, der für den Zugriff auf die Datenbank verwendet wird. Wenn eine ältere Version von dbunload mit einem neueren Datenbankserver verwendet wird oder umgekehrt, wird ein Fehler zurückgegeben.

Wenn Sie eine Datenbank neu aufbauen, die an der SQL Remote-Replikation beteiligt ist oder bei der es sich um eine entfernte Datenbank in einer MobiLink-Installation handelt, und das dbunload-Dienstprogramm benutzen, müssen Sie die Option -ar oder -an verwenden. Die Optionen stellen sicher, dass die Transaktionslog-Offsets für die neue Datenbank mit denen der alten Datenbank übereinstimmen.

Datenbanken neu aufbauen (Befehlszeile)

1. Führen Sie das Dienstprogramm zum Entladen (dbunload) aus und verwenden Sie die Option -an, um eine Datenbank zu erstellen.

```
dbunload -c "connection-string" -an new-db-file
```

Dieser Befehl erstellt eine Datenbank. Wenn Sie die vorhandene Datenbank durch eine Datenbank nach Upgrade ersetzen möchten, verwenden Sie statt der Option -an die Option -ar. Um die Option -ar verwenden zu können, müssen Sie eine Verbindung mit einem Personal-Datenbankserver herstellen oder mit einem Netzwerk-Datenbankserver auf demselben Computer, auf dem sich das Dienstprogramm zum Entladen (dbunload) befindet.

Weitere Hinweise zu anderen Optionen des Dienstprogramms Entladen (dbunload) finden Sie unter „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

2. Fahren Sie die Datenbank herunter und speichern Sie das Transaktionslog, bevor Sie die neu geladene Datenbank verwenden.

Wenn Sie die Merkmale der Datenbank während des Entladens und Neuladens ändern möchten (z.B. eine Datenbank so einstellen, dass sie nicht mehr die Groß- und Kleinschreibung berücksichtigt), ist das Verfahren komplizierter. Siehe „Neuaufbau von Datenbanken“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)].

Ergebnisse

Das Upgrade der Datenbank auf die neueste Version wird durchgeführt. Standardmäßig wird die Datenbank gestoppt und neu gestartet.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass der Neuaufbau einwandfrei abgeschlossen wurde, und testen Sie die Datenbank nach dem Upgrade mit Ihrer Anwendung.

Siehe auch

- „Sichern von Datenbanken“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Dienstprogramm zum Entladen (dbunload)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Neuaufbau von Datenbanken“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]

Neuaufbauprozess für Datenbanken der Version 9 und höher

Es wird empfohlen, die Datenbank vor dem Neuaufbau zu sichern.

Hinweis

Die Seitengröße für eine Datenbank (in Byte) kann 2048, 4096, 8192, 16384 oder 32768 betragen, wobei der Standardwert die Seitengröße der ursprünglichen Datenbank ist.

Weitere Informationen über das Upgrade von Windows Mobile-Datenbanken finden Sie unter „Neuaufbau von Datenbanken unter Windows Mobile“ [[SQL Anywhere Server - Datenbankadministration](#)].

Hinweis

SQL Anywhere 9.0.2 für Mac OS X wurde auf PPC unterstützt, während SQL Anywhere 10.0.0 und höher für Mac OS X auf Intel unterstützt wird. Wenn Sie eine Datenbank der Version 9.0.2 oder früher unter Mac OS X betreiben, gibt es zwei Möglichkeiten zum Entladen der Datenbank:

- Sie entladen die Datenbank mit der Version 9.0.2.
- Sie kopieren die Datenbank auf eine andere Plattform, auf der SQL Anywhere 16 installiert ist, und entladen die Datenbank dann mithilfe der Version 16.

Wenn die Datenbank entladen wurde, können Sie sie mit der Version 16 auf Mac OS X neu laden.

Vorsicht

Das Entladen und Neuladen einer großen Datenbank kann zeitaufwändig sein und viel Plattenspeicher benötigen. Der Prozess erfordert auf der Festplatte verfügbaren Speicherplatz, der etwa der doppelten Datenbankgröße entspricht, um die entladenen Daten und neue Datenbankdatei zu speichern.

Einschränkungen für das Upgrade

Es gibt einige Einschränkungen, die beim Neuaufbau von Datenbanken bis Version 9 mit den Tools der Version 16 beachtet werden müssen:

- Trennen Sie die Verbindungen der Datenbank mit allen früheren Versionen des Datenbankservers und fahren Sie alle Datenbankserver früherer Versionen herunter, die auf dem Computer ausgeführt werden. Außerdem müssen alle Datenbankserver der Version 16 heruntergefahren werden, die auf dem Computer laufen. Wenn dbunload nicht fortgesetzt werden kann, da das Programm einen solchen Datenbankserver erkennt, gibt es einen Fehler aus und schlägt fehl.
- Beziehen Sie die Verbindungsparameter ENG, START oder LINKS nicht in die Verbindungszeichenfolge für die alte Datenbank ein (angegeben in der Option -c). Wenn Sie diese Parameter angeben, werden sie ignoriert und eine Warnung wird angezeigt. Geben Sie in Sybase Central im Fenster **Verbinden** keine Werte in die Felder **Servername** und **Startzeile** ein.
- Führen Sie dbunload auf einem Computer aus, der direkten Dateisystemzugriff auf die alte Datenbank hat. (dbunload muss eine Shared Memory-Verbindung mit der Datenbank herstellen können.)
- Führen Sie auf keinen Fall einen Datenbankserver mit dem Namen dbunload_support_engine auf dem Computer aus, auf dem der Neuaufbau durchgeführt wird.
- Entladen Sie eine Datenbank der Version 9 oder früher nicht, wenn für die Datenbankdatei eine Wiederherstellung erforderlich ist. Wenn Sie das Dienstprogramm zum Entladen (dbunload) für eine Datenbankdatei verwenden, für die eine Wiederherstellung erforderlich ist, wird eine Meldung zurückgegeben, die darüber informiert, dass die Datenbank nicht gestartet werden konnte. Verwenden Sie einen Datenbankserver der Version 9, um die Datenbank zu starten, und stoppen Sie dann die Datenbank, bevor Sie nochmals versuchen, dbunload auszuführen. Um eine Datenbank der Version 9 oder höher zu entladen, müssen Sie die Datenbank im schreibgeschützten Modus starten können. Siehe [„Datenbankoption -r“ \[SQL Anywhere Server - Datenbankadministration\]](#).

Besondere Hinweise

- **Groß- und Kleinschreibung bei Kennwörtern** In neu erstellten SQL Anywhere-Datenbanken der Version 16 berücksichtigen alle Kennwörter die Groß- und Kleinschreibung, unabhängig davon, ob die Datenbank die Groß- und Kleinschreibung berücksichtigt. Das DBA-Standardkennwort neuer Datenbanken lautet **sql**.

Wenn Sie eine vorhandene Datenbank neu erstellen, legt SQL Anywhere wie folgt fest, ob das Kennwort die Groß- und Kleinschreibung berücksichtigt:

- Wenn das Kennwort ursprünglich in eine Datenbank eingegeben wurde, in der die Groß- und Kleinschreibung nicht berücksichtigt wurde, wird bei dem Kennwort weiterhin die Groß- und Kleinschreibung nicht berücksichtigt.

- Wenn das Kennwort ursprünglich in eine Datenbank eingegeben wurde, in der die Groß- und Kleinschreibung berücksichtigt wurde, wird die Schreibweise bei Kennwörtern in Großschreibung und in einer Mischung aus Groß- und Kleinschreibung weiterhin berücksichtigt. Wenn das Kennwort jedoch in Kleinschreibung eingegeben wurde, wird die *Groß- und Kleinschreibung nicht berücksichtigt*.
- Bei Änderungen vorhandener Kennwörter sowie bei neuen Kennwörtern wird die Groß- und Kleinschreibung berücksichtigt.
- **Seitengrößen** Die Standardgröße der Datenbankseiten für SQL Anywhere 16 ist 4096 Byte. In Version 16 werden die Seitengrößen 2048 Byte, 4096 Byte, 8192 Byte, 16384 Byte und 32768 Byte unterstützt. Wenn Ihre alte Datenbank eine nicht unterstützte Seitengröße verwendet, hat die neue Datenbank standardmäßig die Seitengröße von 4096 Byte. Sie können die dbinit-Option -p oder die dbunload-Option -ap verwenden, um eine andere Seitengröße festzulegen. Siehe „[Dienstprogramm Initialisierung \(dbinit\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)] und „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Kollationen** In Version 9 und früher unterstützte SQL Anywhere eine Kollation mit CHAR-Datentypen. Diese Kollation hat den SQL Anywhere-Kollationsalgorithmus (SQL Anywhere Collation Algorithm, SACA) verwendet. In Version 10 und höher unterstützt SQL Anywhere die beiden Kollationsalgorithmen SACA und UCA (Unicode Collation Algorithm). Die SACA-Kollation der alten Datenbank wird entladen und in der neu aufgebauten Datenbank erneut verwendet, außer Sie legen eine neue oder andere Kollation für die neu aufgebaute Datenbank fest.

Wenn Sie eine Datenbank mit einer benutzerdefinierten Kollation neu aufbauen, wird die Kollation nur beibehalten, wenn Sie sie in einem einzigen Schritt neu aufbauen (internes Entladen). Wenn Sie wählen, die Datenbank zu entladen und dann das Schema und die Daten in eine erstellte Datenbank zu laden, müssen Sie eine der unterstützten Kollationen verwenden. Siehe „[Alternative Kollationen](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

- **Datenbankdateigröße** Aufgrund von Indexänderungen in SQL Anywhere kann es vorkommen, dass eine Datenbank nach dem Neuaufbau durch Entladen und Neuladen kleiner ist als die ursprüngliche Datenbank. Die geringere Datenbankgröße ist kein Hinweis auf ein Problem oder einen Datenverlust.

Bekannte Probleme

Falls die Neuerstellung fehlschlägt, wenn Sie dbunload oder den **Assistenten zum Entladen einer Datenbank** ausführen, können Sie mit den folgenden Schritten die Ursache für den Fehler ermitteln.

1. Erstellen Sie eine neue, leere Datenbank der Version 16.

```
dbinit -dba DBA,sql test.db
```

2. Wenden Sie die Datei *reload.sql* auf die leere Datenbank an.

```
dbisql -c "DBF=test.db;UID=DBA;pwd=sql" reload.sql
```

3. Ändern Sie die Datei *reload.sql* oder die ursprüngliche Datenbank anhand der Meldungen, die Sie erhalten, wenn Sie die Datei *reload.sql* auf die neue Datenbank anwenden.

Die folgende Tabelle listet bekannte Probleme, die zu einem Fehlschlag der Neuerstellung führen, sowie die jeweiligen Lösungen auf.

Bekanntes Problem	Lösung
Eine DECLARE LOCAL TEMPORARY TABLE-Anweisung in einer Prozedur oder einem Trigger verursacht einen Syntaxfehler, wenn der Tabellename einen Eigentümernamen als Präfix hat.	Entfernen Sie den Eigentümernamen.
Wenn eine CREATE TRIGGER-Anweisung keinen Eigentümernamen für die Tabelle enthält, für die der Trigger definiert wurde, und die Tabelle mit einem Eigentümer qualifiziert werden muss, wenn sie von dem Benutzer referenziert wird, der die Datei <i>reload.sql</i> ausführt, schlägt die Anweisung fehl. Es wird ein Fehler ausgegeben, der darauf hinweist, dass die Tabelle nicht gefunden wurde.	Stellen Sie dem Tabellennamen den Eigentümernamen als Präfix voran.
Wenn ein Objektname (z.B. eine Tabelle, Spalte, Variable oder ein Parametername) ein reserviertes Wort ist, das in einer höheren Version von SQL Anywhere eingeführt wurde (z.B. NCHAR), schlägt das erneute Laden fehl. Beispiel: <pre>CREATE PROCEDURE p() BEGIN DECLARE NCHAR INT; SET NCHAR = 1; END;</pre>	Ändern Sie alle Verweise auf das reservierte Wort, indem Sie einen anderen Namen verwenden. Bei Variablennamen ist die Verwendung des Präfixes @ eine allgemeine Konvention, die Namenskonflikte vermeidet. Eine vollständige Liste der reservierten Wörter finden Sie unter „Reservierte Wörter“ [SQL Anywhere Server - SQL-Referenzhandbuch].
Wenn eine Datenbank mit dbunload aus Version 9 oder früher entladen wird, kann die Datei <i>reload.sql</i> Aufrufe der ml_add_property-Systemprozedur enthalten, doch diese Prozedur ist in einer neuen Datenbank der Version 16 nicht enthalten.	Entladen Sie die Datenbank mit dem dbunload-Dienstprogramm der Version 16. Weitere Hinweise dazu, wie Sie sicherstellen können, dass die richtige Version der Datenbank-Dienstprogramme verwendet wird, finden Sie unter „ Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind “ auf Seite 298.

Bekanntes Problem	Lösung
Wenn Sie eine Datenbank mit dem Dienstprogramm dbunload aus Version 9 oder früher entladen, werden Ansichten, die Transact-SQL-Outer-Joins (Angabe *= or =*) verwenden, möglicherweise nicht richtig erstellt, wenn sie erneut geladen werden.	Fügen Sie die folgende Zeile in das Neulade-Skript ein: <pre>SET TEMPORARY OPTION tsql_outer_joins='on';</pre> Sie sollten später alle Ansichten, die Transact-SQL-Outer-Joins verwenden, neu erstellen.
Die [NOT] DETERMINISTIC-Klausel wird von den Anweisungen CREATE PROCEDURE und ALTER PROCEDURE nicht unterstützt. Wenn die Klausel vorhanden ist, schlägt das erneute Laden fehl und es wird ein Syntaxfehler gemeldet.	Wenn Sie ein Upgrade einer Datenbank durchführen, die benutzerdefinierte Prozeduren enthält, die die [NOT] DETERMINISTIC-Klausel enthalten, müssen Sie die Klausel entfernen, bevor Sie die Datenbank entladen und neu laden.

Datenbanken der Version 9 oder früher neu aufbauen (Sybase Central)

Der **Assistent zum Entladen einer Datenbank** baut eine Datenbank aus Version 9 oder früher neu auf, indem er sie in eine Reload-Datei und in Datendateien entlädt und in eine neue Datenbank lädt oder entlädt und neu in eine vorhandene Datenbank lädt. Neu aufgebaute Datenbanken unterstützen alle neuen Funktionen und Performanceverbesserungen der aktuellen Softwareversion.

Voraussetzungen

Sie müssen die Systemprivilegien SELECT ANY TABLE und SERVER OPERATOR haben.

Es wird empfohlen, die Datenbank vor dem Neuaufbau zu sichern.

Die Datenbankdatei muss sich auf demselben Computer befinden wie die SQL Anywhere 16-Installation.

Es ist nicht möglich, eine Teilmenge von Tabellen aus einer Datenbank zu entladen. Verwenden Sie dazu das Dienstprogramm dbunload.

Wenn der **Assistent zum Entladen der Datenbank** feststellt, dass die Datenbankdatei bereits läuft, wird die Datenbank gestoppt, bevor der Entladevorgang ausgeführt wird.

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Defragmentieren Sie, wenn möglich, das Laufwerk, auf dem die neue Datenbank gespeichert werden soll, da ein fragmentiertes Laufwerk die Performance der Datenbank beeinträchtigen kann.

Stellen Sie sicher, dass nur Sie allein auf die entsprechende Datenbank zugreifen können. Es dürfen keine anderen Benutzer verbunden sein.

Datenbanken der Version 9 oder früher neu aufbauen (Sybase Central)

1. Klicken Sie auf **Start&rarrProgramme&rarrSQL Anywhere 16&rarrAdministrationtools&rarrSybase Central**.
2. Klicken Sie auf **Extras&rarrSQL Anywhere 16&rarrDatenbank entladen**.
3. Lesen Sie die Einführungsseite des **Assistenten zum Entladen einer Datenbank**, und klicken Sie dann auf **Weiter**.
4. Klicken Sie auf **Auf einer früheren Version des Servers laufende oder nicht aktive Datenbank entladen**. Geben Sie die Verbindungsparameter für die Datenbank an. Klicken Sie auf **Weiter**.
5. Klicken Sie auf **Entladen und in eine neue Datenbank laden**. Klicken Sie auf **Weiter**.
6. Geben Sie einen neuen Dateinamen für die Datenbank an. Klicken Sie auf **Weiter**.

Sie können die Seitengröße für die neue Datenbank festlegen. In Version 16 beträgt die standardmäßige (und empfohlene) Seitengröße 4096 Byte.

Sie können die Datenbankdatei verschlüsseln. Wenn Sie die starke Verschlüsselung wählen, benötigen Sie bei jedem Start der Datenbank den Chiffrierschlüssel. Siehe „[Datenbankverschlüsselung und -entschlüsselung](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

7. Wählen Sie "Struktur und Daten entladen". Klicken Sie auf **Weiter**.
8. Legen Sie fest, ob nach dem Neuaufbau eine Verbindung mit der neuen Datenbank hergestellt werden soll.
9. Klicken Sie auf **Fertig stellen**. Überprüfen Sie die neue Datenbank, um sicherzustellen, dass der Neuaufbau einwandfrei abgeschlossen wurde.

Ergebnisse

Das Upgrade der Datenbank auf die neueste Version wird durchgeführt. Standardmäßig wird die Datenbank gestoppt und neu gestartet.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass der Neuaufbau einwandfrei abgeschlossen wurde, und testen Sie die neu aufgebaute Datenbank mit Ihrer Anwendung.

Siehe auch

- „[Sichern von Datenbanken](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „[Neuaufbau von Datenbanken](#)“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]

Datenbanken der Version 9 oder früher mit dem Dienstprogramm zum Entladen neu aufbauen (Befehlszeile)

Sie können das Dienstprogramm zum Entladen (dbunload) mit der Option -an oder -ar verwenden, um eine Datenbank der Version 9 oder früher neu aufzubauen. Neu aufgebaute Datenbanken unterstützen alle neuen Funktionen und Performanceverbesserungen der aktuellen Softwareversion.

Voraussetzungen

Der in *connection-string* angegebene Datenbankbenutzer muss die Systemprivilegien SELECT ANY TABLE und SERVER OPERATOR haben.

Es wird empfohlen, die Datenbank vor dem Neuaufbau zu sichern.

Die Datenbankdatei muss sich auf demselben Computer befinden wie die SQL Anywhere 16-Installation.

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Stellen Sie sicher, dass nur Sie allein auf die entsprechende Datenbank zugreifen können. Es dürfen keine anderen Benutzer verbunden sein.

Stellen Sie sicher, dass die Dienstprogramme der Version 16 in Ihrem Systempfad vor den anderen Dienstprogrammen aufgeführt sind. Siehe „[Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind](#)“ auf Seite 298.

Kontext und Bemerkungen

Die Option -an wird empfohlen, weil sie eine neue Datenbank erstellt und dabei die ursprüngliche Datenbank intakt lässt. Die Option -ar ersetzt die alte Datenbank durch eine neue Datenbank der Version 16.

Datenbanken der Version 9 oder früher mit dem Dienstprogramm zum Entladen neu aufbauen (Befehlszeile)

1. Fahren Sie alle SQL Anywhere- und Adaptive Server Anywhere-Datenbankserver herunter, da das dbunload-Dienstprogramm der Version 16 nicht für eine Datenbank verwendet werden kann, die auf einer früheren Version des Datenbankservers läuft. Beispiel:

```
dbstop -c "DBF=mydb.db;UID=DBA;PWD=sql"
```

2. Defragmentieren Sie, wenn möglich, das Laufwerk, auf dem die neue Datenbank gespeichert werden soll, da ein fragmentiertes Laufwerk die Performance der Datenbank beeinträchtigen kann.
3. Sichern Sie die Datenbank. Beispiel:

```
dbbackup -c "DBF=mydb.db;UID=DBA;PWD=sql" old-db-backup-dir
```

Sie müssen das BACKUP DATABASE-Systemprivileg haben. Siehe „Sichern von Datenbanken“ [[SQL Anywhere Server - Datenbankadministration](#)].

4. Führen Sie das Dienstprogramm Entladen (dbunload) mit der Option -an oder -ar aus, um eine neue Datenbank zu erstellen.

```
dbunload -c "connection-string" -an database-filename
```

Beispiel:

```
dbunload -c "DBF=mydb.db;UID=DBA;PWD=sql" -an mydb16.db
```

Dieser Befehl erstellt eine Datenbank (durch die Option -an). Wenn Sie die Option -ar angeben, wird die vorhandene Datenbank durch eine neu aufgebaute Datenbank ersetzt. Um die Option -ar zu verwenden, müssen Sie eine Verbindung zu einem Personal Server oder einem Netzwerkservers auf demselben Computer herstellen, auf dem sich das Dienstprogramm Entladen (dbunload) befindet.

Ergebnisse

Das Upgrade der Datenbank auf die neueste Version wird durchgeführt. Standardmäßig wird die Datenbank gestoppt und neu gestartet.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass der Neuaufbau einwandfrei abgeschlossen wurde, und testen Sie die neu aufgebaute Datenbank mit Ihrer Anwendung.

Siehe auch

- „Dienstprogramm zum Entladen (dbunload)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Upgradevorgang für Datenbanken der Version 10 und später

Wenn Sie eine Datenbank aktualisieren, werden Systemtabellen, Systemprozeduren und Datenbankoptionen in der Datenbank hinzugefügt und geändert, um die Funktionen der Version 16 zu aktivieren. Dabei wird das Dateiformat, mit dem Daten auf der Festplatte gespeichert und von dort abgerufen werden, nicht geändert. Somit stehen auch nicht alle neuen Funktionen und Performanceverbesserungen der letzten Softwareversion zur Verfügung.

Weitere Hinweise zum Upgrade des Datenbankdateiformats finden Sie unter „Neuaufbauprozess für Datenbanken der Version 10 und höher“ auf Seite 300.

Upgrade von Datenbanken der Version 10 und höher (Sybase Central)

Sie können den **Assistenten zum Upgrade einer Datenbank** verwenden, um ein Upgrade einer Datenbank der Version 10 oder höher durchzuführen, indem Sie Systemtabellen, Systemprozeduren und Datenbankoptionen so hinzufügen und ändern, dass die Funktionen der Version 16 aktiviert werden.

Voraussetzungen

Sie müssen die folgenden Systemprivilegien haben:

- ALTER DATABASE
- BACKUP DATABASE
- SERVER OPERATOR

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Vorsicht

Sichern Sie Ihre Datenbankdateien vor dem Upgrade. Wenn Sie das Upgrade auf vorhandene Dateien anwenden, werden diese Dateien unbrauchbar, falls das Upgrade fehlschlägt. Um Ihre Datenbank sichern zu können, benötigen Sie das BACKUP DATABASE-Systemprivileg. Weitere Hinweise zur Sicherung der Datenbank finden Sie unter „[Sicherung und Datenwiederherstellung](#)“ [*SQL Anywhere Server - Datenbankadministration*].

Kontext und Bemerkungen

Der **Assistent zum Upgrade einer Datenbank** führt kein Upgrade einer Datenbank der Version 9.0.2 oder früher auf die Version 16 durch. Um bestehende Datenbanken der Version 9.0.2 auf Version 16 umzustellen, müssen Sie die Datenbank mit dbunload oder dem **Assistenten zum Entladen einer Datenbank** entladen und neu laden. Siehe [Upgrade von Datenbanken bis Version 9](#) auf Seite 296.

Upgrade von Datenbanken (Sybase Central)

1. Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.
2. Klicken Sie auf **Start→Programme→SQL Anywhere 16→Administrationtools→Sybase Central**.
3. Stellen Sie vom **SQL Anywhere-16**-Plug-In aus eine Verbindung mit der Datenbank her, für die Sie ein Upgrade vornehmen. Die Datenbank muss auf einem Datenbankserver der Version 16 ausgeführt werden.
4. Klicken Sie auf **Extras→SQL Anywhere 16→Datenbank erstellen**.
5. Befolgen Sie die Anweisungen im **Assistenten zum Upgrade einer Datenbank**. Standardmäßig wird die Datenbank gestoppt und neu gestartet.

6. (Optional) Stoppen Sie die Datenbank und archivieren Sie das Transaktionslog, indem Sie eine Kopie davon erstellen, bevor Sie die Datenbank nach dem Upgrade verwenden, falls Sie diese Aktion nicht bereits im Assistenten gewählt haben.

Tipp

Sie können den **Assistenten zum Upgrade einer Datenbank** auch wie folgt aufrufen:

- Klicken Sie mit der rechten Maustaste auf eine Datenbank und klicken Sie auf **Upgrade einer Datenbank**.
- Wählen Sie eine Datenbank aus und klicken Sie dann auf **Datei > Upgrade einer Datenbank**.

Ergebnisse

Das Upgrade der Datenbank wird durchgeführt und enthält neue und geänderte Systemtabellen, Systemprozeduren und Datenbankoptionen.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass das Upgrade einwandfrei abgeschlossen wurde, und testen Sie die Datenbank nach dem Upgrade mit Ihrer Anwendung.

Siehe auch

- „Dienstprogramm zum Upgrade (dbupgrad)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Upgrade von Datenbanken der Version 10 und höher (Befehlszeile)

Sie können das Dienstprogramm dbupgrad verwenden, um ein Upgrade einer Datenbank der Version 10 oder höher durchzuführen, indem Sie Systemtabellen, Systemprozeduren und Datenbankoptionen so hinzufügen und ändern, dass die Funktionen der Version 16 aktiviert werden.

Voraussetzungen

Der in *connection-string* angegebene Datenbankbenutzer muss das ALTER DATABASE-Systemprivileg haben und als einziger Benutzer mit der Datenbank verbunden sein.

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Stellen Sie sicher, dass Sie exklusiven Zugriff auf die Datenbank haben, für die Sie das Upgrade durchführen möchten. Vergewissern Sie sich zudem, dass die Dienstprogramme der Version 16 im Systempfad vor anderen Dienstprogrammen stehen. Siehe „[Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind](#)“ auf Seite 298.

Vorsicht

Sichern Sie Ihre Datenbankdateien vor dem Upgrade. Wenn Sie das Upgrade auf vorhandene Dateien anwenden, werden diese Dateien unbrauchbar, falls das Upgrade fehlschlägt. Weitere Hinweise zur Sicherung der Datenbank finden Sie unter „[Sicherung und Datenwiederherstellung](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

Upgrade von Datenbanken (Befehlszeile)

1. Führen Sie das Dienstprogramm zum Upgrade (dbupgrad) für die Datenbank aus:

```
dbupgrad -c "connection-string"
```

Weitere Hinweise finden Sie unter „[Dienstprogramm zum Upgrade \(dbupgrad\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].

2. Fahren Sie die Datenbank herunter und speichern Sie das Transaktionslog, bevor Sie die Datenbank nach dem Upgrade verwenden.

Ergebnisse

Das Upgrade der Datenbank wird durchgeführt und enthält neue und geänderte Systemtabellen, Systemprozeduren und Datenbankoptionen.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass das Upgrade einwandfrei abgeschlossen wurde, und testen Sie die Datenbank nach dem Upgrade mit Ihrer Anwendung.

Siehe auch

- „[Dienstprogramm zum Upgrade \(dbupgrad\)](#)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Upgrade von Datenbanken der Version 10 und höher (SQL)

Sie können die ALTER DATABASE-Anweisung verwenden, um ein Upgrade einer Datenbank der Version 10 oder höher durchzuführen, indem Sie Systemtabellen, Systemprozeduren und Datenbankoptionen so hinzufügen und ändern, dass die Funktionen der Version 16 aktiviert werden.

Voraussetzungen

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Sie müssen Eigentümer der Datenbank sein oder das ALTER DATABASE-Systemprivileg haben und als einziger Benutzer mit der Datenbank verbunden sein.

Vorsicht

Sichern Sie Ihre Datenbankdateien vor dem Upgrade. Wenn Sie das Upgrade auf vorhandene Dateien anwenden, werden diese Dateien unbrauchbar, falls das Upgrade fehlschlägt.

Upgrade einer Datenbank (SQL)

1. Stellen Sie in Interactive SQL oder einer anderen Anwendung, die SQL-Anweisungen ausführen kann, eine Verbindung zur Datenbank her. Keine andere Verbindung kann die Datenbank zur gleichen Zeit verwenden.
2. Führen Sie eine ALTER DATABASE-Anweisung aus.

Folgende Anweisung führt z. B. ein Upgrade einer Datenbank aus:

```
ALTER DATABASE UPGRADE;
```

3. Fahren Sie die Datenbank herunter und speichern Sie das Transaktionslog, bevor Sie die Datenbank nach dem Upgrade verwenden.

Ergebnisse

Das Upgrade der Datenbank wird durchgeführt und enthält neue und geänderte Systemtabellen, Systemprozeduren und Datenbankoptionen.

Nächste Schritte

Überprüfen Sie die neue Datenbank, um sicherzustellen, dass das Upgrade einwandfrei abgeschlossen wurde, und testen Sie die Datenbank nach dem Upgrade mit Ihrer Anwendung.

Siehe auch

- „ALTER DATABASE-Anweisung“ [[SQL Anywhere Server - SQL-Referenzhandbuch](#)]
- „Vorsichtsmaßnahmen für Upgrade und Neuaufbau“ auf Seite 297
- „Sicherung und Datenwiederherstellung“ [[SQL Anywhere Server - Datenbankadministration](#)]

Upgrade und Neuaufbau in einem Datenbankspiegelungssystem

Alle Datenbankserver in einem Datenbankspiegelungssystem müssen dieselbe Minor Release verwenden.

Datenbankserver-Software für ein Spiegelungssystem auf eine Hauptversion aktualisieren, ohne die Datenbank neu aufzubauen

Aktualisieren Sie die Software der Datenbankserver in einem Spiegelungssystem, indem Sie die Software installieren, die einzelnen Datenbankserver stoppen und die Datenbanken auf Servern starten, auf denen die neue Software ausgeführt wird. Für die Datenbank wird weder ein Upgrade noch ein Neuaufbau durchgeführt.

Voraussetzungen

Sie müssen das BACKUP DATABASE-Systemprivileg haben. Sie müssen Eigentümer der Datenbank sein oder das VALIDATE ANY OBJECT-Systemprivileg haben.

Standardmäßig benötigen Sie das SERVER OPERATOR-Systemprivileg, um Netzwerk-Datenbankserver stoppen zu können.

Kontext und Bemerkungen

Testen Sie die folgenden Schritte in einer Nicht-Produktionsumgebung mit Ihrer Anwendung, bevor Sie sie in einer Produktionsumgebung ausführen.

Alle Datenbankserver in einem Datenbankspiegelungssystem müssen dieselbe Minor Release von SQL Anywhere verwenden. Deshalb wird das Spiegelungssystem während der Aktualisierung vorübergehend gestoppt.

SQL Anywhere-Hauptversionen in Datenbankspiegelungssysteme übernehmen

1. Erstellen Sie eine Sicherung der Primärdatenbank, kopieren Sie die Sicherung und validieren Sie die Kopie der Sicherung.

Mit dem folgenden Befehl wird beispielsweise eine Datenbank namens *mydb.db* gesichert:

```
dbbackup -c "DBN=mydb;ENG=myserver;UID=DBA;PWD=sql" backup-dir
```

Wenn Sie eine Kopie der Sicherung erstellen und die Transaktionslogs auf die Kopie der Sicherung anwenden möchten, führen Sie die folgenden Befehle aus:

```
xcopy backup-dir\mydb.db validatebackup-dir  
xcopy backup-dir\mydb.log validatebackup-dir
```

Validieren Sie diese Kopie.

```
dbvalid -c "DBF=validatebackup-dir\backupmydb.db;UID=DBA;PWD=sql"
```

Wenn die Kopie der Sicherung nicht validiert wird, beheben Sie die Probleme, aufgrund derer die Validierung fehlschlägt, bevor Sie fortfahren. Andernfalls riskieren Sie den Verlust von Daten.

2. Installieren Sie die neue Software auf Primär-, Spiegel- und Arbiterserver.
3. Wenn das Spiegelungssystem in ein Scale-Out-System mit Schreibschutz eingebunden ist, installieren Sie die Software auf den Kopieknoten.
4. Stoppen Sie die Server in der folgenden Reihenfolge:
 - a. Kopieknoten
 - b. Spiegelserver
 - c. Primärserver
 - d. Arbiterserver

Führen Sie beispielsweise das Stopp-Dienstprogramm (dbstop) aus:

```
dbstop -y -c "UID=DBA;PWD=sql;Server=myserver"
```

5. (Optional) Führen Sie das Upgrade bzw. den Neuaufbau der Datenbanken durch.
6. Starten Sie die Datenbanken auf den neuen Servern in der folgenden Reihenfolge:
 - a. Arbiterserver, Primärserver und Spiegelserver
 - b. Kopieknotten

Ergebnisse

Die Datenbanken im Spiegelungssystem werden mit der neuen Version der Software ausgeführt.

Nächste Schritte

Überprüfen Sie das Datenbankspiegelungssystem, um sicherzustellen, dass die Aktualisierung einwandfrei abgeschlossen wurde, und testen Sie das Datenbankspiegelungssystem mit Ihrer Anwendung.

Siehe auch

- „Einen Datenbankserver in einem Spiegelungssystem stoppen“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Benutzerinitiiert Rollentausch (Failover)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Serverstopp-Dienstprogramm (dbstop)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Datenbankserver-Software für ein Spiegelungssystem auf eine andere Minor Release aktualisieren, ohne die Datenbank neu aufzubauen

Aktualisieren Sie die Software der Datenbankserver in einem Spiegelungssystem, indem Sie die Software installieren, die einzelnen Server stoppen und die Datenbank auf dem neuen Server starten. Alle Datenbankserver in einem Datenbankspiegelungssystem müssen dieselbe Minor Release verwenden.

Voraussetzungen

Sie müssen das BACKUP DATABASE-Systemprivileg haben. Sie müssen Eigentümer der Datenbank sein oder das VALIDATE ANY OBJECT-Systemprivileg haben.

Standardmäßig benötigen Sie das SERVER OPERATOR-Systemprivileg, um Netzwerk-Datenbankserver stoppen zu können.

Testen Sie die folgenden Schritte in einer Nicht-Produktionsumgebung mit Ihrer Anwendung, bevor Sie sie in einer Produktionsumgebung ausführen.

Kontext und Bemerkungen

Das Spiegelungssystem wird während der Aktualisierung vorübergehend gestoppt.

Haupt- oder Wartungsversionen von SQL Anywhere in Datenbankspiegelungssysteme übernehmen

1. Erstellen Sie eine Sicherung der Primärdatenbank, kopieren Sie die Sicherung und validieren Sie die Kopie der Sicherung.

Führen Sie beispielsweise den folgenden Befehl aus, um eine Datenbank namens *mydb.db* zu sichern:

```
dbbackup -c "DBN=mydb;ENG=myserver;UID=DBA;PWD=sql" backup-dir
```

Erstellen Sie eine Kopie der Sicherung:

```
xcopy backup-dir\*.db validatebackup-dir\backupmydb.db
```

Validieren Sie die Sicherungskopie:

```
dbvalid -c "DBF=validatebackup-dir\backupmydb.db;UID=DBA;PWD=sql"
```

Wenn die Kopie der Sicherung nicht validiert wird, beheben Sie die Probleme, aufgrund derer die Validierung fehlschlägt, bevor Sie fortfahren. Andernfalls riskieren Sie den Verlust von Daten.

2. Stoppen Sie die Server in der folgenden Reihenfolge:

- a. Kopieknoten
- b. Spiegelserver
- c. Primärserver
- d. Arbiterserver

Führen Sie beispielsweise das Stopp-Dienstprogramm (dbstop) aus:

```
dbstop -y -c "UID=DBA;PWD=sql;Server=myserver"
```

3. Installieren Sie die neue Software auf Primär-, Spiegel- und Arbiterserver.

4. Starten Sie den Server in der folgenden Reihenfolge:

- a. Arbiterserver
- b. Primärserver
- c. Spiegelserver

5. Wenn das Spiegelungssystem in ein Scale-Out-System mit Schreibschutz eingebunden ist, installieren Sie die Software auf den Kopieknoten und starten Sie die Kopieknoten neu.

Ergebnisse

Die Datenbanken im Spiegelungssystem werden mit der neuen Version der Software ausgeführt.

Nächste Schritte

Überprüfen Sie das Datenbankspiegelungssystem, um sicherzustellen, dass das Upgrade einwandfrei abgeschlossen wurde, und testen Sie das Datenbankspiegelungssystem mit Ihrer Anwendung.

Siehe auch

- „Einen Datenbankserver in einem Spiegelungssystem stoppen“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Benutzerinitiiert Rollentausch (Failover)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Serverstopp-Dienstprogramm (dbstop)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Datenbankserver-Software für ein Datenbankspiegelungssystem auf ein Supportpaket aktualisieren, ohne die Datenbank neu aufzubauen

Aktualisieren Sie die Software der Datenbankserver in einem Spiegelungssystem, indem Sie die einzelnen Server stoppen, die Software installieren und dann die Datenbank auf dem neuen Server neu starten.

Voraussetzungen

Sie müssen das BACKUP DATABASE-Systemprivileg haben. Sie müssen Eigentümer der Datenbank sein oder das VALIDATE ANY OBJECT-Systemprivileg haben.

Standardmäßig benötigen Sie das SERVER OPERATOR-Systemprivileg, um Netzwerk-Datenbankserver stoppen zu können.

Kontext und Bemerkungen

Da Sie Server im System einzeln stoppen können, kann das Spiegelungssystem während einer Supportpaket-Aktualisierung der Serversoftware weiterlaufen. Ein Failover vom Primärserver auf den Spiegelserver erfolgt während der Aktualisierung mindestens einmal. Verbindungen mit dem Primärserver und dem Spiegelserver werden während eines Failovers getrennt.

Testen Sie die folgenden Schritte in einer Nicht-Produktionsumgebung mit Ihren Anwendungen, bevor Sie sie in einer Produktionsumgebung ausführen.

Supportpakete in ein Datenbankspiegelungssystem übernehmen

1. Erstellen Sie eine Sicherung der Primärdatenbank, kopieren Sie die Sicherung und validieren Sie die Kopie der Sicherung.

Führen Sie beispielsweise den folgenden Befehl aus, um eine Datenbank namens *mydb.db* zu sichern:

```
dbbackup -c "DBN=mydb;ENG=myserver;UID=DBA;PWD=sql" backup-dir
```

Erstellen Sie eine Kopie der Sicherung:

```
xcopy backup-dir\*.db validatebackup-dir\backupmydb.db
```

Validieren Sie die Sicherungskopie:

```
dbvalid -c "DBF=validatebackup-dir\backupmydb.db;UID=DBA;PWD=sql"
```

Wenn die Kopie der Sicherung nicht validiert wird, beheben Sie die Probleme, aufgrund derer die Validierung fehlschlägt, bevor Sie fortfahren. Andernfalls riskieren Sie den Verlust von Daten.

2. Wenn das Spiegelungssystem Teil eines Scale-Out-Systems mit Schreibschutz ist, führen Sie für jeden Kopieknoten die folgenden Schritte durch:
 - a. Stoppen Sie den Kopieknotenserver.
 - b. Installieren Sie die Software.
 - c. Starten Sie den Kopieknoten.
3. Stoppen Sie den Spiegelserver.
4. Installieren Sie die Software auf dem Spiegelserver.
5. Starten Sie den Spiegelserver und stellen Sie sicher, dass er sich in einem synchronisierten Zustand befindet.
6. Stoppen Sie den Arbiterserver.
7. Installieren Sie die Software auf dem Arbitr.
8. Starten Sie den Arbiterserver.
9. Initiieren Sie einen Failover, indem Sie eine Verbindung mit der Primärdatenbank herstellen und die folgende Anweisung ausführen:

```
ALTER DATABASE SET PARTNER FAILOVER;
```

Verbindungen mit dem Primärserver und dem Spiegelserver werden während des Failovers getrennt. Der aktuelle Primärserver wird zum Spiegelserver.

10. Stoppen Sie den Spiegelserver.
11. Installieren Sie die Software auf dem Spiegelserver.
12. Starten Sie den Spiegelserver und stellen Sie sicher, dass er synchronisiert ist.

Ergebnisse

Die Datenbanken im Spiegelungssystem werden mit der neuen Version der Software ausgeführt.

Nächste Schritte

Überprüfen Sie das Datenbankspiegelungssystem, um sicherzustellen, dass das Upgrade einwandfrei abgeschlossen wurde, und testen Sie das Datenbankspiegelungssystem mit Ihrer Anwendung.

Siehe auch

- „Einen Datenbankserver in einem Spiegelungssystem stoppen“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Benutzerinitiiert Rollentausch (Failover)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Serverstopp-Dienstprogramm (dbstop)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Upgrade oder Neuaufbau (Entladen/Neuladen) der Datenbanken in einem Datenbankspiegelungssystem

Führen Sie ein Upgrade der Primärdatenbank durch und kopieren Sie anschließend die Datenbank nach dem Upgrade und das Transaktionslog auf den Spiegelserver. Das Spiegelungssystem wird vorübergehend gestoppt. Upgrades von Datenbanken werden in der Regel durchgeführt, wenn die Software auf eine neue Hauptversion oder Minor Release aktualisiert werden muss.

Voraussetzungen

Sie müssen das BACKUP DATABASE-Systemprivileg haben. Sie müssen Eigentümer der Datenbank sein oder das VALIDATE ANY OBJECT-Systemprivileg haben.

Standardmäßig benötigen Sie das SERVER OPERATOR-Systemprivileg, um Netzwerk-Datenbankserver starten oder stoppen zu können.

Um ein Upgrade einer Datenbank durchführen zu können, müssen Sie das ALTER DATABASE-Systemprivileg haben und als einziger Benutzer mit der Datenbank verbunden sein.

Um den Neuaufbau (Entladen/Neuladen) einer Datenbank mit dem Dienstprogramm zum Entladen (dbunload) durchführen zu können, benötigen Sie das SELECT ANY TABLE-Systemprivileg. Für einen Entladevorgang mit Neuladen benötigen Sie außerdem das SERVER OPERATOR-Systemprivileg.

Testen Sie die folgenden Schritte in einer Nicht-Produktionsumgebung mit Ihrer Anwendung, bevor Sie sie in einer Produktionsumgebung ausführen.

SQL Anywhere-Minor Releases in Datenbankspiegelungssysteme übernehmen

1. Installieren Sie die neue Version der Software auf jedem Computer im System. Alle Datenbankserver in einem Datenbankspiegelungssystem müssen dieselbe Wartungsversion von SQL Anywhere verwenden.
2. Erstellen Sie eine Sicherung der Primärdatenbank, kopieren Sie die Sicherung und validieren Sie die Kopie der Sicherung.

Führen Sie beispielsweise den folgenden Befehl aus, um eine Datenbank namens *mydb.db* zu sichern:

```
dbbackup -c "DBN=mydb;ENG=myserver;UID=DBA;PWD=sql" backup-dir
```

Erstellen Sie eine Kopie der Sicherung:

```
xcopy backup-dir\*.db validatebackup-dir\backupmydb.db
```

Validieren Sie die Sicherungskopie:

```
dbvalid -c "DBF=validatebackup-dir\backupmydb.db;UID=DBA;PWD=sql"
```

Wenn die Kopie der Sicherung nicht validiert wird, beheben Sie die Probleme, aufgrund derer die Validierung fehlschlägt, bevor Sie fortfahren. Andernfalls riskieren Sie den Verlust von Daten.

3. Stoppen Sie die Server in der folgenden Reihenfolge:

- a. Kopieknoten
- b. Spiegelserver
- c. Primärserver

4.

Option	Aktion
Upgrade der Datenbank	<p>Führen Sie das Dienstprogramm zum Upgrade (dbupgrad) für die Primärdatenbank aus. Beispiel:</p> <pre>dbupgrad -c "UID=DBA;PWD=sql;DBF=C:\primary-database.db"</pre> <p>Das Upgrade der Datenbank wird durchgeführt, ein neues Transaktionslog wird erstellt und die Datenbank wird gestoppt. Sie können die alten Transaktionslogs löschen.</p>
Neuaufbau der Datenbank (Entladen/Neuladen)	<pre>dbunload -c "UID=DBA;PWD=sql;DBF=C:\primary-database.db"</pre>

5. Kopieren Sie die Datenbank nach dem Upgrade bzw. Neuaufbau sowie ihr neues Transaktionslog auf den Spiegelserver und auf alle Scale-Out-Kopieknoten mit Schreibschutz.

6. Starten Sie den Server in der folgenden Reihenfolge:

- a. Primärserver
- b. Spiegelserver
- c. Kopieknoten

Ergebnisse

Für die Datenbanken im Spiegelungssystem wurde ein Upgrade bzw. Neuaufbau durchgeführt und das Spiegelungssystem läuft.

Nächste Schritte

Überprüfen Sie das Datenbankspiegelungssystem, um sicherzustellen, dass das Upgrade einwandfrei abgeschlossen wurde, und testen Sie das Datenbankspiegelungssystem mit Ihrer Anwendung.

Siehe auch

- „Einen Datenbankserver in einem Spiegelungssystem stoppen“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Benutzerinitiiert Rollentausch (Failover)“ [[SQL Anywhere Server - Datenbankadministration](#)]
- „Serverstopp-Dienstprogramm (dbstop)“ [[SQL Anywhere Server - Datenbankadministration](#)]

Fehlerbehandlung: Datenbank-Upgrades

In diesem Abschnitt werden einige häufige Probleme beschrieben, die beim Upgrade der Datenbank auftreten können.

Fehlerbehandlung: Sicherstellen, dass während des Übernehmens eines Supportpakets keine JDBC-Anwendungen laufen

Nach dem Übernehmen eines Supportpakets stellen Sie möglicherweise fest, dass Ihre JDBC-Anwendungen nicht mehr funktionieren, und Sie erhalten eine Fehlermeldung ähnlich der folgenden:

`Der sajdbc4.jar-Build stimmt nicht mit dem Shared Object-Build überein.`

Diese Meldung wird möglicherweise zurückgegeben, weil entweder Interactive SQL, Sybase Central, die Schnelllader oder Ihre eigenen JDBC-Anwendungen ausgeführt wurden, als das Supportpaket übernommen wurde. In diesem Fall sperrt die Java VM zwar alle DLLs bzw. Shared Objects, die von ihr geladen werden, aber nicht die JAR-Dateien. Dies führt dazu, dass beim Übernehmen des Supportpakets häufig zwar die JAR-Dateien sajdbc4 und jodbc4 aktualisiert werden, aber nicht die entsprechenden DLLs (dbjdbc16 bzw. dbjodbc16) oder Shared Objects. Wenn die JDBC-Anwendung neu gestartet wird, passt die JDBC-JAR-Datei nicht zum Build der dazugehörigen DLL bzw. des dazugehörigen Shared Object. Dann wird eine Meldung wie die oben erwähnte zurückgegeben.

Schließen Sie zunächst alle JDBC-basierten Anwendungen und übernehmen Sie dann das Supportpaket erneut. Wenn das erneute Übernehmen des Supportpakets nicht funktioniert, versuchen Sie, das Problem mit einer der folgenden Methoden zu lösen:

- **Überprüfen Sie, ob das Installationsprogramm des Supportpakets DLLs und Shared Objects korrekt aktualisiert hat**
 - Unter Windows müssen Sie überprüfen, ob die Dateien *dbjdbc16.dll* und *dbjodbc16.dll* beim Übernehmen des Supportpakets korrekt aktualisiert wurden.
 - Unter Linux/Unix müssen Sie überprüfen, ob die Shared Objects *libdbjdbc16.so.1* und *libdbjodbc16.so.1* beim Übernehmen des Supportpakets korrekt aktualisiert wurden.
 - Unter Mac OS X müssen Sie überprüfen, ob die Shared Objects *libdbjdbc16.dylib* und *libdbjodbc16.dylib* beim Übernehmen des Supportpakets korrekt aktualisiert wurden.

Hinweis

Möglicherweise sind auf Ihrem System sowohl 32-Bit-Versionen als auch 64-Bit-Versionen der DLLs bzw. Shared Objects installiert. Sie müssen jeweils diejenigen DLLs bzw. Shared Objects überprüfen, die mit dem Bitwert der JAVA VM übereinstimmen, nicht diejenigen, die mit dem Bitwert des SQL Anywhere-Servers übereinstimmen.

- **Vergewissern Sie sich, dass nicht mehrere Kopien der DLLs oder Shared Objects auf Ihrem System vorhanden sind** Wenn die DLLs und Shared Objects korrekt aktualisiert wurden, vergewissern Sie sich, dass Sie nicht mehrere Kopien der DLLs oder Shared Objects mit demselben Bitwert haben. Dies kann vorkommen, wenn Sie die DLLs bzw. Shared Objects in die Erweiterungsordner der Java VM kopiert haben, um die Java-Einschränkung zu umgehen, die nicht zulässt, dass DLLs und Shared Objects in mehreren Klassenladern geladen werden.
- **Überprüfen Sie, ob die JAR-Dateien korrekt aktualisiert wurden** Wenn die DLLs bzw. Shared Objects korrekt aktualisiert wurden und nicht mehrere Kopien auf Ihrem System vorhanden sind, vergewissern Sie sich, dass die verschiedenen JAR-Dateien korrekt aktualisiert wurden. Um die einzelnen JAR-Dateien zu überprüfen, führen Sie die folgenden Befehle aus und vergewissern Sie sich, dass die von der JAR-Datei gemeldete Versions- und Build-Nummer mit der Versions- und Build-Nummer von SQL Anywhere für das installierte Supportpaket übereinstimmt.

Um die Datei *sajdbc4.jar* zu überprüfen, führen Sie den folgenden Befehl aus (wobei *path* durch den Pfad zu der JAR-Datei zu ersetzen ist):

```
java -cp path\sajdbc4.jar sybase.jdbc4.sqlanywhere.IBuildNum
```

Um die Datei *jodbc4.jar* zu überprüfen, führen Sie den folgenden Befehl aus (wobei *path* durch den Pfad zu der JAR-Datei zu ersetzen ist):

```
java -cp path\jodbc4.jar ianywhere.ml.jdbcodbc.jdbc4.IBuildNum
```

Wenn Sie ermittelt haben, welche JAR-Datei, welche DLL bzw. welches Shared Object nicht der Build-Nummer des Supportpakets entspricht, vergewissern Sie sich, dass die Datei nicht durch eine Anwendung gesperrt ist, und übernehmen Sie das Supportpaket erneut.

Fehlerbehandlung: Aggregatfunktionen und äußere Referenzen

SQL Anywhere befolgt die Standards von SQL/2008, um die Verwendung von Aggregatfunktionen zu regeln, wenn sie in einer Unterabfrage vorkommen. Diese Änderungen betreffen das Verhalten von Anweisungen, die für frühere Versionen der Software geschrieben wurden: Abfragen, die bisher gültig waren, können nun Fehlermeldungen erzeugen, und es kann sein, dass sich Ergebnismengen ändern.

Wenn eine Aggregatfunktion in einer Unterabfrage erscheint und die Spalte, die von der Aggregatfunktion referenziert wird, eine äußere Referenz darstellt, wird die gesamte Aggregatfunktion als äußere Referenz behandelt. Die Aggregatfunktion wird im äußeren Abfrageblock berechnet, nicht in der Unterabfrage, und wird dadurch zu einer Konstanten in der Unterabfrage.

Die folgenden Einschränkungen gelten für die Verwendung von Aggregatfunktionen mit äußerer Referenz in Unterabfragen:

- Die Aggregatfunktion mit äußerer Referenz kann nur in Unterabfragen erscheinen, die sich in der SELECT-Liste oder der HAVING-Klausel befinden, und diese Klauseln müssen immer im unmittelbar anschließenden äußeren Block enthalten sein.
- Aggregatfunktionen mit äußerer Referenz können nur eine äußere Spaltenreferenz enthalten.
- Lokale Spaltenreferenzen und äußere Spaltenreferenzen können in derselben Aggregatfunktion nicht gemischt verwendet werden.

Einige Probleme, die in Verbindung mit diesen neuen Standard auftauchen, können vermieden werden, indem Sie die Aggregatfunktion neu formulieren und darin nur lokale Referenzen verwenden. Die Unterabfrage (`SELECT MAX(S.y + R.y) FROM S`) beispielsweise enthält sowohl eine lokale (`S.y`) als auch eine äußere (`R.y`) Spaltenreferenz, was nun nicht mehr zulässig ist. Sie kann als (`SELECT MAX(S.y) + R.y FROM S`) umgeschrieben werden. In der Neuprogrammierung hat die Aggregatfunktion nur eine lokale Spaltenreferenz. Die gleiche Art der Neuprogrammierung kann angewendet werden, wenn eine Aggregatfunktion mit äußerer Referenz nicht in einer SELECT- oder einer HAVING-Klausel auftritt.

Beispiel 1

Die folgende Abfrage konnte in SQL Anywhere 7 und früheren Versionen gültige Ergebnisse liefern:

```
SELECT Name,  
       ( SELECT SUM( p.Quantity )  
         FROM SalesOrderItems )  
FROM Products p WHERE p.ID = 300;
```

Name	SUM(p.Quantity)
Tee shirt	30,716

In SQL Anywhere 8 und höheren Versionen erzeugt dieselbe Abfrage eine Fehlermeldung, wonach die Funktion oder Spaltenreferenz in einer GROUP BY-Klausel erscheinen muss. Die Anweisung ist nicht mehr gültig, weil `SUM(p.Quantity)` als äußere Referenz behandelt und im äußeren Abfrageblock berechnet wird. Die oben erwähnte Abfrage ist äquivalent zur folgenden Abfrage:

```
SELECT Name,  
       SUM( p.Quantity ) AS Z,  
       ( SELECT Z  
         FROM SalesOrderItems )  
FROM Products p WHERE p.ID = 300;
```

Da jedoch die Berechnung der Aggregatfunktion nun im äußeren Abfrageblock erfolgt, wird der äußere Abfrageblock als gruppierte Abfrage behandelt und der Spaltenname muss in der SELECT-Liste in einer GROUP BY-Klausel erscheinen. Deshalb ist diese Abfrage ebenfalls ungültig und erzeugt dieselbe Fehlermeldung. Wenn die zurückgegebene Ergebnismenge in SQL Anywhere 8 und höheren Versionen mit derjenigen übereinstimmen soll, die in SQL Anywhere 7 und früheren Versionen von der ersten Abfrage zurückgegeben wurde, müssen Sie die folgende Abfrage verwenden:

```
SELECT Name,
p.Quantity * ( SELECT COUNT( * ) FROM SalesOrderItems )
FROM Products p WHERE p.ID = 300;
```

Beispiel 2

In SQL Anywhere 7 und früheren Versionen lieferte die folgende Abfrage das Ergebnis 30 , 716, da SUM(p.Quantity) innerhalb des verschachtelten SELECT-Abfrageblocks berechnet und nur p.Quantity als äußere Referenz behandelt wurde.

```
SELECT ( SELECT FIRST SUM( p.Quantity ) FROM SalesOrderItems ) AS ss
FROM Products p WHERE p.ID = 300;
```

In SQL Anywhere 8 und späteren Versionen gibt dieselbe Abfrage das Ergebnis 28 zurück, da SUM(p.Quantity) als äußere Referenz behandelt und im äußeren Abfrageblock berechnet wird. Die oben erwähnte Abfrage ist also äquivalent zur folgenden Abfrage:

```
SELECT DT.ss
FROM ( SELECT SUM( p.Quantity ) AS asum,
      ( SELECT FIRST asum FROM SalesOrderItems ) AS ss
      FROM Products p WHERE p.ID = 300 ) AS DT;
```

Wenn die zurückgegebenen Ergebnisse in SQL Anywhere 8 und höheren Versionen mit denjenigen übereinstimmen sollen, die in SQL Anywhere 7 und früheren Versionen von der ersten Abfrage zurückgegeben wurden, müssen Sie die folgende Abfrage verwenden:

```
SELECT p.Quantity * ( SELECT COUNT( * ) FROM SalesOrderItems ) AS ss
FROM Products p WHERE p.ID = 300;
```

Siehe auch

- „Die HAVING-Klausel: Datengruppen auswählen“ [[SQL Anywhere Server - SQL-Benutzerhandbuch](#)]

MobiLink-Upgrades

Kompatibilität mit bestehender Software

- MobiLink-Clients der Version 16 sind nicht mit Versionen des MobiLink-Servers vor Version 16.0 kompatibel.
- Der MobiLink-Server der Version 16 kann zusammen mit Clients der Version 11 oder höher verwendet werden. Wenn Sie frühere Clients unterstützen müssen, behalten Sie für diesen Zweck eine frühere Version des MobiLink-Servers bei.
- Überprüfen Sie, ob die dokumentierten Verhaltensänderungen Ihre Anwendung betreffen. Wenn dies der Fall ist, aktualisieren Sie Ihre Anwendung. Siehe [SQL Anywhere 16 - Änderungen und Upgrades auf Seite 1](#).

Reihenfolge des Upgrades

Wenn Sie ein Upgrade einer vorhandenen MobiLink-Installation durchführen möchten, beachten Sie für die Komponenten die folgende Reihenfolge:

1. Fahren Sie die MobiLink-Server herunter.
2. Stellen Sie die konsolidierte Datenbank um.

Siehe „[Upgrades der konsolidierten Datenbank](#)“ auf Seite 326.

3. Führen Sie ein Upgrade der MobiLink-Server durch.

Siehe „[Upgrades von MobiLink-Server](#)“ auf Seite 334.

4. Starten Sie die MobiLink-Server.
5. Stellen Sie die MobiLink-Clients um.

Der MobiLink-Server der Version 16 kann nur zusammen mit Clients der Version 11 oder höher verwendet werden. Für MobiLink-Clients vor Version 11 ist ein Upgrade erforderlich, damit sie mit einem MobiLink-Server der Version 16 funktionieren.

Weitere Hinweise zu entfernten SQL Anywhere-Datenbanken finden Sie unter „[Upgrades von SQL Anywhere MobiLink-Clients](#)“ auf Seite 335.

6. Weitere Hinweise zu UltraLite-Anwendungen finden Sie unter „[UltraLite-Upgrades](#)“ auf Seite 337.

Überprüfen Sie vor dem Upgrade, ob Verhaltensänderungen die Datenbank betreffen, und ergreifen Sie die Standard-Vorsichtsmaßnahmen.

Siehe auch

- „[Verhaltensänderungen von MobiLink](#)“ auf Seite 49
- „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297

Upgrades der konsolidierten Datenbank

Bevor Sie den neuen MobiLink-Server mit einer vorhandenen konsolidierten Datenbank verwenden können, müssen Sie Upgradeskripten ausführen, die neue Systemobjekte installieren. Die Upgrade-Skripten müssen vom Eigentümer der aktuell installierten MobiLink-Systemtabellen ausgeführt werden. Sie können auch die folgenden Methoden verwenden, um die MobiLink-Systemeinrichtung zu aktualisieren:

- Klicken Sie im MobiLink-Plug-In für Sybase Central auf **MobiLink 16** **Projekt** **Konsolidierte Datenbanken**, rechtsklicken Sie auf den Datenbanknamen und klicken Sie auf **MobiLink-Systemkonfiguration prüfen**. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert.
- Wenn Sie den **Assistenten für das Deployment eines Synchronisationsmodells** verwenden, wird die Systemkonfiguration beim Verbinden mit der konsolidierten Datenbank überprüft. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert. Siehe „[Synchronisationsmodelle](#)“ [*MobiLink - Erste Orientierung*].

Die MobiLink-Upgradeskripten für 6.0.x wurden entfernt. Wenn Sie dieses Upgrade benötigen, wenden Sie sich an den technischen Support (<http://www.sybase.com/support>).

Hinweise

- Verwenden Sie die gespeicherte Prozedur `ml_add_missing_dnld_scripts`, um das Fehlen von `download_cursor`- und/oder `download_delete_cursor`-Skripten zu beheben. Wenn diese Prozedur mit einem Skriptversionsnamen aufgerufen wird, werden die fehlenden Skripten `download_cursor` und/oder `download_delete_cursor` für alle Synchronisationstabellen, die von der angegebenen Skriptversion verwendet werden, als ignorierte Skripten definiert.
- Wenn Sie `authenticate_user_hashed`-Skripten haben, die mit einer früheren Version als 10.0.0 erstellt wurden, ändern Sie diese mithilfe des Binäräquivalenttyps für Ihr RDBMS so, dass sie `VARBINARY(32)` statt `VARBINARY(20)` akzeptieren.

Upgrades für konsolidierte Datenbanken (SQL Anywhere 10.0.0 und höher)

Bevor Sie den neuen MobiLink-Server mit einer vorhandenen konsolidierten SQL Anywhere-Datenbank der Version 10.0.0 oder höher verwenden können, müssen Sie Upgradeskripten ausführen, die neue Systemobjekte installieren.

Voraussetzungen

Sie müssen Eigentümer der derzeit installierten MobiLink-Systemtabellen sein, um die Setupskripten ausführen zu können.

Das MONITOR-Systemprivileg ist erforderlich, um die Erkennungslogik für Sperren/Blockierungen aufrufen zu können.

Kontext und Bemerkungen

Sie können auch die folgenden Methoden verwenden, um die MobiLink-Systemeinrichtung zu aktualisieren:

- Klicken Sie im MobiLink-Plug-In für Sybase Central auf **MobiLink 16&rrarProjekt&rrarKonsolidierte Datenbanken**, rechtsklicken Sie auf den Datenbanknamen und klicken Sie auf **MobiLink-Systemkonfiguration prüfen**. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert.
- Wenn Sie den **Assistenten für das Deployment eines Synchronisationsmodells** verwenden, wird die Systemkonfiguration beim Verbinden mit der konsolidierten Datenbank überprüft. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert. Siehe „Synchronisationsmodelle“ [*MobiLink - Erste Orientierung*].

Aufgabe

1. Führen Sie ein Upgrade der SQL Anywhere-Software durch.

Siehe [Upgrade von Datenbanken der Version 10 und höher auf Seite 296](#).

2. Führen Sie ein Upgrade der MobiLink-Systemkonfiguration durch, indem Sie das Upgradeskript für die Version ausführen, von der das Upgrade ausgeht.

Das Upgrade-Skript heißt *upgrade_sa.sql*. Es befindet sich unter dem SQL Anywhere-Installationsverzeichnis im Verzeichnis *MobiLink\upgrade\version*, wobei *version* die SQL Anywhere-Version ist, von der Sie umstellen.

Beispiel: Stellen Sie in Interactive SQL eine Verbindung mit der Datenbank her und führen Sie die folgende Anweisung aus:

```
READ "C:\Program Files\SQL Anywhere 16\MobiLink\upgrade\10.0.x  
\upgrade_sa.sql"
```

Ergebnisse

Die konsolidierte Datenbank kann nun mit den neuen MobiLink-Server verwendet werden.

Upgrades für konsolidierte Datenbanken (Adaptive Server Enterprise, SAP Sybase IQ, Oracle, MySQL oder Microsoft SQL Server)

Bevor Sie den neuen MobiLink-Server mit einer vorhandenen konsolidierten Datenbank verwenden können, müssen Sie Upgradeskripten ausführen, die neue Systemobjekte installieren.

Voraussetzungen

Sie müssen Eigentümer der derzeit installierten MobiLink-Systemtabellen sein, um die Setupskripten ausführen zu können.

Bei ASE benötigt die Login-ID für den MobiLink-Server ein SELECT-Privileg für MASTER.SYSTRANSCTIONS.

Bei SAP Sybase IQ ist die EXECUTE-Berechtigung für SP_IQTRANSACTION erforderlich, damit der MobiLink-Server die Snapshot-Isolation für den Download verwenden kann.

Bei Oracle muss der RDBMS-Benutzer, den der MobiLink-Server zum Verbinden mit der konsolidierten Datenbank verwendet, berechtigt sein, die MobiLink-Systemtabellen, -Prozeduren usw. ohne Qualifizierer (z.B. SELECT * FROM ml_user) zu verwenden. Der RDBMS-Benutzer benötigt außerdem das SELECT-Privileg für GV\$TRANSACTION, GV\$SESSION, GV\$LOCK und DBA_OBJECTS sowie EXECUTE-Privilegien für DBMS_UTILITY. Sie können eine Berechtigung nicht direkt für die Synonyme GV\$TRANSACTION, GV\$SESSION und GV\$LOCK erteilen. Stattdessen müssen Sie die Berechtigung für die zugrunde liegenden dynamischen Performanceansichten GV_\$TRANSACTION, GV_\$SESSION und GV_\$LOCK erteilen. Dazu müssen Sie eine Verbindung als SYS herstellen.

Bei Microsoft SQL Server muss der RDBMS-Benutzer, den der MobiLink-Server zum Verbinden mit der konsolidierten Datenbank verwendet, die VIEW SERVER STATE-Berechtigung haben sowie die SELECT-Berechtigung für SYS.DATABASES, SYS.DM_TRAN_LOCKS, SYS.PARTITIONS und SYS.SYSPROCESSES.

Kontext und Bemerkungen

Für die MobiLink-Systemobjekte in Ihrer konsolidierten Adaptive Server Enterprise-, Oracle-, MySQL- oder Microsoft SQL Server-Datenbank ist nur dann ein Upgrade erforderlich, wenn Ihre MobiLink-Serverversion älter ist als Version 16.0.

Sie können auch die folgenden Methoden verwenden, um die MobiLink-Systemeinrichtung zu aktualisieren:

- Klicken Sie im MobiLink-Plug-In für Sybase Central auf **MobiLink 16** **Projekt** **Konsolidierte Datenbanken**, rechtsklicken Sie auf den Datenbanknamen und klicken Sie auf **MobiLink-Systemkonfiguration prüfen**. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert.
- Wenn Sie den **Assistenten für das Deployment eines Synchronisationsmodells** verwenden, wird die Systemkonfiguration beim Verbinden mit der konsolidierten Datenbank überprüft. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert. Siehe „Synchronisationsmodelle“ [[MobiLink - Erste Orientierung](#)].

Aufgabe

1. Setzen Sie bei Adaptive Server Enterprise-Datenbanken die SELECT INTO-Datenbankoption. Führen Sie die folgende Anweisung in Sybase Interactive SQL aus:

```
USE MASTER
go
sp_dboption your-database-name, "SELECT INTO", true
go
USE your-database-name
go
checkpoint
go
```

2. Führen Sie das zutreffende Upgrade-Skript für die Version aus, von der Sie umstellen.

Die Upgrade-Skripten haben den Namen *upgrade_XXX.sql*, wobei *XXX* das RDBMS der konsolidierten Datenbank angibt. Sie befinden sich im SQL Anywhere-Installationsverzeichnis im Verzeichnis *MobiLink\upgrade\version*, wobei *version* die MobiLink-Version ist, von der Sie umstellen.

Beispiel: Für ein Upgrade einer Microsoft SQL Server-Datenbank, in die die MobiLink-Systemtabellen aus Version 9.0.2 übernommen wurden, führen Sie folgenden Befehl aus:

```
osql -S server_name -U user_name -P password -i
"C:\Program Files\SQL Anywhere 16\MobiLink\upgrade\9.0.2\upgrade_mss.sql"
```

Ergebnisse

Die konsolidierte Datenbank kann nun mit den neuen MobiLink-Server verwendet werden.

Upgrades für konsolidierte IBM DB2 LUW-Datenbanken

Bevor Sie den neuen MobiLink-Server mit einer vorhandenen konsolidierten IBM DB2 LUW-Datenbank verwenden können, müssen Sie Upgradeskripten ausführen, die neue Systemobjekte installieren.

Voraussetzungen

Sie müssen Eigentümer der derzeit installierten MobiLink-Systemtabellen sein, um die Setupskripten ausführen zu können.

Bei DB2 LUW 10.1 benötigt der MobiLink-Server die SELECT-Berechtigung für SYSIBMADM.MON_LOCKWAITS und SNAPSHOT_APPL_INFO, um die Erkennungslogik für Sperren/Blockierungen aufrufen zu können.

Bei DB2 LUW 9 benötigt der MobiLink-Server die SELECT-Berechtigung für SYSIBMADM.LOCKWAITS und SNAPSHOT_APPL_INFO, um die Erkennungslogik für Sperren/Blockierungen aufrufen zu können.

Kontext und Bemerkungen

Ein Upgrade Ihrer konsolidierten IBM DB2 LUW-Datenbank ist nur erforderlich, wenn Ihre MobiLink-Serverversion älter ist als Version 16.0.

Weitere Hinweise dazu, wie Sie das IBM DB2 LUW-Setupskript ausführen, finden Sie unter „Konsolidierte IBM DB2 LUW-Datenbank“ [[MobiLink - Serveradministration](#)].

Sie können auch die folgenden Methoden verwenden, um die MobiLink-Systemeinrichtung zu aktualisieren:

- Klicken Sie im MobiLink-Plug-In für Sybase Central auf **MobiLink 16** und **Projekt Konsolidierte Datenbanken**, rechtsklicken Sie auf den Datenbanknamen und klicken Sie auf **MobiLink-Systemkonfiguration prüfen**. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert.
- Wenn Sie den **Assistenten für das Deployment eines Synchronisationsmodells** verwenden, wird die Systemkonfiguration beim Verbinden mit der konsolidierten Datenbank überprüft. Wenn die Datenbank ein Setup oder Upgrade erfordert, werden Sie zum Fortfahren aufgefordert. Siehe „Synchronisationsmodelle“ [[MobiLink - Erste Orientierung](#)].

Aufgabe

1. Suchen Sie das IBM DB2 LUW-Upgrade-Skript.

Das Upgrade-Skript hat den Namen *upgrade_db2.sql* und befindet sich im SQL Anywhere-Installationsverzeichnis im Unterverzeichnis *MobiLink/upgrade/version*. Das Verzeichnis *version* bezieht sich auf die MobiLink-Version, von der aus Sie das Upgrade vornehmen.

2. Kopieren Sie *upgrade_db2.sql* und bearbeiten Sie die Kopie. Ändern Sie die Anweisung CONNECT oben im Skript, sodass sie für die Instanz geeignet ist, mit der Sie sich verbinden wollen. Führen Sie das kopierte SQL-Skript für die konsolidierte Datenbank aus.

Ergebnisse

Die konsolidierte Datenbank kann nun mit den neuen MobiLink-Server verwendet werden.

Upgrades für konsolidierte Datenbanken (SQL Anywhere vor Version 10.0.0)

Bevor Sie den neuen MobiLink-Server mit einer vorhandenen konsolidierten SQL Anywhere-Datenbank vor Version 10.0.0 verwenden können, müssen Sie Upgradeskripten ausführen, die neue Systemobjekte installieren.

Voraussetzungen

Wenn Sie eine konsolidierte SQL Anywhere-Datenbank einrichten, aber nie mit ihr synchronisieren, müssen Sie das Setupskript ausführen (nicht das Upgrade-Skript). Dieser Schritt bezieht sich nur auf konsolidierte SQL Anywhere-Datenbanken. Siehe „Konsolidierte SQL Anywhere-Datenbank“ [[MobiLink - Serveradministration](#)].

Kontext und Bemerkungen

- Vor SQL Anywhere Version 10.0.0 gehörten MobiLink-Systemtabellen dem dbo. Um die Setupskripten für eine SQL Anywhere-Datenbank auszuführen, müssen Sie an der konsolidierten Datenbank als Eigentümer der MobiLink-Systemtabellen angemeldet sein. Es ist nicht ausreichend, diese Skripten als Benutzer mit der Berechtigung zum Ändern der Tabellen auszuführen. Zum Ausführen der Upgrade-Skripten können Sie mit der SQL-Anweisung SETUSER den Benutzer "dbo" impersonieren. Beispiel:

```
SETUSER "dbo" ;
```

Um das Upgrade einer konsolidierten Datenbank in Sybase Central auszuführen, sollten Sie mit der Anweisung GRANT CONNECT ein dbo-Kennwort erstellen und sich dann als dbo verbinden. Beispiel:

```
GRANT CONNECT TO dbo IDENTIFIED BY password;
```

Im zweiten Fall sollten Sie nach dem Upgrade ALTER USER verwenden, um das dbo-Kennwort zu entfernen. Beispiel:

```
ALTER USER TO dbo IDENTIFIED BY "";
```

Aufgabe

1. Wenn Sie ein Upgrade für eine konsolidierte Datenbank einer früheren SQL Anywhere-Version als 10.0.0 ausführen, müssen Sie zunächst die Datenbank auf Version 16 umstellen:
 - a. Fahren Sie den Datenbankserver herunter.
 - b. Führen Sie ein Upgrade der Datenbank auf Version 16 aus.

Entsprechende Anweisungen finden Sie unter:

 - „Neuaufbauprozess für Datenbanken der Version 9 und höher“ auf Seite 303

- c. Melden Sie sich als DBA an und starten Sie den Datenbankserver.

Melden Sie sich als DBA an, um das Upgrade durchzuführen.

2. Führen Sie das zutreffende Upgrade-Skript für die Version aus, von der Sie umstellen.

Das Upgrade-Skript heißt *upgrade_asa.sql*. Es befindet sich unter dem SQL Anywhere-Installationsverzeichnis im Verzeichnis *MobiLink\upgrade\version*, wobei *version* die SQL Anywhere-Version ist, von der Sie umstellen.

Zum Ausführen der Upgrade-Skripten können Sie den Benutzer "dbo" mithilfe der SQL-Anweisung SETUSER impersonieren.

Beispiel: Um eine konsolidierte SQL Anywhere-Datenbank der Version 9.0.2 umzustellen, verbinden Sie sich mit der Datenbank in Interactive SQL und führen die folgende Anweisung aus:

```
SETUSER "dbo";
READ 'C:\Program Files\SQL Anywhere 16\MobiLink\upgrade
\9.0.2\upgrade_asa.sql'
```

3. Entfernen Sie das dbo-Kennwort. Beispiel:

```
GRANT CONNECT TO "dbo";
```

4. Wenn Sie den MobiLink-Server nicht als DBA-Benutzer ausführen, müssen Sie dem betreffenden Benutzer die EXECUTE-Berechtigung für die neuen MobiLink-Systemobjekte erteilen. Welche Systemobjekte neu sind, hängt davon ab, von welcher Version Sie umstellen. Der folgende Code erteilt die erforderlichen Berechtigungen für alle MobiLink-Systemobjekte. Bevor Sie den Code ausführen, müssen Sie den Benutzernamen "my_user" in den Namen des Benutzers ändern, der den MobiLink-Server ausführt.

```
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_column to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_connection_script to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_database to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_device to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_device_address to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_listening to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough_repair to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough_script to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough_status to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_primary_server to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_property to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_delivery to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_delivery_archive to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_global_props to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_notifications to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_archive to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_props to
my_user;
```

```

GRANT SELECT, INSERT, UPDATE, DELETE ON
dbo.ml_qa_repository_props_archive to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_staging to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_status_history to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_status_history_archive
to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_status_staging to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_agent to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_agent_property to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_agent_staging to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_deployed_task to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_event to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_event_staging to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_managed_remote to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_notify to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_remote_db_class to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_task to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_task_command to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_task_command_property
to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_ra_task_property to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_script to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_script_version to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_scripts_modified to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_sis_sync_state to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_subscription to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_table to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_table_script to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_user to my_user;
GRANT EXECUTE ON dbo.ml_add_column to my_user;
GRANT EXECUTE ON dbo.ml_add_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_dnet_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_dnet_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_java_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_java_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_conn_script_chk to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_table_script_chk to my_user;
GRANT EXECUTE ON dbo.ml_add_missing_dnld_scripts;
GRANT EXECUTE ON dbo.ml_add_passthrough to my_user;
GRANT EXECUTE ON dbo.ml_add_passthrough_repair to my_user;
GRANT EXECUTE ON dbo.ml_add_passthrough_script to my_user;
GRANT EXECUTE ON dbo.ml_add_property to my_user;
GRANT EXECUTE ON dbo.ml_add_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_user to my_user;
GRANT EXECUTE ON dbo.ml_delete_device to my_user;
GRANT EXECUTE ON dbo.ml_delete_device_address to my_user;
GRANT EXECUTE ON dbo.ml_delete_listening to my_user;
GRANT EXECUTE ON dbo.ml_delete_passthrough to my_user;
GRANT EXECUTE ON dbo.ml_delete_passthrough_repair to my_user;
GRANT EXECUTE ON dbo.ml_delete_passthrough_script to my_user;
GRANT EXECUTE ON dbo.ml_delete_remote_id to my_user;

```

```
GRANT EXECUTE ON dbo.ml_delete_sync_state to my_user;
GRANT EXECUTE ON dbo.ml_delete_sync_state_before to my_user;
GRANT EXECUTE ON dbo.ml_delete_user to my_user;
GRANT EXECUTE ON dbo.ml_delete_user_state to my_user;
GRANT EXECUTE ON dbo.ml_lock_rid to my_user;
GRANT EXECUTE ON dbo.ml_qa_add_delivery to my_user;
GRANT EXECUTE ON dbo.ml_qa_add_message to my_user;
GRANT EXECUTE ON dbo.ml_qa_handle_error to my_user;
GRANT EXECUTE ON dbo.ml_qa_stage_status_from_client to my_user;
GRANT EXECUTE ON dbo.ml_qa_staged_status_for_client to my_user;
GRANT EXECUTE ON dbo.ml_qa_upsert_global_prop to my_user;
GRANT EXECUTE ON dbo.ml_ra_add_agent_id to my_user;
GRANT EXECUTE ON dbo.ml_ra_assign_task to my_user;
GRANT EXECUTE ON dbo.ml_ra_cancel_notification to my_user;
GRANT EXECUTE ON dbo.ml_ra_cancel_task_instance to my_user;
GRANT EXECUTE ON dbo.ml_ra_clone_agent_properties to my_user;
GRANT EXECUTE ON dbo.ml_ra_delete_agent_id to my_user;
GRANT EXECUTE ON dbo.ml_ra_delete_events_before to my_user;
GRANT EXECUTE ON dbo.ml_ra_delete_task to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_agent_events to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_agent_ids to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_agent_properties to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_latest_event_id to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_orphan_taskdbs to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_remote_ids to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_task_results to my_user;
GRANT EXECUTE ON dbo.ml_ra_get_task_status to my_user;
GRANT EXECUTE ON dbo.ml_ra_int_cancel_notification to my_user;
GRANT EXECUTE ON dbo.ml_ra_int_move_events to my_user;
GRANT EXECUTE ON dbo.ml_ra_manage_remote_db;
GRANT EXECUTE ON dbo.ml_ra_notify_agent_sync to my_user;
GRANT EXECUTE ON dbo.ml_ra_notify_task to my_user;
GRANT EXECUTE ON dbo.ml_ra_reassign_taskdb to my_user;
GRANT EXECUTE ON dbo.ml_ra_set_agent_property to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_agent_auth_file_xfer to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_download_ack to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_download_prop to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_download_remote_dbs to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_download_task to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_download_task2;
GRANT EXECUTE ON dbo.ml_ra_ss_download_task_cmd to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_end_upload to my_user;
GRANT EXECUTE ON dbo.ml_ra_ss_upload_prop to my_user;
GRANT EXECUTE ON dbo.ml_ra_unmanage_remote_id to my_user;
GRANT EXECUTE ON dbo.ml_reset_sync_state to my_user;
GRANT EXECUTE ON dbo.ml_set_device to my_user;
GRANT EXECUTE ON dbo.ml_set_device_address to my_user;
GRANT EXECUTE ON dbo.ml_set_listening to my_user;
GRANT EXECUTE ON dbo.ml_set_sis_sync_state to my_user;
GRANT EXECUTE ON dbo.ml_upload_update_device_address to my_user;
GRANT EXECUTE ON dbo.ml_upload_update_listening to my_user;
```

Ergebnisse

Die konsolidierte Datenbank kann nun mit den neuen MobiLink-Server verwendet werden.

Upgrades von MobiLink-Server

Führen Sie ein Upgrade des MobiLink-Servers auf Version 16 durch, wenn Sie entfernte Datenbanken der Version 16 synchronisieren möchten.

Überprüfen Sie, ob die Verhaltensänderungen Ihre Anwendung betreffen, bevor Sie die Version 16 des MobiLink-Servers verwenden. Siehe [SQL Anywhere 16 - Änderungen und Upgrades auf Seite 1](#).

Die Version 16 des MobiLink-Servers unterstützt nur SQL Anywhere- und UltraLite-Clients der Versionen 10, 11 und 12. Wenn Sie frühere Clients unterstützen müssen, behalten Sie für diesen Zweck eine frühere Version des MobiLink-Servers bei.

Upgrades von SQL Anywhere MobiLink-Clients

In einer Produktionsumgebung sollten Sie entfernte SQL Anywhere-Datenbanken nur umstellen, nachdem Sie sowohl die konsolidierte Datenbank als auch den MobiLink-Server umgestellt haben.

In Version 10.0.0 wurde Adaptive Server Anywhere in SQL Anywhere umbenannt.

Es müssen mehrere Arten von Upgrades berücksichtigt werden:

- Upgrade der Software.
- Upgrade der entfernten Datenbank selbst
- Upgrade der gesamten Anwendung

Upgrade der Software

Es wird empfohlen, dbmlsync und den SQL Anywhere-Datenbankserver gleichzeitig umzustellen. MobiLink-Clients der Version 16 können nur mit Datenbanken der Version 16 synchronisiert werden, die auf Datenbankservern der Version 16 laufen.

MobiLink-Clients der Version 16 erfordern einen MobiLink-Server der Version 16 oder höher für die Synchronisation. MobiLink-Clients der Version 16 können nicht mit älteren MobiLink-Servern als Version 16 synchronisiert werden.

Weitere Hinweise zum Upgrade von MobiLink finden Sie unter „[MobiLink-Upgrades](#)“ auf Seite 325.

Upgrade entfernter SQL Anywhere-Datenbanken

Sie können entfernte MobiLink SQL Anywhere-Datenbanken unter Verwendung der Prozeduren für SQL Anywhere-Datenbanken umstellen. Entsprechende Anweisungen finden Sie unter „[SQL Anywhere-Server-Upgrades](#)“ auf Seite 295.

Wenn eine Schemaänderung oder eine andere wichtige Änderung der Datenbank vorgenommen wurde, müssen Sie die Daten möglicherweise manuell entladen und neu laden. Siehe „[Entfernte SQL Anywhere-Datenbanken manuell entladen/neu laden](#)“ auf Seite 336.

Upgrade von Anwendungen

Wenn Sie für eine neue Version einer MobiLink-Anwendung ein Deployment durchführen, sollten Sie für die Synchronisationsskripten eine neue Skriptversion verwenden. Wenn die vorhandene Anwendung ein Skript namens **v1** verwendet, könnte die umgestellte Anwendung z.B. eine Skriptversion namens **v2** verwenden. Beide Skriptversionen können gleichzeitig verwendet werden. Daher ist es einfacher, ein Upgrade der entfernten Datenbank inkrementell statt in einem Schritt durchzuführen.

Für Version 9.0.0 und höher wurde die MobiLink-Serveroption -zd entfernt. Wenn Sie in Ihrem Deployment die Option -zd verwenden und ein Upgrade durchführen möchten, müssen Sie Ihre Downloadskripten so ändern, dass sie den Zeitstempel des letzten Downloads als ersten Parameter akzeptieren. Alternativ können Sie ein Upgrade Ihres Clients durchführen und ihn unter Verwendung von benannten Parametern starten. Damit können Sie Skriptparameter in beliebiger Reihenfolge anordnen.

Entfernte SQL Anywhere-Datenbanken manuell entladen/neu laden

Wenn eine Schemaänderung oder eine andere wichtige Änderung der Datenbank vorgenommen wurde, müssen Sie die Daten möglicherweise manuell entladen und neu laden.

Voraussetzungen

Für Datenbanken der Version 16 müssen Sie die folgenden Privilegien haben:

- BACKUP DATABASE
- VALIDATE ANY OBJECT
- SERVER OPERATOR
- SELECT ANY TABLE

Befolgen Sie die Standard-Vorsichtsmaßnahmen für ein Software-Upgrade. Siehe „[Vorsichtsmaßnahmen für Upgrade und Neuaufbau](#)“ auf Seite 297.

Aufgabe

1. Stoppen Sie jegliche Datenbankaktivität.
2. Führen Sie eine erfolgreiche Synchronisation aus, validieren Sie die entfernte Datenbank und sichern Sie sie.
3. Führen Sie das Dienstprogramm dbtran aus, um das Start-Offset und End-Offset des Datenbank-Transaktionslogs anzuzeigen. Notieren Sie sich das End-Offset.

Siehe „[Dienstprogramm zur Logkonvertierung \(dbtran\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

4. Benennen Sie das Transaktionslog um, damit es während des Entladens nicht geändert wird. Verschieben Sie die umbenannte Logdatei an einen sicheren Speicherort, z.B. in ein Offline-Verzeichnis.
5. Entladen Sie die Datenbank, ohne die mit -a beginnenden Parameter von dbunload zu verwenden, damit die Datenbank automatisch in eine andere Datenbank geladen wird.

Siehe „[Dienstprogramm zum Entladen \(dbunload\)](#)“ [*SQL Anywhere Server - Datenbankadministration*].

6. Initialisieren Sie eine neue Datenbank.

Siehe „Dienstprogramm Initialisierung (dbinit)“ [[SQL Anywhere Server - Datenbankadministration](#)].

7. Laden Sie die Daten in die neue Datenbank, indem Sie mit dbisql die von dbunload generierte Datei **reload.sql** lesen.

Siehe „Interactive SQL-Dienstprogramm (dbisql)“ [[SQL Anywhere Server - Datenbankadministration](#)].

8. Fahren Sie die neue Datenbank herunter.
9. Löschen Sie das Transaktionslog der neuen Datenbank.

10. Führen Sie dblog mit den folgenden Optionen für die neue Datenbank aus:

- Verwenden Sie -z, um das zuvor notierte End-Offset anzugeben.
- Verwenden Sie -x, um das relative Offset auf Null zu setzen.

Beispiel:

```
dblog -x 0 -z 137829 database-name.db
```

Siehe „Transaktionslog-Dienstprogramm (dblog)“ [[SQL Anywhere Server - Datenbankadministration](#)].

11. Starten Sie dbmlsync. Geben Sie dabei den Speicherort der ursprünglichen Logdatei an, die Sie zuvor verschoben haben.

Siehe „dbmlsync-Syntax“ [[MobiLink - Clientadministration](#)].

12. Wenn Sie die alte Logdatei nicht mehr benötigen, verwenden Sie die Datenbankoption `delete_old_logs`.

Siehe „delete_old_logs-Option [SQL Remote]“ [[SQL Anywhere Server - Datenbankadministration](#)].

Ergebnisse

Die entfernte SQL Anywhere-Datenbank wird entladen und neu geladen.

UltraLite-Upgrades

Berücksichtigen Sie die folgenden Hinweise, bevor Sie ein Upgrade auf eine UltraLite-Datenbank der Version 16 durchführen:

- Die UltraLite-Version 16 kann keine UltraLite-Datenbanken lesen, die mit früheren Versionen der Software erstellt wurden.
- Sie können Upgrades von Datenbanken nicht auf Geräten durchführen.

- Nachdem das Upgrade der Datenbank durchgeführt wurde, kann sie sich nicht mehr mit Anwendungen, Dienstprogrammen und Software früherer Versionen verbinden.
- Wenn auf Ihrem Computer mehrere Versionen von SQL Anywhere installiert sind, achten Sie auf Ihren Systempfad, um sicherzustellen, dass Sie die entsprechenden Dienstprogramme verwenden. UltraLite-Dienstprogramme haben unabhängig von der Version identische Namen und können nur anhand des Verzeichnispfadnamens unterschieden werden.

Bevor Sie vorhandene Anwendungen mit dieser Version der Software verwenden, sollten Sie die Liste der neuen Funktionen und Verhaltensänderungen überprüfen, um festzustellen, ob Auswirkungen auf Ihre Anwendung zu erwarten sind.

Upgrade von Datenbanken bis Version 12

UltraLite-Datenbanken der Version 12 und früher müssen in eine SQL- oder XML-Datei entladen werden, und zwar mit der Version der Software, die zum Erstellen der Datenbank verwendet wurde. Verwenden Sie das Dienstprogramm ulunload, das zum Lieferumfang dieser Software gehört.

Wenn die ältere Version der Software nicht installiert ist und Sie mit einem Windows-Desktopbetriebssystem arbeiten, können Sie eines der älteren ulunload-Dienstprogramme verwenden, die in SQL Anywhere 16 enthalten sind. Wählen Sie das Dienstprogramm ulunload, das der Version der UltraLite-Datenbank entspricht.

Sobald die Datenbank entladen ist, können Sie das Dienstprogramm ulunload der Version 16 verwenden, um die XML-Datei in eine Datenbank der Version 16 zu laden, bzw. das Dienstprogramm dbisql, wenn Sie die Datenbank in eine SQL-Datei entladen haben.

Upgrade von Java Edition-Datenbanken der Version 12.0.1

Generieren Sie mit dem Dienstprogramm ULjUnload der UltraLite Java Edition-Version 12.0.1 eine XML-Datei, die vom Dienstprogramm uljload der Version 16 verwendet werden kann, um eine neue UltraLite Java Edition-Datenbank zu erstellen.

Kompatibilität mit bestehender Software

- UltraLite-Datenbankdateien der Version 16 unterstützen nur Verbindungen von Clientanwendungen der Version 16 oder der UltraLite Engine der Version 16.
- Die UltraLite-Laufzeitumgebung der Version 16 und die UltraLite-Engine der Version 16 funktionieren nicht mit Datenbankdateien und Anwendungscode, die mit UltraLite bis Version 9 erstellt wurden.

Siehe auch

- [„Neue Funktionen von UltraLite“ auf Seite 270](#)
- [„UltraLite-Verhaltensänderungen und nicht mehr empfohlene Funktionen“ auf Seite 275](#)
- [„Sicherstellen, dass Sie die richtige Version der Dienstprogramme ausführen, wenn mehrere Versionen installiert sind“ auf Seite 298](#)

Upgrade von UltraLite-Datenbanken (Windows)

Für Benutzer von früheren Versionen der Software fasst die folgende Aufgabe den Prozess zum Neuaufbau der UltraLite-Datenbank in Version 16 zusammen.

Voraussetzungen

- Wenn Sie ein Upgrade von einer UltraLite-Datenbank der Version 9 oder früher durchführen, müssen Sie die betreffende Version von SQL Anywhere installiert haben.
- Erstellen Sie eine Sicherungskopie der vorhandenen UltraLite-Datenbank.
- Synchronisieren Sie die Datenbank, wenn es sich um eine Produktionsdatenbank handelt, die möglicherweise nicht synchronisierte Änderungen enthält.

Kontext und Bemerkungen

Die UltraLite-Version 16 kann keine UltraLite-Datenbanken lesen, die mit einer früheren Version der Software erstellt wurden.

Aufgabe

1. Erstellen Sie eine XML-Datei (bzw. mehrere Dateien) mit dem Inhalt der Datenbank.

Öffnen Sie eine Eingabeaufforderung, navigieren Sie zum Verzeichnis der früheren SQL Anywhere-Installation und wenden Sie das Dienstprogramm ulunload auf die Datenbank an.

Wenn die UltraLite-Datenbank mit der Version 10, 11 oder 12 von SQL Anywhere erstellt wurde, können Sie alternativ eine Eingabeaufforderung öffnen, zum Verzeichnis `%SQLANY16%\UltraLite\Unload\` navigieren, das für Ihre Version der UltraLite-Datenbank zutreffende Verzeichnis wählen, z.B. `V10`, `V11` oder `V12`, und anschließend das Dienstprogramm ulunload aus diesem Verzeichnis anwenden.

2. Verwenden Sie das Dienstprogramm ulload der UltraLite-Version 16 oder den **Assistenten zum Laden einer Datenbank** des **UltraLite 16**-Plug-Ins für Sybase Central, um das Schema und die Daten in eine neue Datenbank der Version 16 zu laden.

UltraLite-Datenbanken verwenden nun standardmäßig die UTF-8-Kodierung. Falls diese Kodierung nicht Ihren Anforderungen entspricht, setzen Sie den `utf8_encoding`-Parameter explizit auf "off".

3. Überprüfen Sie in der generierten XML-Datei die Einstellung für die UTF-8-Kodierung.

Ergebnisse

Das Upgrade der UltraLite-Datenbank auf die neueste Version wird durchgeführt.

Siehe auch

- „UltraLite-Erstellungsparameter utf8_encoding“ [*UltraLite - Datenbankverwaltung*]
- „UltraLite-Dienstprogramm zum Entladen von Datenbanken (ulunload)“ [*UltraLite - Datenbankverwaltung*]
- „UltraLite-Dienstprogramm zum Laden von Daten aus XML-Dateien (ulload)“ [*UltraLite - Datenbankverwaltung*]

Upgrade von UltraLite-Datenbanken (Linux)

Für Benutzer von früheren Versionen der Software fasst die folgende Aufgabe den Prozess zum Neuaufbau der Datenbank in Version 16 zusammen.

Voraussetzungen

- Installieren Sie die Version von SQL Anywhere, mit der die Datenbank erstellt wurde.
- Erstellen Sie eine Sicherungskopie der vorhandenen UltraLite-Datenbank.
- Synchronisieren Sie die Datenbank, wenn es sich um eine Produktionsdatenbank handelt, die möglicherweise nicht synchronisierte Änderungen enthält.

Kontext und Bemerkungen

Die UltraLite-Version 16 kann keine UltraLite-Datenbanken lesen, die mit einer früheren Version der Software erstellt wurden.

Aufgabe

1. Erstellen Sie eine XML-Datei (bzw. mehrere Dateien) mit dem Inhalt der Datenbank.

Navigieren Sie an einer Eingabeaufforderung zum Verzeichnis `%SQLANY%/bin32`, wobei `%SQLANY%` der Speicherort der früheren SQL Anywhere-Installation ist, und wenden Sie das Dienstprogramm `ulunload` auf die Datenbanken an.

2. Verwenden Sie das Dienstprogramm `ulload` der UltraLite-Version 16 oder den **Assistenten zum Laden einer Datenbank** des **UltraLite 16**-Plug-Ins für Sybase Central, um das Schema und die Daten in eine neue Datenbank der Version 16 zu laden.

UltraLite-Datenbanken verwenden nun standardmäßig die UTF-8-Kodierung. Falls diese Kodierung nicht Ihren Anforderungen entspricht, setzen Sie den `utf8_encoding`-Parameter explizit auf "off".

3. Überprüfen Sie in der generierten XML-Datei die Einstellung für die UTF-8-Kodierung.

Ergebnisse

Das Upgrade der UltraLite-Datenbank auf die neueste Version wird durchgeführt.

Siehe auch

- „UltraLite-Erstellungsparameter utf8_encoding“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Dienstprogramm zum Entladen von Datenbanken (ulunload)“ [[UltraLite - Datenbankverwaltung](#)]
- „UltraLite-Dienstprogramm zum Laden von Daten aus XML-Dateien (ulload)“ [[UltraLite - Datenbankverwaltung](#)]

SQL Remote-Upgrades

SQL Remote-Installationen umfassen eine konsolidierte Datenbank und viele entfernte Datenbanken sowie an jedem Standort einen SQL Remote-Nachrichtenagenten.

An jedem Standort verwaltet der SQL Remote-Nachrichtenagent das Senden und Empfangen von Nachrichten. Die Nachrichten haben die Form von SQL-Anweisungen, und der Datenbankserver wickelt die praktische Durchführung dieser SQL-Anweisungen ab.

Die Upgrade-Anforderungen für SQL Remote sind wie folgt:

- **Software-Upgrades können jeweils an einem Standort vorgenommen werden** Ältere Nachrichtenagenten (dbremote) können Nachrichten mit Nachrichtenagenten der Version 16 austauschen. Bei SQL Remote Version 5 können Nachrichtenagenten der Version 5 Nachrichten mit Nachrichtenagenten der Version 16 austauschen, wenn die compression-Datenbankoption auf -1 (minus Eins) gesetzt ist. Es besteht keine Notwendigkeit, ein Software-Upgrade in der gesamten Installation gleichzeitig durchzuführen. Siehe „[compression-Option \[SQL Remote\]](#)“ [[SQL Anywhere Server - Datenbankadministration](#)].
- **Datenbank-Upgrade durchführen** Wenn Sie ein Upgrade einer entfernten oder konsolidierten Datenbank vornehmen, die SQL Anywhere Version 9 oder früher verwendet, müssen Sie ein Upgrade des Datenbankdateiformats vornehmen, indem Sie Ihre Datenbank entladen und neu laden. Es besteht keine Notwendigkeit, ein Upgrade aller Datenbanken gleichzeitig vorzunehmen.

Anweisungen zum Entladen und Neuladen der Datenbank finden Sie unter „[Neuaufbauprozess für Datenbanken der Version 9 und höher](#)“ auf Seite 303.

- **Konsolidierte Adaptive Server Enterprise-Datenbanken umstellen** SQL Remote unterstützt keine konsolidierten Adaptive Server Enterprise-Datenbanken mehr. Um Adaptive Server Enterprise-Datenbanken zu synchronisieren, müssen Sie auf MobiLink umstellen.

Weitere Hinweise zum Migrieren von SQL Remote zu MobiLink finden Sie unter <http://www.sybase.com/detail?id=1034174>.

- **Upgrade mit SQL Remote** Die Unterstützung für die Nachrichtensysteme VIM und MAPI für SQL Remote wurde in Version 11.0.0 entfernt. Wenn Sie eine Datenbank, die VIM oder MAPI verwendet, auf SQL Anywhere Version 16 umstellen, müssen Sie den Nachrichtentyp auf File, FTP oder SMTP ändern. Wenn der Nachrichtentyp MAPI oder VIM ist, kann *dbremote.exe* nicht gestartet werden.

Beispiel

Ein Ansatz für ein Upgrade von Version 5 von SQL Remote sieht wie folgt aus:

1. Führen Sie ein Upgrade des konsolidierten Datenbankservers und des SQL Remote-Nachrichtenagenten und anschließend ein Upgrade der Datenbankdatei aus, indem Sie die konsolidierte Datenbank entladen und neu laden. Setzen Sie die Datenbankoption "compression" auf -1, sodass alle Nachrichten mit der Software der Version 5 an den entfernten Standorten kompatibel sind. Hinweise zum Entladen und Neuladen der konsolidierten Datenbank finden Sie unter „[An Replikations- oder Synchronisationssystemen beteiligte Datenbanken neu aufbauen \(Befehlszeile\)](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
2. Führen Sie jeweils ein Upgrade der entfernten Datenbankserver und der Nachrichtenagenten durch und anschließend ein Upgrade des Datenbankdateiformats, indem Sie die entfernten Datenbanken entladen und neu laden. Sie können die Datenbankoption "compression" wieder auf einen anderen Wert als -1 setzen, um die Komprimierungs- und Kodierungsmöglichkeiten für die Nachrichten zu nutzen, die an den konsolidierten Datenbankserver geschickt werden. Hinweise zum Entladen und Neuladen der entfernten Datenbanken finden Sie unter „[An Replikations- oder Synchronisationssystemen beteiligte Datenbanken neu aufbauen \(Befehlszeile\)](#)“ [*SQL Anywhere Server - SQL-Benutzerhandbuch*].
3. Wenn das Upgrade aller entfernten Datenbankserver und Nachrichtenagenten abgeschlossen ist, setzen Sie die Datenbankoption "compression" auf einen anderen Wert als -1.

Upgrade des SQL Anywhere-Monitors durchführen und Ressourcen migrieren

Verwenden Sie das Migrator-Dienstprogramm, um die Ressourcen aus einem Monitor in einen neu installierten Monitor zu migrieren.

Voraussetzungen

Vorsicht

Beim Deinstallieren des Monitors werden neben der Anwendung auch die Ressourcen und gesammelten Metriken entfernt. Um Ihre aktuellen Monitor-Ressourcen beizubehalten, müssen Sie die folgenden Schritte durchführen:

1. Installieren Sie eine neue Version des Monitors.
2. Migrieren Sie die Ressourcen.
3. Deinstallieren Sie die ältere Version des Monitors.

Um unerwünschte Alarmer zu vermeiden, planen Sie im SQL Anywhere-Monitor eine Blackoutperiode für Ihre Server, bevor Sie sie herunterfahren.

Aufgabe

1. Erstellen Sie eine Sicherungskopie der vorhandenen Monitor-Datenbankdatei *samonitor.db*.

Die Standardspeicherorte für die Datenbankdateien der Monitor-Versionen 11.0.1 und 16.0 werden in der folgenden Tabelle aufgelistet:

Betriebssystem	Monitortyp	Verzeichnisse der Versionen 11.0.1 und 12	Verzeichnis der Version 16.0
Windows XP	Monitor Developer Edition	C:\Documents and Settings\All Users\Documents\SQL Anywhere 11\Monitor\samonitor.db C:\Documents and Settings\All Users\Documents\SQL Anywhere 12\Monitor\samonitor.db	C:\Documents and Settings\All Users\Documents\SQL Anywhere 16\Monitor\samonitor.db
Windows XP	Monitor Production Edition	C:\Documents and Settings\All Users\Documents\SQL Anywhere 11 Monitor\samonitor.db C:\Documents and Settings\All Users\Documents\SQL Anywhere 12 Monitor\samonitor.db	C:\Documents and Settings\All Users\Documents\SQL Anywhere 16 Monitor\samonitor.db
Windows Vista und höhere Versionen von Windows	Monitor Developer Edition	C:\Users\Public\Documents\SQL Anywhere 11\Monitor\samonitor.db C:\Users\Public\Documents\SQL Anywhere 12\Monitor\samonitor.db	C:\Users\Public\Documents\SQL Anywhere 16\Monitor\samonitor.db
Windows Vista und höhere Versionen von Windows	Monitor Production Edition	C:\Users\Public\Documents\SQL Anywhere 11 Monitor\samonitor.db C:\Users\Public\Documents\SQL Anywhere 12 Monitor\samonitor.db	C:\Users\Public\Documents\SQL Anywhere 16 Monitor\samonitor.db
Linux	Monitor Developer Edition	/opt/sqlanywhere11/samonitor.db /opt/sqlanywhere12/samonitor.db	/opt/sqlanywhere16/samonitor.db

Betriebssystem	Monitortyp	Verzeichnisse der Versionen 11.0.1 und 12	Verzeichnis der Version 16.0
Linux	Monitor Production Edition	/opt/samonitor11/ samonitor.db /opt/samonitor12/ samonitor.db	/opt/ sqlanywhere16/ samonitor.db

2. Installieren Sie den neuen Monitor. Führen Sie die Datei *setup.exe* im Verzeichnis *Monitor* Ihres Installationsdatenträgers aus und befolgen Sie die Anweisungen. Wenn die Installation abgeschlossen ist, stoppen Sie den neuen Monitor (falls er ausgeführt wird).

Auf einem Computer kann jeweils nur eine Version des Monitors ausgeführt werden.

Wenn Sie den Monitor der Version 16.0 auf einem Computer installieren, auf dem eine ältere Version des Monitors läuft, stoppt das Installationsprogramm die alte Version des Monitors.

3. Starten Sie an einer Eingabeaufforderung das Migrator-Dienstprogramm.

Führen Sie das Migrator-Dienstprogramm mit den folgenden Optionen aus:

- **-t temporary-directory** Gibt das Verzeichnis für temporäre Dateien an. Standardmäßig werden die temporären Dateien in demselben Verzeichnis erstellt wie die *run_migrator*-Datei.

Hinweis

Der Monitor Migrator erstellt temporäre Dateien, die am Ende des Migrationsprozesses gelöscht werden. Sie können mit der Option -t ein Verzeichnis für diese temporären Dateien festlegen. Die temporären Dateien erfordern eine ähnliche Menge an Speicherplatz wie die alte Monitor-Datenbankdatei. Vergewissern Sie sich, dass der Speicherplatz des angegebenen Verzeichnisses ausreicht.

- **source-filename** Gibt den Pfad und den Dateinamen der alten Monitor-Datenbankdatei an. Zum Beispiel der Pfad der Datei *samonitor.db* der Version 11.0.1.
- **destination-filename** Gibt den Pfad und den Dateinamen der neuen Monitor-Datei an, in die die Ressourcen und Konfigurationseinstellungen geladen werden. Zum Beispiel der Pfad der Datei *samonitor.db* der Version 16.0.

Beispiel:

```
C:\Program Files\SQL Anywhere 16\run_migrator.cmd -t c:\monitorbackup c:\
\Program Files\SQL Anywhere 11\Monitor\samonitor11.db C:\Program Files
\SQL Anywhere 16\Monitor\samonitor16.db
```

Verwenden Sie die folgende Tabelle, um den Speicherort des Migrator-Dienstprogramms zu ermitteln.

Betriebssystem	Monitortyp	Standardspeicherort des Migrator-Dienstprogramms
Windows	Monitor Developer Edition	<i>C:\Program Files\SQL Anywhere 16\Bin32\run_migrator.cmd</i>
	Monitor Production Edition	<i>C:\Program Files\SQL Anywhere 16\Bin32\run_migrator.cmd</i>
Linux	Monitor Developer Edition	<i>/opt/sqlanywhere16/bin32/run_migrator.sh</i>
	Monitor Production Edition	<i>/opt/sqlanywhere16/bin32/run_migrator.sh</i>

Ergebnisse

Das Upgrade des Monitor wird durchgeführt.

Nächste Schritte

Monitor starten

Siehe auch

- „Monitor starten“ [[SQL Anywhere Server - Datenbankadministration](#)]

Index

Symbole

hook_dict-Tabelle

Version 16.0, Verhaltensänderung, MobiLink,51

%, Operator

Version 11.0.0, Verhaltensänderung,260

-ac, Option

Version 12.0.0, Erweiterung, dbsupport-Dienstprogramm,130

-al, Option

Version 16.0, neue Funktion,18

-aw, Option

Version 16.0, neue Funktion,14

-bc, Option

Version 16.0, neue Funktion,14

-bk, Option

Version 12.0.0, neue Funktion, MobiLink,164

-bkr, Option

Version 12.0.0, neue Funktion, MobiLink,164

-c, Option

Version 11.0.1, Erweiterung, MobiLink (mlmon) ,212

Version 12.0.1, Verhaltensänderung,94

-ca, Option

Version 12.0.0, neue Funktion, MobiLink,160

-cc, Option

Version 16.0, Verhaltensänderung,32

-ch, Option

Version 16.0, dbsupport, nicht unterstützt,33

Version 16.0, Erweiterung,30

-chx, Option

Version 12.0.0, neue Funktion,147

-ci, Option

Version 12.0.0, neue Funktion, MobiLink,164

-cinit, Option

Version 12.0.0, neue Funktion, MobiLink,160

Version 12.0.1, Erweiterung, MobiLink,97

-cl, Option

Version 12.0.0, neue Funktion, MobiLink,164

Version 12.0.1, Verhaltensänderung,94

-cm, Option

Version 11.0.1, neue Funktion, dbunload,200

Version 12.0.0, neue Funktion, MobiLink,164

Version 12.0.1, Verhaltensänderung, MobiLink (mlsrv12),102

Version 16.0, nicht unterstützt,40

-cmax, Option

Version 12.0.0, neue Funktion, MobiLink,160

Version 12.0.1, Erweiterung, MobiLink,97

-cmin, Option

Version 12.0.0, neue Funktion, MobiLink,160

Version 12.0.1, Erweiterung, MobiLink,97

-cn, Option

Version 12.0.0, Verhaltensänderung, MobiLink,167

-codepage, Option

ab Version 11.0.0 nicht mehr weiterentwickelt,285

ab Version 12.0.0 nicht mehr weiterentwickelt,193

-cp, Option

Version 11.0.0, Erweiterung, dbunload,227

Version 16.0, Verhaltensänderung,32

-cs, MobiLink-Option (mlsrv11)

Version 11.0.0, neue Funktion,265

-cw, Option

Version 16.0, nicht unterstützt,40

-df, Option

Version 12.0.0, Erweiterung, MobiLink,168

Version 12.0.0, Verhaltensänderung,

MobiLink,168

-dp, Option

Version 12.0.0, Erweiterung, MobiLink,168

Version 12.0.0, Verhaltensänderung,

MobiLink,168

-e, Option

Version 11.0.0, nicht unterstützt, Initialisierungs-Dienstprogramm (dbinit), Option,260

-ec, Datenbankserveroption

Version 16.0, nicht mehr empfohlen,7

-es, Option

Version 11.0.0, neue Funktion,228

-f, mlstop-Option

Version 16.0, nicht mehr empfohlen, MobiLink,52

-f, Option

Version 12.0.0, nicht mehr weiterentwickelt,

MobiLink,170

-fr, Option

Version 12.0.0, nicht mehr weiterentwickelt,

MobiLink,170

-ftru, Option

Version 12.0.0, neue Funktion, MobiLink,160

-g, Option

in Version 12.0.0 entfernt, Transaktionslog-Dienstprogramm, (dblog), Option,118

- Version 11.0.0, Erweiterung, dbunload,227
- Version 11.0.0, nicht mehr empfohlen, MobiLink
- Listener-Dienstprogramm (dblsn), Option,268
- ga, Option
 - Version 11.0.0, nicht mehr empfohlen, MobiLink
 - Listener-Dienstprogramm (dblsn), Option,268
- gb, Option
 - Version 11.0.0, Erweiterung,228
- gd, Option
 - Version 16.0, Verhaltensänderung,3
- gi, Option
 - Version 11.0.0, nicht mehr empfohlen, MobiLink
 - Listener-Dienstprogramm (dblsn), Option,268
- gk, Option
 - Version 16.0, Verhaltensänderung,3
- gl, Option
 - Version 16.0, Verhaltensänderung,3
- gn, Option
 - Version 12.0.0, Erweiterung,125
- gna, Option
 - Version 12.0.0, Erweiterung,125
 - Version 12.0.1, Verhaltensänderung,94
- gnh, Option
 - Version 12.0.0, Erweiterung,125
 - Version 12.0.1, Verhaltensänderung,94
- gnl, Option
 - Version 12.0.0, Erweiterung,125
 - Version 12.0.1, Verhaltensänderung,94
- gns, Option
 - Version 12.0.0, neue Funktion,125
 - Version 12.0.1, Verhaltensänderung,94
- gss, Option
 - Version 11.0.0, Erweiterung,245
 - Version 11.0.0, Verhaltensänderung,252
- gta, Option
 - Version 16.0, neue Funktion,18
- gu, Option
 - Version 12.0.0, Verhaltensänderung,156
 - Version 16.0, Verhaltensänderung,3
- gx, Option
 - Version 11.0.0, nicht unterstützt,252
- h, Option
 - Version 16.0, neue Funktion,14
- i, Option
 - Version 12.0.0, neue Funktion, MobiLink,165
- il, Option
 - in Version 12.0.0 entfernt, Transaktionslog-Dienstprogramm (dblog), Option,118
- im, Option
 - Version 11.0.0, neue Funktion,228
- is, Option
 - Version 12.0.0, Verhaltensänderung, Dienstprogramm Transaktionslogübersetzung (dbtran), Option,117
- k, Option
 - Version 12.0.0, Erweiterung, dblic-Dienstprogramm,129
- kd, Option
 - Version 12.0.0, Erweiterung, dbunload-Dienstprogramm ,130
- kp, Option
 - Version 12.0.1, neue Funktion,86
- kr, Option
 - ab Version 12.0.1 nicht mehr empfohlen,96
- ks, Option
 - Version 11.0.0, neue Funktion,229
- ksc, Option
 - Version 11.0.0, neue Funktion,229
- ksd, Option
 - Version 11.0.0, neue Funktion,229
- l, Option
 - Version 11.0.1, neue Funktion, dbunload,200
- ls, Option
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),99
 - Version 16.0, neue Funktion, MobiLink,50
- lsc, MobiLink-Option (mlsrv11)
 - Version 11.0.0, neue Funktion,265
- lu, Option
 - Version 16.0, neue Funktion, MobiLink,50
- n, Option
 - Version 12.0.0, nicht mehr empfohlen, dbmlsync,171
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),98
 - Version 12.0.1, Verhaltensänderung, MobiLink (mlreplay),102
- nba, Option
 - Version 12.0.0, nicht mehr weiterentwickelt, MobiLink,170
- ni, Option
 - Version 11.0.0, neue Funktion, dblsn,267
- no, Option
 - Version 11.0.0, Erweiterung, dbunload,227
- ns, Option
 - Version 16.0, Verhaltensänderung,50

-
- nu, Option
 - Version 11.0.0, neue Funktion, dblsn,267
 - Version 16.0, Verhaltensänderung,50
 - o, Option
 - Version 12.0.1, Verhaltensänderung, MobiLink (mlgenreplayapi),102
 - Version 12.0.1, Verhaltensänderung, MobiLink (mlreplay),102
 - os, Option
 - Version 12.0.1, neue Funktion, MobiLink (mlgenreplayapi),99
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),99
 - ot, Option
 - Version 12.0.1, Verhaltensänderung, MobiLink (mlgenreplayapi),102
 - Version 12.0.1, Verhaltensänderung, MobiLink (mlreplay),102
 - p, Option
 - Version 16.0, Erweiterung,30
 - pc, Option
 - Version 11.0.0, neue Funktion, dblsn,267
 - pd, Option
 - dbinit, Version 16.0, neue Funktion,28
 - dbupgrad, Version 16.0, neue Funktion,29
 - ppv, Option
 - Version 11.0.1, Erweiterung, MobiLink (mlsrv11),211
 - Version 12.0.0, Erweiterung, MobiLink,160
 - qi, Option
 - Version 12.0.1, Verhaltensänderung, MobiLink,84
 - qr, Option
 - Version 12.0.0, Erweiterung, dbunload-Dienstprogramm,130
 - r, Option
 - Version 12.0.0, nicht mehr weiterentwickelt, MobiLink,170
 - rep, Option
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),98
 - rnt, Option
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),98
 - rp-Option
 - Version 12.0.0, neue Funktion, MobiLink,161
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),99
 - rrp, Option
 - Version 12.0.0, neue Funktion, MobiLink,160
 - rs, Option
 - Version 12.0.0, Erweiterung, dbsvc-Dienstprogramm,130
 - rsu, Option
 - in Version 12.0.0 entfernt, Dienstprogramm Transaktionslogübersetzung (dbtran), Option,117
 - s, Option
 - Version 12.0.0, Erweiterung, Initialisierungs-Dienstprogramm (dbinit),148
 - Version 12.0.0, neue Funktion, MobiLink,163,164
 - sbx, Option
 - Version 16.0, neue Funktion,16
 - sci, Option
 - Version 12.0.1, Verhaltensänderung, MobiLink (mlreplay),102
 - Version 16.0, Verhaltensänderung, MobiLink (mlreplay),44
 - sf, Option
 - Version 11.0.0, Erweiterung,228,244
 - Version 16.0, Erweiterung,15
 - sk, Option
 - Version 16.0, Erweiterung,15
 - sl dnet
 - Version 12.0.0, Verhaltensänderung, MobiLink,167
 - sl, Java-Option
 - Version 12.0.0, Verhaltensänderung, MobiLink,160
 - sm, Option
 - Version 11.0.0, neue Funktion,223
 - Version 11.0.1, Verhaltensänderung, MobiLink ,212
 - Version 12.0.0, Verhaltensänderung,156
 - sp, Option
 - Version 11.0.0, neue Funktion ,266
 - ss, Option
 - Version 12.0.0, Verhaltensänderung, MobiLink,171
 - sv, Option
 - Version 11.0.1, Erweiterung, MobiLink (dblsn),211
 - Version 16.0, neue Funktion, MobiLink (mlreplay),44
 - t, Option
 - Version 12.0.0, Verhaltensänderung, Dienstprogramm für Dienste (dbsvc), Option,118
 - tc, MobiLink-Option (mlsrv11)

- Version 11.0.0, neue Funktion,266
- tdsl, Option
 - Version 16.0, neue Funktion,14
- tf, MobiLink-Option (mlsrv11)
 - Version 11.0.0, neue Funktion,266
- ts, Option
 - Version 16.0, neue Funktion, MobiLink, dblsn-Option,41
 - Version 16.0, neue Funktion, MobiLink, dbmlsync-Option,41
 - Version 16.0, neue Funktion, MobiLink, mlsrv16-Option,41
- u, Option
 - Version 12.0.0, nicht mehr empfohlen, dbmlsync,171
- uc, Serveroption
 - Version 11.0.0, Verhaltensänderung,259
- ud, Option
 - Version 12.0.0, Verhaltensänderung,118
- uf, Option
 - Version 16.0, Erweiterung,19
- ufd, Option
 - Version 16.0, neue Funktion,19
- ui, Serveroption
 - Version 11.0.0, Verhaltensänderung,259
- um, Option
 - Version 11.0.0, neue Funktion,229
- version
 - Version 11.0.0, neue Funktion, dbisql,280
- vi, Option
 - Version 11.0.1, Erweiterung, MobiLink (mlsrv11) ,210
- vk, Option
 - Version 12.0.0, neue Funktion, MobiLink,161
- vm, Option
 - Version 11.0.1, Erweiterung, MobiLink (mlsrv11),210
- vq, Option
 - Version 11.0.1, Erweiterung, MobiLink (mlsrv11) ,210
- vR, Option
 - Version 12.0.0, neue Funktion, MobiLink,161
- vU, Option
 - Version 12.0.0, neue Funktion, MobiLink,161
- w, Option
 - Version 12.0.0, Verhaltensänderung, Dienstprogramm für Dienste (dbsvc), Option,118
- Version 12.0.1, Verhaltensänderung, MobiLink (mlsrv12),102
- wa, Option
 - Version 16.0, neue Funktion,14
- wb, Option
 - Version 16.0, neue Funktion,14
- wc, Option
 - Version 12.0.0, neue Funktion, Datenbankserver,127
- wm, Option
 - Version 12.0.1, neue Funktion, MobiLink,97
- wn, mlsrv16-Option
 - Version 16.0, neue Funktion, MobiLink,42
- x, Option
 - Version 12.0.0, Erweiterung, MobiLink,160
 - Version 12.0.1, Erweiterung, MobiLink,97
- xa, Option
 - Version 12.0.1, Erweiterung,76
- xd, Option
 - Version 11.0.1, neue Funktion, Datenbankserver,205
- xm, Option
 - Version 12.0.0, neue Funktion, Datenbankserver,131
- xo, Option
 - Version 12.0.0, nicht mehr weiterentwickelt, MobiLink,170
- xp, Option
 - Version 11.0.0, Verhaltensänderung,254
 - Version 12.0.0, Verhaltensänderung,151
- xs, Option
 - Version 16.0, Erweiterung,19
- z, Option
 - Version 12.0.1, neue Funktion,82
- zoc, Option
 - Version 11.0.0, neue Funktion,244
- zt, Option
 - Version 12.0.1, Verhaltensänderung, SQL Anywhere,95
- .NET
 - Version 11.0.0, Verhaltensänderung,288
- .scRepository
 - Version 11.0.0, umbenannt,281
- 256-Bit-AES-Verschlüsselung
 - Version 11.0.0, neue Funktion,225
 - Version 16.0, neue Funktion, UltraLite,59
- 32-Bit-MobiLink-Server
 - Version 16.0, Verhaltensänderung,50

-
- 32-Bit-Relay Server
 - Version 16.0, nicht mehr weiterentwickelt (Linux),56
 - Version 16.0, nicht mehr weiterentwickelt (Windows),56
 - 64-Bit
 - Version 11.0.0, neue Funktion, MobiLink-Unterstützung,265
 - Version 11.0.0, Verhaltensänderung,257
 - @data-Option
 - Version 11.0.0, Erweiterung,227
 - Version 12.0.1, neue Funktion, MobiLink (mlgenreplayapi),98
 - Version 12.0.1, neue Funktion, MobiLink (mlreplay),98
 - A**
 - a_backup_db-Struktur
 - Version 16.0, Verhaltensänderung,36
 - a_change_log, Struktur
 - Version 12.0.0, Verhaltensänderung,117
 - a_sync_db-Struktur
 - Version 16.0, Änderung,52
 - Version 16.0, Verhaltensänderung,36
 - a_validate_type-Enumeration
 - Version 11.0.0, Erweiterung,247
 - Abfragen
 - Version 11.0.0, neue Funktion, BLOBs,223
 - Version 11.0.0, neue Funktion, Dateien ,223
 - Version 16.0, Erweiterung,9
 - Abfrageoptimierung
 - Version 11.0.1, Erweiterung,204
 - Version 12.0.1, Erweiterung,77
 - Abfrageperformance
 - Version 11.0.1, Erweiterung,204
 - AcceptCharset, Option
 - Version 11.0.1, neue Funktion,201
 - active_cookie-Option
 - Version 12.0.0, neue Funktion, Relay Server,166
 - active_header
 - Version 12.0.0, neue Funktion, Relay Server,166
 - Adaptive Server Enterprise
 - Version 12.0.1, neue Funktion, MobiLink,97
 - Adaptive Server Enterprise 12.5.x
 - Version 12.0.0, nicht mehr weiterentwickelt, MobiLink,170
 - Adaptive Server Enterprise-Unterstützung
 - Version 12.0.1, neue Funktion, MobiLink,96
 - addBatch
 - Version 12.0.0, Erweiterung,142
 - Address Windowing Extensions
 - Version 12.0.0, nicht mehr empfohlen,154
 - Version 16.0, nicht unterstützt,40
 - Administrationstools
 - Version 11.0.1, Erweiterung,205
 - Version 11.0.1, Verhaltensänderung,210
 - Version 16.0, Erweiterung,67
 - ADO.NET
 - Version 11.0.1, Erweiterung,203
 - Version 12.0.1, Erweiterung,77,90
 - Version 16.0, neue Funktionen,24
 - Adressenspeicher
 - Version 12.0.0, Verhaltensänderungen,148
 - ADSODBC-Serverklasse
 - Version 11.0.0, neue Funktion,247
 - AES-Verschlüsselungsalgorithmus
 - Version 16.0, Erweiterung,26
 - AES256, Verschlüsselungsalgorithmus
 - Version 11.0.0, neue Funktion,225
 - AES256_FIPS, Verschlüsselungsalgorithmus
 - Version 11.0.0, neue Funktion,225
 - Aggregatfunktionen
 - äußere Referenzen,323
 - Aktualisieren
 - Version 11.0.0, Verhaltensänderung, materialisierte Ansichten,255
 - Version 16.0, Erweiterung, MobiLink-Plug-In,49
 - Alarme
 - Monitor Version 12.0.1, neue Funktion,114,115
 - Aliasnamen
 - Version 16.0, Erweiterung,28
 - ALLOW-Klausel
 - Version 16.0, neue Funktion,35
 - allow_read_client_file-Eigenschaft
 - Version 11.0.0, neue Funktion,230
 - allow_read_client_file-Option
 - Version 11.0.0, neue Funktion,228
 - allow_write_client_file-Option
 - Version 11.0.0, neue Funktion,228
 - allowPasswordsInFavorites
 - Version 12.0.0, neue Funktion,191
 - Als ODBC-Datenquelle speichern
 - Version 11.0.0, neue Funktion,277
 - ALTER DATABASE-Anweisung

- Upgrade von Datenbanken der Version 10 und höher,313
- Version 11.0.0, Erweiterung,235
- Version 11.0.1, Erweiterung,202
- Version 12.0.0, Erweiterung,140
- Version 16.0, Erweiterung,13,24,29
- Version 16.0, Verhaltensänderung,36
- ALTER EVENT-Anweisung
 - Version 11.0.0, Erweiterung,237
 - Version 16.0, Erweiterung,21
 - Version 16.0, neue Funktion,11
- ALTER EXTERNAL ENVIRONMENT-Anweisung
 - Version 12.0.0, Erweiterung,147
- ALTER FUNCTION-Anweisung
 - Version 11.0.0, Erweiterung,235
- ALTER LDAP SERVER-Anweisung
 - Version 16.0, neue Funktion,6
- ALTER LOGIN POLICY-Anweisung
 - Version 11.0.0, neue Funktion,236
 - Version 16.0, Erweiterung,2
- ALTER MATERIALIZED VIEW-Anweisung
 - Version 11.0.0, Erweiterung,235
- ALTER MIRROR SERVER-Anweisung
 - Version 12.0.0, neue Funktion,123
- ALTER PROCEDURE-Anweisung
 - Version 11.0.0, Erweiterung,235
 - Version 12.0.0, Verhaltensänderung,118
- ALTER ROLE-Anweisung
 - Version 16.0, neue Funktion,4
- ALTER SEQUENCE-Anweisung
 - Version 12.0.0, neue Funktion,124
- ALTER SERVER-Anweisung
 - Version 12.0.0, Erweiterung,136
 - Version 12.0.1, Verhaltensänderung,77
 - Version 16.0, Verhaltensänderung,37
- ALTER SPATIAL REFERENCE SYSTEM-Anweisung
 - Version 12.0.0, neue Funktion,119
- ALTER SYNCHRONIZATION SUBSCRIPTION-Anweisung
 - Version 12.0.0, Erweiterung,159
 - Version 12.0.0, Erweiterung, MobiLink,163
 - Version 12.0.1, Erweiterung, OnDemand,74
 - Version 16.0, Erweiterung,46
- ALTER SYNCHRONIZATION USER-Anweisung
 - Version 12.0.1, Verhaltensänderung, OnDemand,74
 - Version 16.0, Erweiterung,46
- ALTER TABLE-Anweisung
 - Version 12.0.0, Verhaltensänderung,118
 - Version 12.0.1, Erweiterung,78
 - Version 16.0, Erweiterung,21,27
- ALTER TEXT CONFIGURATION-Anweisung
 - Version 11.0.0, neue Funktion,237
 - Version 12.0.0, Erweiterung,136
- ALTER TEXT INDEX-Anweisung
 - Version 11.0.0, neue Funktion,237
- ALTER TRACE EVENT SESSION-Anweisung
 - Version 16.0, neue Funktion,22
- ALTER USER-Anweisung
 - Version 11.0.0, neue Funktion,236
 - Version 16.0, Erweiterung,6
- AlternateMirrorServerName-Eigenschaft
 - Version 11.0.0, neue Funktion,232
- Alternative Servernamen
 - Version 12.0.0, Verhaltensänderung,150
- an_unload_db, Struktur
 - Version 12.0.0, Erweiterung,141
- Änderungen in Version
 - 11.0.0,219
 - 11.0.1,199
 - 12.0.0,117
 - 12.0.1,73
 - 16.0,1
 - Version 9.0.2 und früher,vii
- Android
 - Version 12.0.1, Erweiterung, UltraLite,107
 - Version 12.0.1, neue Funktion, UltraLite-Unterstützung für Android,106
 - Version 12.0.1, neue Unterstützung, UltraLite,104
- Anfängliche Cachegröße
 - Version 12.0.1, Verhaltensänderung,94
- Anforderungslog
 - Version 16.0, Verhaltensänderung,31
- Anforderungsprotokollierung
 - Version 16.0, Erweiterung,72
- Angepasstes Benutzerauthentifizierungsverfahren, Meldungen
 - Version 16.0, Erweiterung, UltraLite,60
- ansi_integer_overflow-Option
 - Version 11.0.0, nicht unterstützt,260
- ansi_substring, Option
 - Version 11.0.1, nicht mehr empfohlen,210
- ansi_substring-Option
 - Version 11.0.0, nicht unterstützt,260
- Ansichten

Version 11.0.0, Erweiterung,246
 Version 16.0, Verhaltensänderung,29,36
 Anweisungen
 Version 12.0.1, Erweiterung,76,77
 Apache
 Version 12.0.1 schnelle Deployment, Relay Server,84
 Version 16.0, 32-Bit-Relay Server nicht mehr weiterentwickelt,56
 Version 16.0, neue Funktion, Schnellsetup für Relay Server,54
 APP-Verbindungsparameter
 Version 11.0.0, Erweiterung,224
 AppInfo-Verbindungsparameter
 Version 11.0.0, Erweiterung,224
 APPLY-Ausdrücke
 Version 11.0.0, neue Funktion,239
 ApproximateCPUTime, Eigenschaft
 Version 12.0.1, neue Funktion,92
 Arbiterserver
 Version 12.0.1, Erweiterung,76
 Archivsicherungen
 Version 11.0.1, Erweiterung,205
 Version 12.0.0, Erweiterung,128
 ARRAY-Konstruktor
 Version 16.0, neue Funktion,9
 Array-Operatoren
 Version 16.0, neue Funktion,9
 ARRAY_AGG-Funktion
 Version 16.0, neue Funktion,9
 ARRAY_MAX_CARDINALITY-Funktion
 Version 16.0, neue Funktion,9
 ASE 15.7-Unterstützung
 Version 16.0, neue Funktion, MobiLink,42
 ASEJDBC-Serverklasse
 Version 12.0.0, nicht mehr empfohlen,155
 Version 16.0, nicht unterstützt,39
 ASP.NET
 Version 11.0.1, neue Funktion,203
 Assertierungen
 Version 16.0, Verhaltensänderung,31
 Assistent für das Deployment eines Synchronisationsmodells
 Version 11.0.0, Erweiterung,281
 Version 16.0, Verhaltensänderung,47
 Assistent zum Entladen einer Datenbank
 Datenbanken der Version 10 und höher neu aufbauen,300
 Datenbanken der Version 9 und früher neu aufbauen,307
 Assistent zum Erstellen der Datenbankdokumentation
 Version 11.0.0, neue Funktion,278
 Version 12.0.0, Erweiterung,188
 Assistent zum Erstellen einer Datenbank
 Version 12.0.0, Erweiterung,190
 Assistent zum Erstellen eines neuen Dienstes
 Version 11.0.1, Erweiterung,214
 Assistent zum Erstellen eines Passthrough-Downloads
 Version 11.0.0, neue Funktion,279
 Assistent zum Erstellen eines Projekts
 Version 16.0, neue Funktion, MobiLink-Plug-In,47
 Assistent zum Erstellen eines räumlichen Bezugssystems
 Version 12.0.0, neue Funktion,121
 Assistent zum Erstellen eines Synchronisationsmodells
 Version 16.0, Erweiterung, MobiLink-Plug-In,48
 Assistent zum Erstellen eines Synchronisationsprofils
 Version 11.0.0, neue Funktion,278
 Assistent zum Erstellen eines Textkonfigurationsobjekts
 Version 12.0.0, Erweiterung,188
 Assistent zum Erstellen von Benutzern
 Version 11.0.0, Erweiterung,281
 Assistent zum Erstellen von Funktionen
 Version 12.0.0, Erweiterung,189
 Assistent zum Erstellen von Passthrough-Skripten
 Version 11.0.0, neue Funktion,279
 Assistent zum Erstellen von Wartungsplänen
 Version 11.0.1, Erweiterung,215
 Version 12.0.0, Erweiterung,188
 Assistent zum Extrahieren einer Datenbank
 Version 11.0.0, Erweiterung, UltraLite,272
 Assistent zum Sichern einer Datenbank
 Version 12.0.0, Erweiterung,189
 Assistent zum Upgrade einer Datenbank
 Upgrade von Datenbanken der Version 10 und höher,311
 Assistent zum Validieren einer Datenbank
 Version 11.0.0, neue Funktion,278
 Assistent zur Erstellung eines Sequenzgenerators
 Version 12.0.0, neue Funktion,124
 Assistenten
 Version 11.0.0, Erweiterung,281
 Version 11.0.0, neue Funktion, Assistenten,278
 Asynchrone Synchronisation

- UltraLite.NET,106
 - Auditing
 - Version 11.0.0, Erweiterung,225
 - Version 11.0.0, neue Funktion,278
 - Ausdrücke
 - Version 12.0.0, Erweiterung,146
 - Äußere Referenzen
 - Aggregatfunktionen,323
 - Ausführen, SQL-Anweisungen
 - Version 11.0.0, Erweiterung, Interactive SQL ,282
 - Ausführungspläne
 - Version 11.0.0, Erweiterung,248
 - Version 11.0.1, Erweiterung,204
 - Ausgabe
 - Version 11.0.0, Erweiterung, Interactive SQL,282
 - Auswählen aus DML
 - Version 12.0.0, neue Funktion,126
 - authenticate_file_upload-Verbindungsereignis
 - Version 12.0.0, neue Funktion, MobiLink,162
 - Authenticated-Eigenschaft
 - Version 11.0.1, neue Funktion, Datenbankeigenschaft,200
 - Version 11.0.1, neue Funktion, Verbindungseigenschaft,199
 - Authentifizierung, Synchronisationsskripten
 - Version 16.0, Erweiterung, UltraLite,60
 - Authentifizierungsparameter
 - Version 12.0.1, neue Funktion, MobiLink,97
 - AuthType-Eigenschaft
 - Version 11.0.0, neue Funktion,230
 - auto_commit, Option
 - Version 12.0.0, Verhaltensänderung,193
 - auto_commit_on_create_local_temp_index-Eigenschaft
 - Version 16.0, neue Funktion,19
 - auto_commit_on_create_local_temp_index-Option
 - Version 16.0, neue Funktion,18
 - Version 16.0, Verhaltensänderung,30
 - auto_refetch
 - Version 12.0.0, Verhaltensänderung,193
 - auto_unlock_time-Option
 - Version 16.0, neue Funktion,2
 - autocommit
 - Version 11.0.0, Verhaltensänderung,258
 - AUTOINCREMENT
 - nächsten verfügbaren Wert in neu erstellter Datenbank abfragen,300
 - reset.sql-Skript zum Abfragen des nächsten verfügbaren Werts verwenden,300
 - automatic_timestamp-Option
 - Version 11.0.0, nicht unterstützt,260
 - Automatische Cachedimensionierung
 - Version 12.0.1 neue Funktion, UltraLite,105
 - Automatische Startoption
 - Relay Server, Version 16, nicht unterstützt,56
 - Automatisches Verbindungspooling
 - Version 12.0.0, Erweiterung,141
 - AutoMultiProgrammingLevel, Eigenschaft:
 - Version 12.0.0, neue Funktion,133
 - AutoMultiProgrammingLevelStatistics, Eigenschaft
 - Version 12.0.0, neue Funktion,133
 - AWE
 - Version 12.0.0, nicht mehr empfohlen,154
 - Version 16.0, nicht unterstützt,40
- ## B
- Back Quotes
 - Version 12.0.0, neue Funktion,147
 - background_priority-Option
 - Version 11.0.0, nicht mehr empfohlen,262
 - BACKUP-Anweisung
 - Version 11.0.1, Erweiterung,205
 - Version 12.0.0, Erweiterung,128
 - BACKUP-Berechtigung
 - Version 16.0, nicht mehr empfohlen,37
 - Batches
 - Version 11.0.0, Erweiterung,246
 - BatchUpdateCount
 - Version 12.0.0, Erweiterung,142
 - BatchUpdateException
 - Version 12.0.0, Erweiterung,142
 - Befehlszeilen-Dienstprogramm
 - mehrere UltraLite-Versionen,337
 - Befehlszeilen-Dienstprogramme
 - mehrere Versionen,298
 - UltraLite-Upgrade,337
 - Upgrade,298
 - BEGIN SNAPSHOT-Anweisung
 - Version 11.0.0, neue Funktion,238
 - BEGIN-Anweisung
 - Version 12.0.0, Erweiterung,137
 - Begrenzende SQL-Zeichenfolgen
 - Version 12.0.0, Erweiterung,147
 - Beispieldatenbank

Version 11.0.0, Erweiterung, SQL Anywhere ,287
 Version 12.0.0, Erweiterung,146
 Beispielzertifikate
 Version 16.0, Verhaltensänderung,7,51
 bell, Option
 Version 12.0.0, Verhaltensänderung,193
 Benachrichtigungsunterstützung
 Version 16.0, entfernt, MobiLink-Plug-In,51
 Benannte Parameter
 Version 16.0, neue Funktion, MobiLink,43
 Benannte Subskription
 Version 12.0.0, neue Funktion, MobiLink,163
 Benannte Systemparameter
 Version 12.0.1, neue Funktion, MobiLink,97
 Benutzerdefinierte Funktionen
 Version 12.0.0, Erweiterung,146
 Version 12.0.1, Erweiterung,76,77
 Benutzerdefinierte Kollationen
 nur bei internem Entladen bewahren,305
 Benutzerdefinierte Schlüssel für gesicherte Funktionen
 Version 16.0, Verhaltensänderung,31
 Berechnete Spalten
 Version 11.0.0, Verhaltensänderung,253
 Berechtigungen
 Version 16.0, Verhaltensänderung, durch Privilegien ersetzt,1
 Version 16.0, Verhaltensänderung, durch Rollen ersetzt,37
 Beschädigte Datenbanken
 Version 12.0.0, Erweiterung,147
 Bezeichner
 Version 12.0.0, Erweiterung,136,147,151
 Version 16.0, Erweiterung,28
 Bezeichner in Anführungszeichen
 Version 11.0.0, Verhaltensänderung,255
 Big-Endian, UTF-16-Kodierung
 Version 11.0.0, neue Funktion,246
 BIGINT UNSIGNED-Datentyp
 Version 16.0, neue Funktion,24
 BIGINT-Datentyp
 Version 12.0.0, Verhaltensänderung, MobiLink,168
 bin32, Verzeichnis
 Version 11.0.0, Verhaltensänderung,288
 bin64, Verzeichnis
 Version 11.0.0, Verhaltensänderung,288
 Bindestriche
 Version 11.0.1, Verhaltensänderung, Volltextsuche,206
 BINSEARCH-Protokolloption
 Version 11.0.0, nicht unterstützt,260
 BINTOHEX-Funktion
 Version 16.0, neue Funktion,21
 BIT_AND-Funktion
 Version 12.0.0, Erweiterung,135
 BIT_OR-Funktion
 Version 12.0.0, Erweiterung,135
 BIT_XOR-Funktion
 Version 12.0.0, Erweiterung,135
 BlackBerry
 Version 11.0.0, neue Funktion, UltraLite-Unterstützung für BlackBerry,275
 BlackBerry Enterprise Server
 Version 12.0.1, Erweiterung,82
 BLOBs
 Version 11.0.0, neue Funktion, Abfragen,223
 Blockierung
 Version 11.0.0, Verhaltensänderung,258
 Version 12.0.0, Erweiterung,130
 Version 16.0, Erweiterung, MobiLink,41
 blocking_others_timeout-Option
 Version 12.0.0, neue Funktion,130
 Blockmodus, Übertragungskodierung für HTTP Clients
 Version 11.0.1, Verhaltensänderung,210
 Breitzeichen-Datenstreaming
 Version 16.0, Erweiterung, UltraLite C/C++,61
 Broadcast Repeater, Dienstprogramm(dbns12)
 Version 12.0.0, Verhaltensänderung,118
 BufferDownload, erweiterte Option
 Version 12.0.0, neue Funktion, MobiLink,164
 Bypass-Abfragen
 Version 12.0.1, Erweiterung,90
 BYTE ORDER MARK
 Version 11.0.0, Erweiterung,284
 BytesReceived, Eigenschaft
 Version 12.0.1, neue Funktion,92
 BytesReceivedUncomp, Eigenschaft
 Version 12.0.1, neue Funktion,92
 BytesSent, Eigenschaft
 Version 12.0.1, neue Funktion,92
 BytesSentUncomp, Eigenschaft
 Version 12.0.1, neue Funktion,92

C

- C++-Anwendung migrieren
 - Version 11.0.0, nicht unterstützt,276
- Cache
 - Version 12.0.0, Erweiterung,147
 - Version 12.0.1, Verhaltensänderung,94
 - Version 16.0, Erweiterung, UltraLite,60
- CacheHits-Eigenschaft
 - Version 11.0.0, neue Funktion,258
- CacheHitsEng-Eigenschaft
 - Version 11.0.0, umbenannt,258
- Cachelesevorgänge, Arbeitstabelle, Statistik
 - Version 11.0.0, neue Funktion,232
- CacheRead-Eigenschaft
 - Version 11.0.0, neue Funktion,258
- CacheReadEng-Eigenschaft
 - Version 11.0.0, umbenannt,258
- CacheReadWorkTable-Eigenschaft
 - Version 11.0.0, neue Funktion,230,232
- CALL-Anweisung
 - Version 12.0.0, nicht mehr empfohlen, Funktionen aufrufen,154
- CancelSync-Methode
 - Version 12.0.0, neue Funktion, MobiLink,164
- CarverHeapPages, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- CASE-Anweisung
 - Version 11.0.0, Erweiterung,238
 - Version 11.0.0, Erweiterung, UltraLite,274
 - Version 12.0.0, Erweiterung,138
- CASE-Ausdruck
 - Version 11.0.0, Erweiterung,238
 - Version 11.0.0, Erweiterung, UltraLite,274
- CATALOGS-Zeilengruppe, OLE DB
 - Version 11.0.1, neue Funktion,203
- certificate-Protokolloption
 - Version 11.0.0, nicht unterstützt,288
 - Version 11.0.0, Verhaltensänderung,268
- certificate_password-Protokolloption
 - Version 11.0.0, nicht unterstützt,288
 - Version 11.0.0, Verhaltensänderung,268
- CESU-8-Zeichensatz
 - Version 16.0, Erweiterung,27
- CESU8BIN-Kollation
 - Version 16.0, Erweiterung,27
- change_password_dual_control-Option
 - Version 16.0, neue Funktion,13
- CHAR, Datentyp
 - Version 12.0.0, Erweiterung,140
- Checkpoint erstellen
 - Tabellen mit Autoinkrement-Spalten,300
- Checkpoint setzen
 - Version 12.0.0, Verhaltensänderung,148
- Checkpoint-Logs
 - Version 12.0.0, Verhaltensänderung,148
- checkpoint-Methode
 - Version 16.0, entfernte API, UltraLite ,66
- CheckpointLogBitmapPagesWritten, Eigenschaften
 - Version 12.0.0, nicht unterstützt,155
- CheckpointLogBitmapSize, Eigenschaft
 - Version 12.0.0, nicht unterstützt,155
- Checkpoints
 - Version 12.0.0, Verhaltensänderung,148
- CHECKSUM-Klausel
 - Version 12.0.0, Erweiterung,140
 - Version 12.0.0, Erweiterung, ALTER DATABASE-Anweisung,127
 - Version 12.0.0, Erweiterung, CREATE DATABASE-Anweisung,148
 - Version 12.0.0, Erweiterung, START DATABASE-Anweisung,127
- Chiffrierschlüssel, Aktualisierungen unter Android
 - Version 16.0, Erweiterung, UltraLiteJ,63
- ClassLoader
 - Version 12.0.1, Erweiterung,90
- CLEAR-Anweisung
 - Version 12.0.0, Verhaltensänderung,193
- clearBatch
 - Version 12.0.0, Erweiterung,142
- Client-Authentifizierung mit Common Access Cards
 - Version 11.0.1, neue Funktion,211
- Clientbibliotheken
 - Version 16.0, Verhaltensänderung,35
- Clientdateien, lesen aus
 - Version 11.0.0, neue Funktion,221
- Clientdateien, schreiben in
 - Version 11.0.0, neue Funktion,221
- ClientNodeAddress-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- ClientStmtCacheHits-Eigenschaft
 - Version 12.0.1, neue Funktion,92
- ClientStmtCacheMisses-Eigenschaft
 - Version 12.0.1, neue Funktion,92
- Cloud-Unterstützung
 - Version 12.0.1, neue Funktion,73

Codebeispiel, Speicherorte
 UltraLiteJ,106

collect_network_data-Protokolloption
 Version 12.0.1, neue Funktion, MobiLink,83

CommBufferSize-Verbindungsparameter
 Version 16.0, Erweiterung,30

COMMENT-Anweisung
 Version 11.0.0, Erweiterung,238
 Version 11.0.1, nicht mehr empfohlene Klausel,210
 Version 12.0.0, Erweiterung,123,124
 Version 16.0, Erweiterung,6

Commit, Eigenschaft
 Version 12.0.1, neue Funktion,93

COMMIT-Anweisung
 Version 12.0.0, Erweiterung,191

commit_on_exit, Option
 Version 12.0.0, Verhaltensänderung,193

Common Access Cards
 Version 11.0.1, neue Funktion,211

Compact Framework 3.5-Unterstützung
 Version 16.0, neue Funktion,24

ConfigPersistent.setRowScoreFlushSize-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,82

ConfigPersistent.setRowScoreMaximum-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,82

conflicted_deletes-Eigenschaft
 Version 16.0, entfernt, MobiLink,52

conflicted_inserts-Eigenschaft
 Version 16.0, entfernt, MobiLink,52

Connect-Methode
 Version 16.0, neue Funktion, Dbmlsync-.NET-API,46
 Version 16.0, neue Funktion, Dbmlsync-C++-API,45

ConnectedTime, Eigenschaft
 Version 12.0.1, neue Funktion,91

CONNECTION CLOSE-Klausel
 Version 12.0.1, Verhaltensänderung,77
 Version 16.0, nicht unterstützt,37

Connection.getNewUUID-Methode [UltraLite für M-Business Anywhere]
 Version 12.0.1, neue Funktion,83

CONNECTION_EXTENDED_PROPERTY-Funktion
 Version 16.0, Verhaltensänderung,33

CONNECTION_PROPERTY-Funktion
 Version 16.0, Verhaltensänderung,33

ConnectionPool-Verbindungsparameter
 Version 12.0.0, neue Funktion,128

ConnPoolCachedCount, Eigenschaft
 Version 12.0.0, neue Funktion,132

ConnPoolHits, Eigenschaft
 Version 12.0.0, neue Funktion,132

ConnPoolMisses, Eigenschaft
 Version 12.0.0, neue Funktion,132

CONTAINS-Suchbedingung
 Version 11.0.0, neue Funktion,237
 Version 16.0, Erweiterung,27

CONVERT-Funktion
 Version 12.0.1, Verhaltensänderung,95

Cookies
 Relay Server,54

cooperative_commit_timeout-Option
 Version 11.0.0, Verhaltensänderung,255

cooperative_commits-Option
 Version 11.0.0, Verhaltensänderung,255

CopyNodeParent
 Version 16.0, neue Funktion,12

COUNT_BIG-Funktion
 Version 12.0.0, neue Funktion,136

Counter1, Statistik
 Version 12.0.1, neue Funktion,93

Counter2, Statistik
 Version 12.0.1, neue Funktion,93

Counter3, Statistik
 Version 12.0.1, neue Funktion,93

Counter4, Statistik
 Version 12.0.1, neue Funktion,93

Counter5, Statistik
 Version 12.0.1, neue Funktion,93

CPOOL-Verbindungsparameter
 Version 12.0.0, neue Funktion,128

CPUs
 Version 16.0, Verhaltensänderung,32

CRC32, Algorithmus
 Version 12.0.0, Erweiterung,134

CREATE CERTIFICATE-Anweisung
 Version 16.0, neue Funktion,21

CREATE DATABASE-Anweisung
 Version 16.0, Erweiterung,24,27,28
 Version 16.0, Verhaltensänderung,32

CREATE DECRYPTED DATABASE-Anweisung
 Version 11.0.1, neue Funktion,201
 Version 16.0, Erweiterung,24

- Version 16.0, Verhaltensänderung,32
- CREATE DECRYPTED FILE-Anweisung
 - Version 16.0, Erweiterung,24
 - Version 16.0, Verhaltensänderung,32
- CREATE ENCRYPTED DATABASE-Anweisung
 - Version 11.0.1, neue Funktion,201
 - Version 16.0, Erweiterung,24
 - Version 16.0, Verhaltensänderung,32
- CREATE ENCRYPTED FILE-Anweisung
 - Version 16.0, Erweiterung,24
 - Version 16.0, Verhaltensänderung,32
- CREATE ENCRYPTED TABLE-Anweisung
 - Version 16.0, Verhaltensänderung,32
- CREATE EVENT-Anweisung
 - Version 11.0.0, Erweiterung,239
 - Version 16.0, Erweiterung,3,21
 - Version 16.0, neue Funktion,11
- CREATE EXISTING TABLE-Anweisung
 - Version 12.0.1, Erweiterung,88
- CREATE EXTERNLOGIN-Anweisung
 - Version 16.0, Verhaltensänderung,32
- CREATE FUNCTION-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,139
 - Version 16.0, Erweiterung,22
- CREATE INDEX-Anweisung
 - Version 12.0.0, Erweiterung,136,188
 - Version 16.0, Erweiterung,22
- CREATE LDAP SERVER-Anweisung
 - Version 16.0, neue Funktion,6
- CREATE LOGIN POLICY-Anweisung
 - Version 11.0.0, neue Funktion,236
 - Version 16.0, Erweiterung,2,6
- CREATE MATERIALIZED VIEW-Anweisung
 - Version 11.0.0, Erweiterung,238
 - Version 11.0.1, Erweiterung,202
- CREATE MIRROR SERVER-Anweisung
 - Version 12.0.0, neue Funktion,123
- CREATE PROCEDURE-Anweisung
 - Version 11.0.1, Erweiterung,202,203
 - Version 12.0.0, Erweiterung,139
 - Version 12.0.0, Verhaltensänderung,151
 - Version 12.0.1, Erweiterung,88
 - Version 16.0, Erweiterung,22
- CREATE PUBLICATION-Anweisung
 - Version 12.0.0, Erweiterung,137
- CREATE ROLE-Anweisung
 - Version 16.0, neue Funktion,3,4
- CREATE SEQUENCE-Anweisung
 - Version 12.0.0, neue Funktion,124
- CREATE SERVER-Anweisung
 - Version 12.0.0, Erweiterung,136
 - Version 12.0.1, Erweiterung,88
 - Version 16.0, Erweiterung,22
- CREATE SPATIAL REFERENCE SYSTEM-Anweisung
 - Version 12.0.0, neue Funktion ,119
- CREATE SPATIAL UNIT OF MEASURE-Anweisung
 - Version 12.0.0, neue Funktion,119
- CREATE SYNCHRONIZATION PROFILE-Anweisung
 - Version 11.0.0, Erweiterung, UltraLite,270
 - Version 12.0.0, neue Funktion,136
- CREATE SYNCHRONIZATION SUBSCRIPTION-Anweisung
 - Version 12.0.0, Erweiterung,159
 - Version 12.0.0, Erweiterung, MobiLink,163
- CREATE TABLE-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,137
- CREATE TEMPORARY TRACE EVENT SESSION-Anweisung
 - Version 16.0, neue Funktion,22
- CREATE TEMPORARY TRACE EVENT-Anweisung
 - Version 16.0, neue Funktion,22
- CREATE TEXT CONFIGURATION-Anweisung
 - Version 11.0.0, neue Funktion,237
- CREATE TEXT INDEX-Anweisung
 - Version 11.0.0, neue Funktion,237
 - Version 12.0.0, Erweiterung,137
 - Version 16.0, Erweiterung,22
- CREATE TRIGGER-Anweisung
 - Fehlerbehandlung, Upgrades auf Version 16,306
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,139
- CREATE USER-Anweisung
 - Version 11.0.0, neue Funktion,236
- CREATE VARIABLE-Anweisung
 - Version 12.0.0, Erweiterung,137,139
- CREATE VIEW-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,139
 - Version 16.0, Verhaltensänderung,29,36
- createcert, Dienstprogramm

Version 16.0, Erweiterung, 17
 createkey-Dienstprogramm
 Version 11.0.0, neue Funktion, 267
 CROSS APPLY-Klausel
 Version 11.0.0, neue Funktion, 239
 CURRENT UTC TIMESTAMP-Spezialwert
 Version 12.0.0, Verhaltensänderungen, 149
 current_download_row_count
 Version 16.0, Erweiterung, UltraLite C/C++, 61
 Version 16.0, Erweiterung, UltraLite.NET, 61
 CurrentMultiProgrammingLevel, Eigenschaft
 Version 12.0.0, neue Funktion, 133
 Cursor
 Version 12.0.0, Erweiterung, 133
 Cursor, Eigenschaft
 Version 12.0.1, neue Funktion, 93
 CursorOpen, Eigenschaft
 Version 12.0.1, neue Funktion, 93
 CustDB
 Version 11.0.0, Verhaltensänderung, UltraLite, 276
 CustDB, UltraLite-Beispiel
 Version 11.0.0, Verhaltensänderung, 276
 CustDB-Beispiel
 Version 16.0, entfernte Funktion, UltraLite ,65
 Version 16.0, Erweiterung, UltraLiteJ, 64

D

Daemon
 Version 12.0.0, Verhaltensänderung, 118
 DatabaseInfo.getNumberRowsToUpload-Methode
 [UltraLiteJ]
 Version 12.0.1, Erweiterung, 81
 DatabaseInfo.getPageSize-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung, 81
 DatabaseInfo.getRelease-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung, 81
 DatabaseManager.createFileTransfer-Methode
 [UltraLiteJ]
 Version 12.0.1, Erweiterung, 78
 DatabaseMetaData
 Version 16.0, Verhaltensänderung, 36
 DatabaseName-Verbindungsparameter
 Version 12.0.1, Verhaltensänderung,
 OnDemand, 74
 DATEADD-Funktion
 Version 12.0.0, Erweiterung, 135
 DATEDIFF-Funktion

 Version 12.0.0, Erweiterung, 135
 Version 12.0.0, Verhaltensänderung, 152
 Dateien
 Version 11.0.0, neue Funktion, Abfragen, 223
 Dateiformate
 Upgrade, 303
 Daten entladen
 Version 12.0.0, Erweiterung, 130
 DATENAME-Funktion
 Version 12.0.0, Erweiterung, 135
 Datenbank dokumentieren
 Version 12.0.0, Erweiterung, 188
 Datenbank erstellen, Assistent
 Version 11.0.0, Erweiterung, 281
 Datenbank-Worker-Threads
 Version 12.0.1, neue Funktion, MobiLink, 97
 Datenbankaufräumvorgang
 Version 12.0.0, Verhaltensänderung, 152
 Version 12.0.1, Verhaltensänderung, 90
 Datenbankdateiformat
 Upgrade, 303, 310
 Datenbankdokumentation erstellen
 Version 11.0.0, neue Funktion, 278
 Datenbankeigenschaften
 Version 11.0.0, neue Funktion, 232
 Version 12.0.0, neue Funktion, 92, 132
 Version 12.0.1, Erweiterungen, 76
 Version 16.0, Erweiterungen, 19
 Datenbanken
 in Sybase Central neu aufbauen, 300
 mit dem Dienstprogramm dbunload neu
 aufbauen, 302
 Upgrade, 295
 Upgrade des Dateiformats von Version 10.0.0 und
 früher, 310
 Upgrade des Dateiformats von Version 9 und
 früher, 303
 Upgrade in einem
 Datenbankspiegelungssystem, 320
 Upgrade von Version 10 und höher mit
 dbupgrad, 312
 Upgrade von Version 10 und höher mit Sybase
 Central, 311
 Upgrade von Version 10 und später mit ALTER
 DATABASE, 313
 vergleichen, Version 12.0.1, neue Funktion, 75
 Version 9 und früher mit dbunload neu
 aufbauen, 309

- Version 9 und früher neu aufbauen,307
- Datenbanken vergleichen, Fenster
 - Version 12.0.1, neue Funktion,75
- Datenbankgrößen
 - Version 11.0.0, Erweiterung,247
- Datenbankoptionen
 - Version 11.0.0, Erweiterung,221
 - Version 12.0.1, Verhaltensänderung, OnDemand,75
 - Version 16.0, Erweiterungen,18
- Datenbankservereigenschaften
 - Version 12.0.0, neue Funktion,133
 - Version 12.0.1, Erweiterungen,76
 - Version 12.0.1, neue Funktion,93
 - Version 16.0, Erweiterungen,19
- Datenbankservermeldungen
 - Version 12.0.0, Erweiterung,197
- Datenbankspiegelung
 - Supportpakete übernehmen,318
 - Upgrade,314
 - Upgrade (Datenbankserver) auf Minor Releases,316
 - Upgrade der Datenbanken,320
 - Version 11.0.0, Erweiterung,223
 - Version 11.0.0, Verhaltensänderung,254
 - Version 11.0.1, Verhaltensänderung,209
 - Version 12.0.0, Erweiterung,123
 - Version 12.0.0, Verhaltensänderung,151
 - Version 12.0.1, Erweiterung,76
- Datenbanktools, Importbibliotheken
 - Version 11.0.0, neue Funktion,253
- Datenbankvalidierung unter Android
 - Version 16.0, Erweiterung, UltraLiteJ,63
- Datenbankverschlüsselung
 - Version 11.0.0, Erweiterung,225
- Datenquellen
 - Version 12.0.0, Erweiterung,128
- Datenskripten
 - Version 12.0.0, Verhaltensänderung, MobiLink,168
 - Version 16.0, neue Funktion, MobiLink,40
- Datentypen
 - Version 12.0.0, neue Funktion,140
 - Version 16.0, Erweiterungen,24
- Datentypkonvertierungen
 - Version 12.0.0, Erweiterung,146
- DATEPART-Funktion
 - Version 12.0.0, Erweiterung,135
- DATETIMEOFFSET, Datentyp
 - Version 12.0.0, neue Funktion,140
- DateTimeWithTimeZone-Klasse
 - Version 12.0.0, neue Funktion, MobiLink,161
- Datumsangaben
 - Version 12.0.0, Erweiterung,135
- DB_EXTENDED_PROPERTY-Funktion
 - Version 12.0.0, Erweiterung,135
 - Version 12.0.1, Erweiterung,86
 - Version 16.0, Verhaltensänderung,33
- DB_NAME-Funktion
 - Version 16.0, Verhaltensänderung,33
- DB_PROPERTY-Funktion
 - Version 16.0, Verhaltensänderung,33
- db_publisher-Eigenschaft
 - Version 16.0, neue Funktion,3
- DBA
 - Version 12.0.1, Erweiterung,75
- DBA-Berechtigung
 - Version 16.0, nicht mehr empfohlen,37
- dBase-Formate, INPUT-Anweisung
 - Version 11.0.0, entfernt, Unterstützung,284
- dBase-Formate, OUTPUT-Anweisung
 - Version 11.0.0, entfernt, Unterstützung,284
- dbbackup, Dienstprogramm
 - Version 16.0, Erweiterung,14
- dbcapi.dll
 - Version 11.0.0, neue Funktion,240
- dbconsole, Dienstprogramm
 - Version 11.0.0, Erweiterung,285
- dbeng12
 - Version 12.0.0, Verhaltensänderung,118,149
 - Version 12.0.1, Verhaltensänderung,94
- dbfhide, Dienstprogramm
 - Version 12.0.0, Erweiterung,129
- dbinfo, Dienstprogramm
 - Version 12.0.0, Erweiterung,147
- dbinit, Dienstprogramm
 - Version 16.0, Erweiterung,27,28
- dbinit-Dienstprogramm
 - Version 12.0.0, Verhaltensänderung,153
- dbisql, Dienstprogramm
 - Version 16.0, neue Funktion,69
- dbisql-Dienstprogramm
 - Version 11.0.0, Verhaltensänderung,259
- dbisql.exe
 - Version 11.0.0, Verhaltensänderung,285
- dbisqlc, Dienstprogramm

Version 11.0.0, Verhaltensänderung,284
 dbisqlc-Dienstprogramm
 Version 11.0.1, Verhaltensänderung,209
 dbisqlg
 Version 11.0.0, umbenannt,285
 dbngen12.dll, Registrierungseintrag
 Version 12.0.0, Erweiterung,147
 DBLIB
 Version 11.0.0, neue Funktion,253
 dblibtb.dll
 Version 11.0.0, neue Funktion,253
 dblibtw.dll
 Version 11.0.0, neue Funktion,253
 dblic, Dienstprogramm
 Version 12.0.0, Erweiterung,129
 Version 12.0.1, Erweiterung,95
 dblog, Dienstprogramm
 Version 12.0.0, Verhaltensänderung,118
 dbltm, Dienstprogramm
 in Version 12.0.0 nicht unterstützt,117
 Version 12.0.0, Verhaltensänderung,118
 dbmlsync, APIs für .NET und C++
 Version 11.0.0, neue Funktion,266
 dbmlsync, Dienstprogramm
 Version 12.0.0, Verhaltensänderung,118
 dbmlsync, Integrationskomponente
 Version 11.0.0, nicht mehr empfohlen,268
 Version 12.0.0, entfernt,172
 dbmlsync, Meldungslog-Scan
 Version 11.0.0, Erweiterung,267
 dbmlsync, Option -sp
 Version 11.0.0, neue Funktion,266
 dbmlsync, StreamCompression-Option
 Version 11.0.0, nicht unterstützt,268
 dbmlsync-Nachrichtenlog
 Version 12.0.1, Erweiterung, MobiLink,84
 dbmodenv
 Version 11.0.1, Verhaltensänderung,209
 dbns12, Dienstprogramm
 Version 12.0.0, Verhaltensänderung,118
 dbremote, Dienstprogramm
 Version 12.0.0, Verhaltensänderung,118
 DBSC_CancelRet-Enumeration
 Version 12.0.0, neue Funktion, MobiLink,164
 DBSC_ERR_ACTIVE_SYNC_NOT_CANCELED-Mitglied
 Version 12.0.0, neue Funktion, MobiLink,165
 DBSC_ERR_DEAD_SERVER-Mitglied

Version 12.0.0, neue Funktion, MobiLink,165
 DBSC_Event-Struktur
 Version 16.0, neue Funktion, Dbmlsync-.NET-API,46
 Version 16.0, neue Funktion, Dbmlsync-C++-API,45
 DBSC_EVENTTYPE_DOWNLOAD_COMMITTED
 Version 16.0, neue Funktion, Dbmlsync-.NET-API,46
 Version 16.0, neue Funktion, Dbmlsync-C++-API,45
 DBSC_EVENTTYPE_DOWNLOAD_START-Ereignis
 Version 16.0, neue Funktion, Dbmlsync-.NET-API,46
 Version 16.0, neue Funktion, Dbmlsync-C++-API,46
 DBSC_EVENTTYPE_UPLOAD_SENT-Ereignis
 Version 16.0, neue Funktion, Dbmlsync-.NET-API,46
 Version 16.0, neue Funktion, Dbmlsync-C++-API,46
 DBSC_EVENTTYPE_UPLOAD_START-Ereignis
 Version 16.0, neue Funktion, Dbmlsync-.NET-API,46
 Version 16.0, neue Funktion, Dbmlsync-C++-API,45
 dbspaces
 Version 11.0.0, Verhaltensänderung,254,258
 dbspawn, Dienstprogramm
 Version 16.0, Erweiterung,17
 dbsrv12
 Version 12.0.0, Verhaltensänderung,118
 dbstats, Dienstprogramm
 Version 12.0.0, neue Funktion,145
 dbsupport, Dienstprogramm
 Version 11.0.0, Erweiterung,287
 Version 12.0.0, Erweiterung,130
 Version 12.0.0, Verhaltensänderung,118
 Version 16.0, Verhaltensänderung,32
 dbsvc, Dienstprogramm
 Version 11.0.1, Erweiterung,200
 Version 12.0.0, Erweiterung,130
 Version 12.0.0, Verhaltensänderung,118,156
 Version 16.0, Verhaltensänderung,10
 dbsvc-Dienstprogramm
 Version 12.0.0, neue Funktion,145
 dbtltb.dll

- Version 11.0.0, neue Funktion,253
- dbtlstw.dll
 - Version 11.0.0, neue Funktion,253
- dbtran, Dienstprogramm
 - Version 12.0.0, Verhaltensänderung,117
- dbtran-Dienstprogramm
 - Version 11.0.0, Erweiterung,227
- DBTYPE_DBTIME
 - Borland Delphi,39
- DBTYPE_DBTIME2
 - Version 16.0, Erweiterung,25
- DBTYPE_DBTIMESTAMPOFFSET
 - Borland Delphi,39
 - Version 12.0.1, Erweiterung,89
- dbunload, Dienstprogramm
 - Datenbanken der Version 10 und höher neu aufbauen,302
 - Datenbanken der Version 9 und früher neu aufbauen,309
 - Version 11.0.0, Erweiterung,227
 - Version 11.0.1, Erweiterung,200
 - Version 16.0, Erweiterung,17
- dbunload-Dienstprogramm
 - Fehlerbehandlung, Upgrades auf Version 16,306
 - Version 11.0.0, Verhaltensänderung,257
- dbupgrad, Dienstprogramm
 - Upgrade von Datenbanken der Version 10 und höher,312
 - Version 16.0, Erweiterung,13,29
- dbupgrad-Dienstprogramm
 - Version 11.0.1, Verhaltensänderung,209
- dbvalid-Dienstprogramm
 - Version 11.0.0, Erweiterung,227
- dbxtract, Dienstprogramm
 - Version 11.0.0, Erweiterung,269
 - Version 12.0.0, Erweiterung,129,172
- dbxtract-Dienstprogramm
 - Version 11.0.0, Verhaltensänderung,257
- DCX
 - Version 11.0.1, neue Funktion,216
 - Version 12.0.0, Erweiterung,196
 - Version 12.0.1, Erweiterung,116
- Deadlock-Systemereignis
 - Version 11.0.0, neue Funktion,248
- Deaktivieren, Bearbeitung von Ergebnismengen
 - Version 11.0.0, neue Funktion,280
- Deaktivieren, Tabellenbearbeitung
 - Version 11.0.0, neue Funktion,279
- DEAULT VALUES-Klausel
 - Version 11.0.1, neue Funktion,201
- Debug-Modus
 - Version 11.0.0, Erweiterung,281
- DECLARE CURSOR-Anweisung
 - Version 12.0.0, Verhaltensänderung,151
- DECLARE LOCAL TEMPORARY TABLE-Anweisung
 - Fehlerbehandlung, Upgrades auf Version 16,306
- DECLARE-Anweisung
 - Version 12.0.0, Erweiterung,137
- DECRYPT-Funktion
 - Version 11.0.0, Erweiterung,225
 - Version 16.0, Erweiterung,14
 - Version 16.0, Verhaltensänderung,32
- defaultShowMultipleResultSets
 - Version 16.0, neue Funktion,70
- defaultShowResultsForAllStatements
 - Version 16.0, neue Funktion,70
- DELETE-Anweisung
 - Version 11.0.0, Erweiterung,239
 - Version 11.0.0, Verhaltensänderung,257
 - Version 11.0.1, Erweiterung,204
 - Version 12.0.0, Erweiterung,140
 - Version 12.0.0, Verhaltensänderung,149
 - Version 12.0.1, Erweiterung,88
- delete_old_logs, Option
 - Version 12.0.0, Verhaltensänderung,118
- Delphi (Borland)
 - Version 16.0, nicht unterstützt,39
- demo.db
 - Version 11.0.0, Erweiterung,287
- Deployment-Assistent
 - Version 11.0.0, Erweiterung,281
 - Version 12.0.0, Erweiterung,117
- DER-kodierte Schlüssel
 - Version 12.0.1 neue Funktion, MobiLink,98
- DESCRIBE-Anweisung
 - Version 11.0.0, Erweiterung,283
 - Version 16.0, Verhaltensänderung,31
- Diagnoseprotokollierung
 - Version 16.0, Verhaltensänderung,31
- Dialogfreie Installation
 - Version 12.0.0, neue Funktion,144
- Dialogfreie Installationen
 - Version 11.0.0, Verhaltensänderung,261
- Dienste
 - Version 11.0.1, Erweiterung,200

- Version 12.0.0, Erweiterung,130,145
- Version 12.0.0, Verhaltensänderung,118
- Dienste-Dienstprogramm (dbsvc)
 - Version 11.0.1, Erweiterung,200
 - Version 12.0.0, Erweiterung,130
 - Version 12.0.0, Verhaltensänderung,156
- Dienstprogramm für die Zertifikatserstellung (viewcert)
 - Version 16.0, Erweiterung,17
- Dienstprogramm für Dienste
 - Version 12.0.0, Verhaltensänderung,118
- Dienstprogramm für Serverstart im Hintergrund (dbspawn)
 - Version 16.0, Erweiterung,17
- Dienstprogramm für Supportanfragen (dbsupport)
 - Version 16.0, Verhaltensänderung,32
- Dienstprogramm Initialisierung (dbinit)
 - Version 16.0, Erweiterung,27,28
- Dienstprogramm Transaktionslogübersetzung
 - Version 12.0.0, Verhaltensänderung,117
- Dienstprogramm zum Anzeigen von Zertifikaten (viewcert)
 - Version 16.0, Erweiterung,17
- Dienstprogramm zum Entladen (dbunload)
 - Datenbanken der Version 10 und höher neu aufbauen,302
 - Datenbanken der Version 9 und früher neu aufbauen,309
 - Version 16.0, Erweiterung,17
- Dienstprogramm zum Upgrade (dbupgrad)
 - Version 16.0, Erweiterung,13,29
 - Version 16.0, Verhaltensänderung,32
- Dienstprogramm-Datenbank
 - Version 11.0.0, Verhaltensänderung,253
- Dienstprogrammdatenbank
 - Version 12.0.0, Erweiterung,130
- Dienstprogramme
 - Version 12.0.0, Erweiterung,117
 - Version 12.0.0, Verhaltensänderung,118
 - Version 16.0, Verhaltensänderung, UltraLite,66
- Direkte Page Scans
 - Version 11.0.0, neue Funktion, UltraLite-Tabellen-API,274
- disk_sandbox-Eigenschaft
 - Version 16.0, neue Funktion,17
- disk_sandbox-Option
 - Version 16.0, neue Funktion,16
- DiskRead-Eigenschaft
 - Version 11.0.0, neue Funktion,258
- DiskReadEng-Eigenschaft
 - Version 11.0.0, umbenannt,258
- DiskReadHint-Eigenschaft
 - Version 11.0.0, neue Funktion,258
- DiskReadHintPages-Eigenschaft
 - Version 11.0.0, neue Funktion,258
- DiskReadHintScatterLimit-Eigenschaft
 - Version 11.0.0, neue Funktion,258
- DiskReadWorkTable-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- DiskRetryRead-Eigenschaft
 - Version 11.0.0, neue Funktion,231
- DiskRetryReadScatter
 - Version 11.0.0, neue Funktion,231
- DiskRetryReadScatter-Eigenschaft
 - Version 11.0.0, neue Funktion,232
- DiskRetryWrite-Eigenschaft
 - Version 11.0.0, neue Funktion,231
- DiskSandbox-Eigenschaft
 - Version 16.0, neue Funktion,17
- DiskSyncRead-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- DiskSyncWrite-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- DiskWaitRead-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- DiskWaitWrite-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- DiskWriteHint-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- DiskWriteHintPages-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- divide_by_zero_error, Option
 - in Version 12.0.0 unterstützt,152
- divide_by_zero_error-Option
 - Version 11.0.0, nicht unterstützt,260
- DLL-Protokolloption
 - Version 11.0.0, nicht unterstützt,262
- DML
 - Version 12.0.0, neue Funktion, auswählen,126
- DocCommentXchange (DCX)
 - Version 11.0.1, neue Funktion,216
- Dokumentation erstellen, Datenbank
 - Version 11.0.0, neue Funktion,278
- Dokumentation, Erweiterungen
 - Version 11.0.0,286
 - Version 11.0.1,217

- Version 12.0.0,196
- Version 12.0.1,116
- Doppelkontrolle, Kennwort
 - Version 16.0, neue Funktion,13
- Download-Löschen-Teilmenge
 - Version 12.0.0, neue Funktion, MobiLink,157
- Downloadbestätigung
 - Version 11.0.0, Verhaltensänderung,268
- DriveBus, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- DriveModel, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- driver-Verbindungsparameter
 - Version 11.0.0, Erweiterung,242
- DROP CERTIFICATE-Anweisung
 - Version 16.0, neue Funktion,21
- DROP DATABASE-Anweisung
 - Version 16.0, Erweiterung,24
 - Version 16.0, Verhaltensänderung,32
- DROP EVENT-Anweisung
 - Version 11.0.0, Erweiterung,239
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,138
- DROP FUNCTION-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,138
- DROP INDEX-Anweisung
 - Version 12.0.0, Erweiterung,138
- DROP LDAP SERVER-Anweisung
 - Version 16.0, neue Funktion,6
- DROP LOGIN POLICY-Anweisung
 - Version 11.0.0, neue Funktion,236
- DROP MATERIALIZED VIEW-Anweisung
 - Version 12.0.0, Erweiterung,138
- DROP MIRROR SERVER-Anweisung
 - Version 12.0.0, neue Funktion,123
- DROP PROCEDURE-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,138
- DROP PUBLICATION-Anweisung
 - Version 12.0.0, Erweiterung,138
- DROP REMOTE CONNECTION-Anweisung
 - Version 16.0, neue Funktion,22
- DROP ROLE-Anweisung
 - Version 16.0, neue Funktion,4
- DROP SEQUENCE-Anweisung
 - Version 12.0.0, neue Funktion,124

- DROP SPATIAL REFERENCE SYSTEM-Anweisung
 - Version 12.0.0, neue Funktion,119
- DROP SPATIAL UNIT OF MEASURE-Anweisung
 - Version 12.0.0, neue Funktion,119
- DROP SYNCHRONIZATION PROFILE-Anweisung
 - Version 11.0.0, Erweiterung, UltraLite,270
- DROP TABLE-Anweisung
 - Version 11.0.1, Erweiterung,202
- DROP TEXT CONFIGURATION-Anweisung
 - Version 11.0.0, neue Funktion,237
- DROP TEXT INDEX-Anweisung
 - Version 11.0.0, neue Funktion,237
- DROP TRACE EVENT SESSION-Anweisung
 - Version 16.0, neue Funktion,23
- DROP TRACE EVENT-Anweisung
 - Version 16.0, neue Funktion,23
- DROP TRIGGER-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,138
- DROP USER-Anweisung
 - Version 11.0.0, neue Funktion,236
- DROP VARIABLE-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,138
- DROP VIEW-Anweisung
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, Erweiterung,138
- DropBadStatistics, Eigenschaft
 - Version 12.0.0, neue Funktion,134
- DropUnusedStatistics, Eigenschaft
 - Version 12.0.0, neue Funktion,134
- Dynamische Cachedimensionierung
 - Version 12.0.1, Verhaltensänderung,94

E

- E-Mail-Versand
 - Monitor Version 12.0.1, neue Funktion,114,115
 - Monitor, Version 16.0, Erweiterung,71
- e2ee_type-Protokolloption
 - Version 16.0, nicht mehr empfohlen, MobiLink,53
- ECC-Option
 - Version 16.0, nicht mehr empfohlen,7
- ECC-Unterstützung
 - Version 16.0, entfernte Funktion, Relay Server,56
 - Version 16.0, entfernte Funktion, UltraLite ,65
- ECC-Zertifikate

Version 16.0, nicht mehr empfohlen,7
 echo, Option
 Version 12.0.0, Verhaltensänderung,193
 Eigenschaften der MobiLink-Server-Befehlszeile,
 Fenster
 Version 16.0, neue Funktion, MobiLink-Plug-In,48
 Eigenschaftsfenster
 Version 11.0.0, nicht mehr empfohlen, UltraLite-
 Funktionen,281
 Eingabehilfen
 Version 12.0.0, Erweiterung,187
 Einzelne Anwendung, Zugriff
 Version 16.0, neue Funktion, UltraLite,59
 Einzelschritt
 Version 11.0.0, neue Funktion,280
 Elevate-Verbindungsparameter
 Version 11.0.0, neue Funktion,224
 Elliptische Kurve, Zertifikate
 Version 16.0, nicht mehr empfohlen,7
 Embedded SQL
 Version 12.0.0, Verhaltensänderung,151
 Embedded SQL, Importbibliotheken
 Version 11.0.0, neue Funktion,253
 enable status-Eigenschaft aktivieren
 Version 12.0.0, neue Funktion, MobiLink,165
 EnableFlush, Registrierungseintrag
 Version 12.0.0, neue Funktion,147
 ENAME-Protokolloption
 Version 11.0.0, nicht unterstützt,260
 ENCRYPT-Funktion
 Version 11.0.0, Erweiterung,225
 Version 16.0, Erweiterung,14
 Version 16.0, Verhaltensänderung,32
 encrypt_aes_random_iv-Option
 Version 11.0.0, nicht unterstützt,262
 ENCRYPT_PASSWORD-Verbindungseigenschaft
 Version 11.0.0, neue Funktion,226
 ENCRYPTION-Verbindungsparameter
 Version 16.0, nicht mehr empfohlen,7
 end_synchronization-Skripten
 Version 16.0, Verhaltensänderung,49
 Ende-zu-Ende-Verschlüsselung
 Version 11.0.0, Erweiterung, UltraLite,273
 Version 11.0.0, neue Funktion,267
 Endianness
 Version 11.0.0, Erweiterung,246
 Version 11.0.1, Verhaltensänderung,209
 ENG-Verbindungsparameter
 Version 12.0.0, nicht mehr empfohlen,155
 EngineName-Verbindungsparameter
 Version 12.0.0, nicht mehr empfohlen,155
 Entfernte Aufgabe
 Version 12.0.1, neue Funktion, MobiLink,98
 Version 16.0, Erweiterung, MobiLink-Plug-In,49
 Entfernte Datenbanken
 Upgrade,335
 Version 12.0.0, Erweiterung,136
 Entfernte IDs
 Version 12.0.1, neue Funktion,87
 Entfernte Tabellen, Eigentümer
 Version 16.0, Änderungen, MobiLink-Plug-In,51
 Entferntes Schema, Name
 Version 12.0.1, neue Funktion, MobiLink,100
 Entity Framework-Unterstützung
 Version 11.0.1, neue Funktion,203
 Entity Framework 5.0-Unterstützung
 Version 16.0, neue Funktion,24
 Entity Framework, Unterstützung
 Version 12.0.1, neue Funktion,77
 Entlade-Dienstprogramm (dbunload)
 Version 11.0.0, Erweiterung,227
 Version 11.0.1, Erweiterung,200
 Version 12.0.0, Erweiterung,129,130
 Entladen
 Daten aus Tabellen mit Autoinkrement-Spalten,300
 Version 11.0.0 neue Funktion, in eine Variable,222
 Version 12.0.0, Erweiterung,130
 Entladen-Dienstprogramm (dbunload)
 Version 11.0.0, Verhaltensänderung,257
 environment.plist
 Version 11.0.1, Verhaltensänderung,209
 ER-Diagramm
 Version 16.0, Erweiterung,68
 Ereignisprotokollierung
 Version 16.0, neue Funktion,10
 Version 16.0, neue Funktion, MobiLink,41
 Ereignisprotokollierung, Datendateien
 Version 16.0, neue Funktion,10
 Ereignisse
 Version 11.0.1, Erweiterung,214
 Ergebnismengen
 Version 11.0.0, neue Funktion, mit Schreibschutz
 versehen,279
 Version 16.0, Erweiterung,70
 Ergebnisse

- Version 11.0.0, neue Funktion, mit Schreibschutz versehen, 279
- Ergebnisse, Fensterausschnitt
 - Version 11.0.0, Erweiterungen, 282
 - Version 12.0.0, Erweiterungen, 191
- Erzwungener Konflikt, Modus
 - Version 16.0, entfernt, MobiLink, 53
- Erzwungener Konfliktmodus
 - Version 12.0.0, nicht mehr empfohlen, MobiLink, 172
- ESC-Schlüssel
 - Version 12.0.0, Verhaltensänderung, 193
- Escape-Verbindungsparameter
 - Version 12.0.0, neue Funktion, 128
- Escapezeichen
 - Erweiterung für Version 12.0.0, Dienstprogramme, 117
 - Version 12.0.1, Verhaltensänderung, 94
- ESTIMATE_SOURCE-Funktion
 - Version 12.0.0, Erweiterung, 146
- ETD-Dateien
 - Version 16.0, neue Funktion, 10
- ETD-Dateien, Dienstprogramm für die Verwaltung (dbmanageetd)
 - Version 16.0, neue Funktion, 10
- EVENT_PARAMETER-Funktion
 - Version 11.0.0, Erweiterung, 234
- EventTypeDesc
 - Version 11.0.0, neue Funktion, 231
- EventTypeName
 - Version 11.0.0, neue Funktion, 231
- EXCEL Format, INPUT-Anweisung
 - Version 11.0.0, entfernt, Unterstützung, 284
- EXCEL-Format, OUTPUT-Anweisung
 - Version 11.0.0, entfernt, Unterstützung, 284
- EXCEPT-Anweisung
 - Version 11.0.0, Verhaltensänderung, 257
- EXCEPT-Klausel
 - Version 11.0.0, Erweiterung, 239
- executeBatch
 - Version 12.0.0, Erweiterung, 142
- Export-Assistent
 - Version 11.0.0, Erweiterung, 284
- ExtendedName-Protokolloption
 - Version 11.0.0, nicht unterstützt, 260
- extern_login_credentials-Eigenschaft
 - Version 16.0, neue Funktion, 19
- extern_login_credentials-Option

- Version 16.0, neue Funktion, 18
- Externe Entladevorgänge
 - Version 11.0.0, Erweiterung, 248
- Externe Logins
 - Version 12.0.0, Verhaltensänderung, 152
- Extraktionsdienstprogramm (dbxtract)
 - Version 11.0.0, Erweiterung, 269
 - Version 11.0.0, Verhaltensänderung, 258
 - Version 12.0.0, Erweiterung, 172

F

- Failover
 - Version 11.0.0, neue Funktion, MobiLink-Serverfarm, 264
- FAQ
 - Version 12.0.0, neue Funktion, 196
- Fasern
 - Version 12.0.0, Verhaltensänderung, 148
- Favoritenliste
 - Version 11.0.0, neue Funktion, 280
 - Version 12.0.0, Erweiterung, 191
- Fehler-Berichterstellung
 - Version 11.0.0, Erweiterung, 287
- Fehler-Berichtserstellung
 - Version 12.0.0, Erweiterung, 130
- Fehlerbehandlung
 - Aggregatfunktion, 323
 - Datenbank-Upgrade auf Version 16, 305
 - JDBC, 322
 - Version 12.0.0, Erweiterung, 197
- Fehlerbehandlungsroutine
 - Version 16.0, Verhaltensänderung, 50
- Fehlercodes
 - Version 12.0.0, neue Funktion, Relay Server, 166
- Fehlermeldungen
 - Version 12.0.0, neue Funktion, Relay Server, 166
 - Version 12.0.1, neue Funktion, UltraLite, 110
- Fehlersuche
 - Befehlszeilen-Dienstprogramme, 298
 - UltraLite-Befehlszeilen-Dienstprogramme, 337
- Fehlgeschlagene Neuerstellung
 - Datenbank-Upgrade auf Version 16, 305
- Ferndatenzugriff
 - Version 11.0.0, Verhaltensänderung, 259
 - Version 12.0.0, Erweiterung, 146
 - Version 12.0.1, Erweiterung, 88
- Ferndatenzugriffsklasse

Version 16.0, Erweiterung,11
 FileTransfer-Schnittstelle [UltraLiteJ]
 Version 12.0.1, Erweiterung,78
 FileTransferProgressData-Schnittstelle [UltraLiteJ]
 Version 12.0.1, Erweiterung,78
 FileTransferProgressListener-Schnittstelle [UltraLiteJ]
 Version 12.0.1, Erweiterung,78
 fill_sqlda_ex-Funktion
 Version 12.0.0, neue Funktion,144
 FiniIdentifySimulatedClient, Callback-Funktion
 Version 12.0.1, Verhaltensänderung, MobiLink (mlgenreplayapi),102
 FIPS
 Version 11.0.0, Erweiterung,225
 Version 12.0.0, Erweiterung,129
 FIRST_VALUE-Funktion
 Version 11.0.1, Erweiterung,201
 float_as_double-Option
 Version 11.0.0, nicht unterstützt,260
 FORCE NO OPTIMIZATION-Klausel
 Version 11.0.1, neue Funktion,204
 FORMAT-Klausel
 Version 11.0.0, nicht mehr empfohlen, FORMAT ASCII,259
 Formdateien
 Version 12.0.1, Erweiterung,85
 Version 12.0.1, Laden von Erweiterungen,85
 FOXPRO-Format, INPUT-Anweisung
 Version 11.0.0, entfernt, Unterstützung,284
 FOXPRO-Format, OUTPUT-Anweisung
 Version 11.0.0, entfernt, Unterstützung,284
 Fragezeichen
 Verwendung in SQL-Skripten, Version 12.0.0, nicht mehr empfohlen,172
 Fragmentierung, Registerkarte
 Version 12.0.0, neue Funktion,188
 Framework 4.5-Unterstützung
 Version 16.0, neue Funktion,24
 FROM-Klausel
 Version 11.0.0, Erweiterung,220
 Version 12.0.0, Erweiterung,126
 Version 12.0.1, Erweiterung,77
 Funktionen
 Version 11.0.1, Erweiterung,203
 Version 12.0.1, Verhaltensänderung, OnDemand,74

G

Gehosteter Relay Server
 Version 11.0.0, neue Funktion ,264
 gencert-Dienstprogramm
 Version 11.0.0, entfernt,288
 generate_next_last_download_timestamp-Skript
 Version 12.0.0, neue Funktion, MobiLink,162
 Geräte
 Version 11.0.0, Erweiterung, UltraLite,273
 Version 12.0.0, Erweiterung, UltraLite,173
 Version 12.0.1, Erweiterung, UltraLite,106
 Version 16.0, Erweiterung, UltraLite,60
 Gesicherte Funktionen
 Version 11.0.0, Verhaltensänderung,262
 Version 16.0, Erweiterung,15
 Version 16.0, Verhaltensänderung,31,69
 Gespeicherte Prozeduren
 Version 12.0.1, Erweiterung,76,77
 Gesundheit und Statistik
 Version 11.0.0, neue Funktion,278
 getAutoCheckpoint-Methode
 Version 16.0, entfernte API, UltraLite ,66
 getDatabaseKey-Methode
 Version 16.0, Verhaltensänderung, UltraLite ,66
 getlastdownloadtime
 Version 11.0.1, Erweiterung,211
 getPageReads()-Methode
 Version 11.0.1, Erweiterung, UltraLiteJ,213
 getPageWrites-Methode
 Version 11.0.1, Erweiterung, UltraLiteJ,213
 getParameter-Methode
 Version 16.0, Erweiterung, UltraLiteJ,62
 getParameterCount-Methode
 Version 16.0, Erweiterung, UltraLiteJ,62
 getParameterType-Methode
 Version 16.0, Erweiterung, UltraLiteJ,62
 getSendColumnNames-Methode
 Version 16.0, Verhaltensänderung, UltraLite,9
 getStreamErrorParameters-Methode
 Version 11.0.1, Erweiterung, UltraLite,213
 getTables
 Version 16.0, Verhaltensänderung,36
 getUpdateCounts
 Version 12.0.0, Erweiterung,142
 Globale Prüfsummen
 Version 12.0.0, Erweiterung,127,140
 Version 12.0.0, Verhaltensänderung,148

- Grafische Pläne
 - Version 16.0, Erweiterung,27
- GRANT CONNECT-Anweisung
 - Version 16.0, Verhaltensänderung,32
- GRANT PUBLISH-Anweisung
 - Version 16.0, Verhaltensänderung,4
- GRANT ROLE
- SYS_REPLICATION_ADMIN_ROLE-Anweisung
 - Version 16.0, neue Funktion,4
- GRANT ROLE SYS_RUN_REPLICATION_ROLE-Anweisung
 - Version 16.0, neue Funktion,4
- GRANT-Anweisung
 - Version 12.0.0, Erweiterung,124
- Grenzen
 - Version 11.0.0 Erhöhungen,248
- GUID-ID
 - Version 11.0.0, Erweiterung, UltraLite,272
- GV_\$TRANSACTION, Oracle-Systemansicht
 - Version 12.0.0, Verhaltensänderung, MobiLink,167

H

- HANA-Ferndatenzugriff
 - Version 16.0, neue Funktion,11
- HANAODBC-Serverklasse
 - Version 16.0, neue Funktion,11
- HasEndianSwapFix-Eigenschaft
 - Version 11.0.0, neue Funktion,232
- HASH-Funktion
 - Version 12.0.0, Erweiterung,134
- Hash-Größen
 - Version 16.0, neue Funktion, UltraLite,58
- HasTornWroteFix-Eigenschaft
 - Version 12.0.0, neue Funktion,132
- header_limit-Option
 - Version 16.0, neue Funktion, MobiLink,42
- HeapsCarver, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- HeapsLocked, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- HeapsQuery, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- HeapsRelocatable, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- HEXTOBIN-Funktion
 - Version 16.0, neue Funktion,21
- Hintergrundsynchronisation
 - Version 11.0.0, Erweiterung, UltraLite,272
 - Version 12.0.0, Erweiterung, MobiLink,162
- Hooks
 - Version 16.0, Verhaltensänderung, MobiLink,51
- Host, Protokolloption
 - Version 12.0.0, Verhaltensänderung,150
- Host-Verbindungsparameter
 - Version 12.0.0, neue Funktion,124
- HTTP
 - Version 11.0.0, Verhaltensänderung,254
- HTTP-Authentifizierung
 - Version 16.0, neue Funktion, UltraLite,58
- HTTP-Dienst
 - Version 12.0.0, Erweiterung,141
- HTTP-Messaging-System
 - Version 12.0.1, Verhaltensänderung,78
- HTTP-Performanceverbesserungen
 - Version 16.0, neue Funktion, MobiLink,45
- HTTP-Server
 - Version 16.0, Verhaltensänderung,31
- http_buffer_responses-Protokolloption
 - Version 12.0.0, neue Funktion, MobiLink,163
- http_connection_pool_basesize, Option
 - Version 12.0.0, neue Funktion,130
- http_connection_pool_timeout-Option
 - Version 12.0.0, neue Funktion,130
- http_password, Option
 - Version 12.0.1, neue Funktion, Relay Server,101
- http_proxy_password, Option
 - Version 12.0.1, neue Funktion, Relay Server,101
- http_proxy_userid, Option
 - Version 12.0.1, neue Funktion, Relay Server,101
- HTTP_RESPONSE_HEADER-Funktion
 - Version 12.0.0, neue Funktion,135
- http_userid, Option
 - Version 12.0.1, neue Funktion, Relay Server,101
- HTTP_VARIABLE-Funktion
 - Version 12.0.0, neue Funktion,135
- HttpAddress-Eigenschaft
 - Version 11.0.0, neue Funktion,231
- HttpAddresses-Eigenschaft
 - Version 11.0.0, neue Funktion,231
- HttpConnPoolCachedCount, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- HttpConnPoolHits, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- HttpConnPoolMisses, Eigenschaft

Version 12.0.0, neue Funktion,132
HttpConnPoolSteals, Eigenschaft
Version 12.0.0, neue Funktion,132
HttpNumActiveReq-Eigenschaft
Version 11.0.0, neue Funktion,231
HttpNumConnections-Eigenschaft
Version 11.0.0, neue Funktion,231
HttpNumSessions-Eigenschaft
Version 11.0.0, neue Funktion,231
HTTPS
Version 11.0.0, Erweiterung,245
HttpsNumActiveReq-Eigenschaft
Version 11.0.0, neue Funktion,231
HttpsNumConnections-Eigenschaft
Version 11.0.0, neue Funktion,231

I

iAnywhere JDBC, Treiber
Version 12.0.0, nicht mehr empfohlen,155
iAnywhere Solutions 12 - Oracle ODBC-Treiber
Version 12.0.0, Erweiterung,159
iAnywhere Solutions Oracle-Treiber
Version 11.0.0, Erweiterung,264
Version 16.0, Verhaltensänderung,7
iastor, Registrierungseintrag
Version 12.0.0, neue Funktion,147
iastorv, Registrierungseintrag
Version 12.0.0, neue Funktion,147
IBM DB2 LUW
MobiLink, konsolidierte Datenbanken,330
Version 16.0, neue Funktion, MobiLink,42
IBM DB2 LUW 8.2
Version 12.0.0, nicht mehr weiterentwickelt,
MobiLink,170
IBM DB2 LUW 9.5-Unterstützung
Version 16.0, entfernt, MobiLink,52
IBM DB2 Mainframe
Version 12.0.0, nicht mehr weiterentwickelt,
MobiLink,170
IdentifySimulatedClient, Callback-Funktion
Version 12.0.1, Verhaltensänderung, MobiLink
(mlgenreplayapi),102
Identitäts-Weiterleitung
Version 12.0.1, Relay Server,84
Identitätsdateien
Version 11.0.0, Verhaltensänderung,288
identity-Synchronisationsparameter

Version 12.0.0, neue Funktion, MobiLink,163
identity_password-Synchronisationsparameter
Version 12.0.0, neue Funktion, MobiLink,163
Idle-Verbindungsparameter
Version 12.0.0, Verhaltensänderung,150
IF EXISTS-Klausel
Version 11.0.1, neue Funktion,202
Version 12.0.0, neue Funktion,138
Version 12.0.0, neue Funktion, UltraLiteJ,175
IF NOT EXISTS-Klausel
Version 11.0.1, neue Funktion,202
Version 12.0.0, neue Funktion,137
Version 12.0.0, neue Funktion, UltraLiteJ,175
IF-Anweisung
Version 11.0.0, Erweiterung,238
Version 11.0.0, Erweiterung, UltraLite,274
IF-Ausdrücke
Version 11.0.0, Erweiterung,238
ignored_deletes
Version 16.0, neue Funktion, UltraLite,57
ignored_updates
Version 16.0, neue Funktion, UltraLite,57
Ignorieren von Skripten
Version 11.0.1, neue Funktion, MobiLink
(mlsrv11),211
IIS
Version 16.0, 32-Bit-Relay Server für Windows
nicht mehr weiterentwickelt,56
IIS 6
Version 12.0.1 schnelle Deployment, Relay
Server,84
IIS 7
Version 12.0.1 schnelle Deployment, Relay
Server,84
IIS 8
Version 16.0, neue Funktion, unterstützt für Relay
Server,53
IMMEDIATE-Klausel
Version 11.0.1, Erweiterung, MESSAGE-
Anweisung,202
Import-Assistent
Version 11.0.0, Erweiterung,284
Version 12.0.1, Erweiterung,85
Version 16.0, Erweiterung,70
INDEX ONLY-Klausel
Version 11.0.0, neue Funktion,238
Index-Hints
Version 12.0.0, Erweiterung,136

- Version 12.0.0, Verhaltensänderung,154
- Indikatorvariablen
 - Version 12.0.0, nicht unterstützt, short int ,155
- Indizes
 - Version 11.0.0, Erweiterung,221
 - Version 11.0.0, Erweiterungen, Performance,246
 - Version 11.0.0, Verhaltensänderung,258
 - Version 12.0.0, Erweiterung,146
 - Version 16.0, Erweiterung,18
 - Version 16.0, Verhaltensänderung,30
- Indizierung, Performanceverbesserungen
 - Version 16.0, neue Funktion, UltraLite,59
- Informationsdienstprogramm (dbinfo)
 - Version 12.0.0, Erweiterung,147
- Initialisierungs-Dienstprogramm (dbinit)
 - Version 12.0.0, Erweiterung,129,148
 - Version 12.0.0, Verhaltensänderung,153
- InList-Algorithmus (IN)
 - Version 11.0.0, Erweiterung,246
- INPUT INTO-Anweisung
 - Version 11.0.0, Verhaltensänderung,283
- INPUT-Anweisung
 - Version 11.0.0, Erweiterung,284
 - Version 11.0.0, Verhaltensänderung,284
 - Version 12.0.0, Erweiterung,194
 - Version 12.0.1, Erweiterung,85
- INSERT-Anweisung
 - mehrzeilig, Erweiterung,139
 - Version 11.0.0, Erweiterung,239,247
 - Version 11.0.0, Verhaltensänderung,257
 - Version 11.0.1, Erweiterung,201
 - Version 12.0.0, Erweiterung,139
- Installationsverzeichnis
 - Version 11.0.0, Verhaltensänderung,288
- INTEGER UNSIGNED-Datentyp
 - Version 16.0, neue Funktion,24
- Integrierter Outbound Enabler
 - Version 12.0.0, neue Funktion, MobiLink,158
- Intel, Speichertreiber
 - Version 12.0.0, Erweiterung,147
- Interactive SQL
 - automatisch vervollständigen,112
 - Textvervollständigung,112
 - Version 11.0.0, Erweiterung,281
 - Version 11.0.0, neue Funktion,279
 - Version 12.0.0, Erweiterung,194
 - Version 12.0.0, neue Funktion,191
 - Version 12.0.1, Erweiterung,85,113
- Version 12.0.1, neue Funktion,112
- Version 16.0, neue Funktionen,69
- Version 16.0, Verhaltensänderung,70
- Interactive SQL, neue Funktionen
 - Version 11.0.0,277
 - Version 11.0.1,214
 - Version 12.0.0,186
 - Version 12.0.1,111
 - Version 16.0,69
- Interactive SQL, Verhaltensänderungen
 - Version 11.0.0,280
 - Version 11.0.1,215
 - Version 12.0.0,192
 - Version 12.0.1,113
- Interactive SQL-Fehlerfenster
 - Version 16.0, Erweiterung,70
- International Resources Development Kit
 - Version 11.0.0, Verhaltensänderung,258
- Interne Entladevorgänge
 - benutzerdefinierte Kollationen bewahren,305
- INTERSECT-Anweisung
 - Version 11.0.0, Verhaltensänderung,257
- INTERSECT-Klausel
 - Version 11.0.0, Erweiterung,239
- INTO LOCAL TEMPORARY TABLE-Klausel
 - Version 11.0.1, neue Funktion,202
- IP, Protokolloption
 - Version 12.0.0, Verhaltensänderung,150
- IP-Adresse
 - Version 12.0.0, Erweiterung,131
- IPAddressMonitorPeriod, Eigenschaft
 - Version 12.0.0, neue Funktion,134
- IPv6
 - Version 12.0.1, Verhaltensänderung,75
- IQJDBC-Serverklasse
 - Version 12.0.0, nicht mehr empfohlen,155
 - Version 16.0, nicht unterstützt,39
- IQODBC-Serverklasse
 - Version 12.0.0, neue Funktion,136
- IRDK
 - Version 11.0.0, Verhaltensänderung,258
- IS DISTINCT FROM, Suchbedingung
 - Version 12.0.0, neue Funktion,136
- IS NOT DISTINCT FROM, Suchbedingung
 - Version 12.0.0, neue Funktion,136
- IS NOT NULL-Suchbedingung
 - Version 12.0.1, Erweiterung,91
- is_online, Variable

Version 12.0.1, neue Funktion, MobiLink,98	Version 11.0.0, Verhaltensänderung,250
IsAesniAvailable-Eigenschaft	ISYSGROLEGRANT
Version 16.0, neue Funktion,19	Version 16.0, neue Funktion,4
IsDebugger-Eigenschaft	ISYSGROLEGRANTEXT
Version 11.0.1, neue Funktion,	Version 16.0, neue Funktion,4
Verbindungseigenschaft,199	ISYSHISTORY
IsEccAvailable-Eigenschaft	Version 16.0, Erweiterung,25
Servereigenschaft, Beschreibung,7	ISYSIDX
ISENCRYPTED-Funktion	Version 11.0.0, Verhaltensänderung,250
Version 12.0.0, neue Funktion,135	ISYSJAR
Isolationsstufe	Version 16.0, Erweiterung,25
Version 11.0.0, Erweiterung, UltraLite,	ISYSJAVACLASS
ReadUncommitted,271	Version 16.0, Erweiterung,25
Isolationsstufe 2	ISYSLOGINMAP
Version 12.0.0, Erweiterung,145	Version 11.0.0, neue Funktion,250
Isolationsstufe 3	ISYSLOGINPOLICY
Version 12.0.0, Erweiterung,145	Version 11.0.0, neue Funktion,250
IsPortableDevice, Eigenschaft	ISYSLOGINPOLICYOPTION
Version 12.0.0, neue Funktion,133	Version 11.0.0, neue Funktion,250
isql_allow_client_file_read	ISYSMIRROROPTION
Version 11.0.0, umbenannt,285	Version 12.0.0, neue Funktion,123
isql_allow_client_file_write	ISYSMIRRORSERVER
Version 11.0.0, umbenannt,285	Version 12.0.0, neue Funktion,123
isql_allow_read_client_file-Option	ISYSMIRRORSERVEROPTION
Version 11.0.0, umbenannt,285	Version 12.0.0, neue Funktion,123
isql_allow_write_client_file-Option	ISYSOBJECT
Version 11.0.0, umbenannt,285	Version 11.0.0, Verhaltensänderung,250
isql_command_timing, Option	Version 12.0.0, Erweiterung,145
Version 12.0.0, Erweiterung,191	Version 16.0, Erweiterung,25
isql_plan-Option	ISYSPROCPARM
Version 11.0.0, entfernt,285	Version 12.0.0, Erweiterung,144
ISYSCAPABILITYNAME	ISYSREMOTE USER
Version 11.0.0, Verhaltensänderung,250	Version 16.0, Erweiterung,25
ISYSCERTIFICATE	ISYSSEQUENCE
Version 16.0, neue Funktion,25	Version 12.0.0, neue Funktion,144
ISYSCOLSTAT	ISYSSEQUENCEPERM
Version 16.0, Erweiterung,25	Version 12.0.0, neue Funktion,144
ISYSDBFILE	ISYSSPATIALREFERENCESYSTEM
Version 11.0.0, Verhaltensänderung,250	Version 12.0.0, neue Funktion,121
ISYSDBSPACE	ISYSTAB
Version 11.0.0, Verhaltensänderung,250	Version 11.0.0, Verhaltensänderung,250
ISYSDBSPACEPERM	Version 16.0, Erweiterung,25
Version 11.0.0, Verhaltensänderung,250	ISYSTABCOL
ISYSEVENTTYPE	Version 12.0.0, Erweiterung,144
Version 11.0.0, Verhaltensänderung,250	Version 16.0, Erweiterung,25
ISYSEXTERNENVOBJECT	ISYSTEXTCONFIG
Version 16.0, Erweiterung,25	Version 11.0.0, neue Funktion,250
ISYSFILE	Version 12.0.0, Erweiterung,144

ISYTEXTIDX

Version 11.0.0, neue Funktion,250

Version 16.0, Erweiterung,25

ISYTEXTIDXTAB

Version 11.0.0, neue Funktion,250

ISYSUNITOFMEASURE

Version 12.0.0, neue Funktion,121

ISYSUSER

Version 16.0, Erweiterung,25

ISYSUSERTYPE

Version 12.0.0, Erweiterung,144

ISYSVIEW

Version 11.0.0, Verhaltensänderung,250

Version 16.0, Erweiterung,25

Itanium

Version 11.0.0, Verhaltensänderung,257

J

Java in der Datenbank

Version 12.0.1, Erweiterung,90

Java ME

Version 11.0.0, neue Funktion, UltraLite-
Unterstützung für Java ME,275

Version 16.0, entfernte Funktion, UltraLite-
Unterstützung für Java ME,66

Java SE

Version 11.0.0, neue Funktion, UltraLite-
Unterstützung für Java SE,275

Java, DownloadTableData-Schnittstelle

Version 11.0.0, Erweiterung,265

Java- und .NET-Skripten, Rückgabe von SQL-Code

Version 12.0.0, nicht mehr empfohlen,
MobiLink,170

java_class_path, Option

Version 12.0.1, neue Funktion,86

java_location

Version 16.0, nicht mehr empfohlen,38

java_main_userid

Version 16.0, nicht mehr empfohlen,38

java_main_userid, Eigenschaft

Version 12.0.0, nicht unterstützt,155

jConnect

Version 11.0.0, Erweiterung,226

JDBC

Fehlerbehandlung,322

Version 12.0.1, Erweiterung,89

Version 16.0, Verhaltensänderung,36

JDBC 3.0

Version 12.0.0, Erweiterung,142

Version 15, nicht mehr empfohlen,96

JDBC 3.0-Treiber

Version 16.0, nicht unterstützt,39

JDBC 4.0

Version 12.0.0, Erweiterung,142

Version 12.0.1, Erweiterung,77

JDBC-basierte Serverklassen

Version 12.0.0, nicht mehr empfohlen,155

Version 16.0, nicht unterstützt,39

JDBC-Treiber

Version 12.0.0, Erweiterung,142

Version 12.0.0, Verhaltensänderungen,149

JDBC-Treiber, SQL Anywhere

Version 12.0.0, neue Funktion,142

JSON-Dienste

Version 16.0, Verhaltensänderung,36

K

Katalog

Version 11.0.0, Verhaltensänderung,250

Kein Speicher mehr

Version 16.0, Verhaltensänderung,35

Kennwörter

Version 11.0.0, Erweiterung,224

Version 12.0.1, Erweiterung,75

Version 16.0, Erweiterung,13

Version 16.0, Verhaltensänderung,29

Kerberos

Version 12.0.1, Verhaltensänderung,96

Knoten kopieren

Version 12.0.0, neue Funktion,123

Kollation

Version 12.0.0, Verhaltensänderung,153

Kollationsanpassung, Unterstützung des SQL

Anywhere-Plug-Ins

Version 11.0.0, neue Funktion,278

Kompatibilität

Client/Server,295

Datenbanken und Datenbankserver,295

Info,295

UltraLite-Software umstellen,338

Komprimierte Spalten

Version 11.0.0, Erweiterung,246

Konfigurationsdateien

Version 11.0.0, Erweiterung,227

-
- Version 12.0.0, Erweiterung,117,129
 - Konflikterkennung
 - Version 16.0, entfernt, MobiLink,52
 - Konfliktverarbeitung
 - Version 16.0, Erweiterung, MobiLink-Plug-In,48
 - Konsolendienstprogramm (dbconsole)
 - Version 16.0, neue Funktionen,71
 - Konsolidierte Datenbanken
 - IBM DB2 LUW,330
 - Upgrade,326
 - Upgrade von Adaptive Server Enterprise, SAP
 - Sybase IQ, Oracle, MySQL oder Microsoft SQL Server,328
 - Upgrade von SQL Anywhere 10.0.0 und höher,327
 - Upgrade von SQL Anywhere vor Version 10.0.0,331
 - Kopieren, Verbindungszeichenfolge in Zwischenablage
 - Version 11.0.0, neue Funktion,277
 - Korrelationsnamen
 - Version 12.0.0, Erweiterung,140
 - L**
 - Laden
 - Version 11.0.0 neue Funktion, aus dem Transaktionslog,222
 - Version 11.0.0 neue Funktion, aus einem Wert,222
 - Version 11.0.0 neue Funktion, aus einer Spalte,222
 - Langtextpläne
 - Version 16.0, Erweiterung,27
 - LAST_VALUE-Funktion
 - Version 11.0.1, Erweiterung,201
 - LastCheckpointTime, Eigenschaft
 - Version 12.0.0, neue Funktion,132
 - LastCommitRedoPos-Eigenschaft
 - Version 12.0.1, neue Funktion,76
 - Version 16.0, neue Funktion,19
 - LastStatement-Eigenschaft
 - Version 16.0, Verhaltensänderung,31
 - LastSyncedRedoPos-Eigenschaft
 - Version 12.0.1, neue Funktion,76
 - Version 16.0, neue Funktion,19
 - Lastverteilung
 - Version 11.0.0, neue Funktion, MobiLink-Serverfarm,264
 - LastWrittenRedoPos-Eigenschaft
 - Version 12.0.1, neue Funktion,76
 - Version 16.0, neue Funktion,19
 - Latenzzeit
 - Relay Server, verbessern,54
 - LazyClose-Verbindungsparameter
 - Version 11.0.0, Verhaltensänderung,253
 - LCLOSE-Verbindungsparameter
 - Version 11.0.0, Verhaltensänderung,253
 - LDAP
 - Version 16.0, neue Funktion,5
 - Version 16.0, neue Funktion, MobiLink,40
 - Version 16.0, neue Funktion, MobiLink-Plug-In,48
 - LDAP-Benutzerauthentifizierung
 - Version 16.0, Erweiterungen,6
 - Version 16.0, neue Funktion,5
 - ldap_auto_failback_period-Option
 - Version 16.0, neue Funktion,6
 - ldap_failover_to_std-Option
 - Version 16.0, neue Funktion,6
 - ldap_primary_server-Option
 - Version 16.0, neue Funktion,6
 - ldap_refresh_dn-Option
 - Version 16.0, neue Funktion,6
 - ldap_secondary_server-Option
 - Version 16.0, neue Funktion,6
 - LEN-Funktion
 - Version 11.0.0, Erweiterung,246
 - LENGTH-Funktion
 - Version 11.0.0, Erweiterung,246
 - Lesesperren
 - Version 12.0.0, Erweiterung,145
 - Lightweight-Polling, Aktionsvariablen
 - Version 11.0.1, Erweiterung,211
 - Lightweight-Polling, Listener-Schlüsselwörter
 - Version 11.0.1, Erweiterung,211
 - LIMIT-Klausel
 - Version 12.0.0, neue Funktion,139
 - Version 12.0.1, Erweiterung,88
 - Linux
 - Version 11.0.0, Erweiterung,245
 - Version 11.0.0, neue Funktion,245
 - Version 11.0.0, Verhaltensänderung,257
 - Version 12.0.0, Erweiterung,145
 - Version 16.0, 32-Bit-Relay Server nicht mehr weiterentwickelt,56
 - Version 16.0, Erweiterung,26
 - Little-Endian, UTF-16-Kodierung
 - Version 11.0.0, neue Funktion,246
 - Lizenzen
-

- Version 12.0.0, Erweiterung,129
- Lizenzierung
 - Version 16.0, Verhaltensänderung,32
- LOAD STATISTICS-Anweisung
 - Version 12.0.0, Verhaltensänderung,149
- LOAD TABLE-Anweisung
 - Version 11.0.0, Erweiterung,235,238,269
 - Version 12.0.0, Erweiterung,137
 - Version 12.0.1, Erweiterung,75
 - Version 12.0.1, Verhaltensänderung,94
 - Version 16.0, Erweiterung,23
 - Version 16.0, Verhaltensänderung,37
- LOAD-Privileg
 - Version 16.0, neue Funktion,2
- LOCK FEATURE-Anweisung
 - Version 12.0.0, neue Funktion,137
- LOCK TABLE-Anweisung
 - Version 12.0.0, Erweiterung,138,145
- LockIndexID-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- LockRowID-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- Log Transfer Manager, Dienstprogramm
 - in Version 12.0.0 nicht unterstützt,117
- Log Transfer Manager, Dienstprogramm (dbltm)
 - Version 12.0.0, Verhaltensänderung,118
- Log-Ausführlichkeitsstufe für bestimmte Benutzer
 - Version 11.0.1, Erweiterung,211
- log4j.jar
 - Version 12.0.0, Verhaltensänderung,
 - MobiLink,167
- LogFormat, Protokolloption
 - Version 12.0.1, Verhaltensänderung,76
- Login-Richtlinien
 - Version 11.0.0, neue Funktion,220
 - Version 12.0.1, Erweiterung,75
 - Version 16.0, Erweiterung,6,13,68
- login_mode-Option (*Siehe* Version 12.0.1, neue Funktion, OnDemand)
 - Version 16.0, Erweiterung,6
- login_procedure-Option
 - Version 11.0.0, Erweiterung,228
- Logkonvertierungs-Dienstprogramm (dbtran)
 - Version 11.0.0, Erweiterung,227
- Lokale temporäre Tabellen
 - Version 16.0, Erweiterung,18
- Lokalisierte Versionen von SQL Anywhere
 - Version 16.0, Erweiterung,26

- LOTUS-Format, INPUT-Anweisung
 - Version 11.0.0, entfernt, Unterstützung,284
- LOTUS-Format, OUTPUT-Anweisung
 - Version 11.0.0, entfernt, Unterstützung,284
- LTM (Log Transfer Manager-Dienstprogramm)
 - in Version 12.0.0 nicht unterstützt,117
- LTMGeneration, Eigenschaft
 - Version 12.0.0, Verhaltensänderung,117
- LTMTrunc, Eigenschaft
 - Version 12.0.0, Verhaltensänderung,117

M

- Mac OS X
 - Datenbanken für SQL Anywhere 16 neu erstellen,297
 - Version 12.0.0, Erweiterung,145
 - Version 12.0.0, Verhaltensänderung,153
 - Version 16.0, Erweiterung,26
- MAPI
 - Version 16.0, Verhaltensänderung,34
- MAPI, Nachrichtentyp
 - Version 11.0.0, nicht unterstützt,269
- MAPI- und SMTP-Systemprozeduren
 - Version 16.0, Verhaltensänderung,34
- MapPages-Eigenschaft
 - Version 11.0.0, nicht unterstützt,259
- Maßeinheit erstellen
 - Version 12.0.0, neue Funktion,121
- Materialisierte Ansichten
 - Hinweise zum Upgrade,296
 - Version 11.0.0, Erweiterung,222
 - Version 12.0.0, Erweiterung,126
 - Version 12.0.0, SQL Remote, Initialisierung in extrahierten Datenbanken,172
 - Version 16.0, Erweiterung,22
- MatView-Verbindungsparameter
 - Version 12.0.1, neue Funktion,86
 - Version 12.0.1, Verhaltensänderung,95
- max_priority-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- max_priority-Option
 - Version 11.0.0, neue Funktion,228
- max_retries-Steuerungsparameter
 - Version 12.0.1, Erweiterung,104
- MaxEventType
 - Version 11.0.0, neue Funktion,231
- MaxMessage-Eigenschaft

Version 11.0.0, nicht mehr empfohlen,259
 MaxMultiProgrammingLevel, Eigenschaft
 Version 12.0.0, neue Funktion,133
 MaxRemoteCapability-Eigenschaft
 Version 11.0.0, neue Funktion,231
 MaxRequestVars-Protokolloption
 Version 16.0, neue Funktion,13
 MAXVARS-Protokolloption
 Version 16.0, neue Funktion,13
 MEDIAN-Funktion
 Version 12.0.0, neue Funktion,134
 Mehrere Netzwerk-Worker-Threads
 Version 16.0, neue Funktion, MobiLink,42
 Mehrere Versionen
 Adaptive Server Anywhere,298
 UltraLite,337
 Mehrfach vorhandene Textindizes
 Version 11.0.1, Verhaltensänderung,206
 Meldungen zum Bearbeitungsfortschritt
 Version 12.0.0, Erweiterung,134
 Memory (mem), erweiterte Option
 Version 12.0.0, nicht mehr weiterentwickelt,
 MobiLink,170
 MERGE-Anweisung
 Version 11.0.0, neue Funktion,220
 MESSAGE LOG-Klausel
 Version 16.0, neue Funktion,35
 MESSAGE-Anweisung
 Version 11.0.1, Erweiterung,202
 Message-Eigenschaft
 Version 11.0.0, nicht mehr empfohlen,259
 MessageCategoryLimit-Eigenschaft
 Version 11.0.0, neue Funktion,231
 MessageText-Eigenschaft
 Version 11.0.0, nicht mehr empfohlen,259
 MessageTime-Eigenschaft
 Version 11.0.0, nicht mehr empfohlen,259
 MessageWindowSize-Eigenschaft
 Version 11.0.0, nicht mehr empfohlen,259
 Microsoft Access
 Version 11.0.0, Erweiterung, Ferndatenzugriff,247
 Microsoft Distributed Transaction Coordinator
 Version 11.0.0, neue Funktion,264
 Version 12.0.0, neue Funktion,159
 Microsoft Excel
 Version 11.0.0, entfernt, Unterstützung der
 INPUT-Anweisung,284
 Version 11.0.0, entfernt, Unterstützung der
 OUTPUT-Anweisung,284
 Microsoft IIS
 Version 16.0, 32-Bit-Relay Server für Windows
 nicht mehr weiterentwickelt,56
 Microsoft IIS 8
 Version 16.0, neue Funktion, unterstützt für Relay
 Server,53
 Microsoft SQL Server 2005-Unterstützung
 Version 16.0, entfernt, MobiLink,52
 Microsoft SQL Server 2008, Unterstützung
 Version 11.0.1, Erweiterung,211
 Microsoft SQL Server 2012-Unterstützung
 Version 16.0, neue Funktion, MobiLink,42
 Microsoft Volume-Schattenkopiedienst (VSS)
 Version 11.0.0, neue Funktion,224
 Migration
 Monitor,342
 min_password_length-Protokolloption
 Version 16.0, Verhaltensänderung,29
 min_role_admins-Eigenschaft
 Version 16.0, neue Funktion,3
 min_role_admins-Option
 Version 16.0, neue Funktion,3
 Minimale Cachegröße
 Version 12.0.1, Verhaltensänderung,94
 MinMultiProgrammingLevel, Eigenschaft
 Version 12.0.0, neue Funktion,133
 Minor Release
 Spiegelung (Datenbankserver),316
 MIRROR-Serverklasse
 Version 16.0, neue Funktion,12
 MIRROR_FILE-Verbindungsparameter
 Version 11.0.1, Erweiterung, UltraLite,212
 MirrorMode-Eigenschaft
 Version 11.0.0, neue Funktion,232
 MirrorRole, Eigenschaft
 Version 12.0.0, neue Funktion,132
 MirrorServerState, Eigenschaft
 Version 12.0.0, neue Funktion,135
 MirrorServerState-Eigenschaft
 Version 12.0.1, Erweiterung,86
 MirrorState, Eigenschaft
 Version 12.0.0, neue Funktion,135
 ml_add_certificates_file-Systemprozedur
 Version 16.0, neue Funktion, MobiLink,40
 ml_add_column-Systemprozedur

- Version 12.0.0, Verhaltensänderung, MobiLink,167
- Version 16.0, nicht mehr empfohlen, MobiLink,53
- ml_add_idap_server-Systemprozedur
 - Version 16.0, neue Funktion, MobiLink,40
- ml_add_missing_dnld_scripts-Systemprozedur
 - Version 12.0.0, neue Funktion, MobiLink,161
- ml_add_user_auth_policy-Systemprozedur
 - Version 16.0, neue Funktion, MobiLink,41
- ml_column-Systemtabelle
 - Version 16.0, nicht mehr empfohlen, MobiLink,53
- ML_GET_SERVER_NOTIFICATION-Funktion
 - Version 11.0.1, Erweiterung, UltraLite,212
- ml_ignore
 - Version 11.0.1, neue Funktion, MobiLink (mlsrv11),211
- ml_model_check_all_schema-Systemprozedur
 - Version 16.0, neue Funktion, MobiLink,41
- ml_model_check_version_schema-Systemprozedur
 - Version 16.0, neue Funktion, MobiLink,41
- ml_model_drop-Systemprozedur
 - Version 16.0, neue Funktion, MobiLink,41
- ml_remote_id, Option
 - Version 11.0.1, Erweiterung, UltraLite,212
- mlarbiter, Dienstprogramm
 - Version 12.0.0, neue Funktion, MobiLink,162
- mlarbiter, Dienstprogramm für Serverfarmen
 - Version 12.0.0, neue Funktion, MobiLink,157
- mlarbstop, Dienstprogramm
 - Version 16.0, neue Funktion,44
- mlfiletransfer
 - Version 12.0.0, Erweiterung, MobiLink,165
- MLFileTransfer-Methode
 - Version 12.0.0, Verhaltensänderung, MobiLink,168
- MLFileUpload-Methode
 - Version 12.0.0, neue Funktion, MobiLink,163
- mlreplay, Dienstprogramm
 - Version 12.0.0, neue Funktion,157
 - Version 16.0, Callbacks entfernt,52
- mlreplay-Dienstprogramm, Fehlermeldungen
 - Version 12.0.1, Erweiterung, MobiLink,84
- mlreplay-Verfügbarkeits-Timeout
 - Version 16.0, Erweiterung, MobiLink (mlreplay),44
- mlsrv12
 - Version 12.0.0, neue Funktion, MobiLink,160
 - Version 12.0.0, Verhaltensänderung,118
- mluser, Dienstprogramm
 - Version 16.0, neue Funktion,43
- MLUser-Synchronisationsprofil
 - Version 12.0.0, nicht mehr empfohlen, MobiLink,172
- MobiLink
 - Upgrade,325
- MobiLink 12-Plug-In für Sybase Central
 - in Version 12.0.0 neu gestaltet,157
- MobiLink Listener, C API für Palm-Geräte
 - Version 12.0.0, entfernt,169
- MobiLink Listener, Dienstprogramm für Palm-Geräte
 - Version 12.0.0, entfernt,169
- MobiLink, Logdateien
 - Version 11.0.0, Erweiterung, UltraLite,272
- MobiLink, neue Funktionen
 - Version 11.0.0,263
 - Version 11.0.1,210
 - Version 12.0.0,156
 - Version 12.0.1,96
 - Version 16.0,40
- MobiLink, nicht mehr empfohlene Funktionen
 - Version 12.0.1,103
- MobiLink, nicht mehr weiterentwickelte Funktionen
 - Version 12.0.1,103
- MobiLink, Relay Server
 - Version 11.0.0, neue Funktion,264
- MobiLink, Serverfarm
 - Version 11.0.0, neue Funktion,264
- MobiLink, Systemdatenbank
 - Version 11.0.0, neue Funktion,263
- MobiLink, Systemobjekte
 - Version 11.0.0, Erweiterung,263
- MobiLink, Verhaltensänderungen
 - Version 11.0.0,267
 - Version 11.0.1,212
 - Version 12.0.0,167
 - Version 12.0.1,101
 - Version 16.0,49
- MobiLink-Agent
 - Version 12.0.1, neue Funktion, MobiLink,98
- MobiLink-Client
 - Version 12.0.1, Verhaltensänderungen,102
- MobiLink-Clients
 - Version 12.0.1, neue Funktionen,100
- MobiLink-Konnektivität
 - Version 16.0, Verhaltensänderungen,49
- MobiLink-Monitor

- Version 11.0.0, Verhaltensänderungen,286
- Version 16.0, entfernt, MobiLink,40,44
- MobiLink-Plug-In
 - Version 12.0.1, Erweiterung, MobiLink,99
 - Version 12.0.1, Verhaltensänderungen,103
- MobiLink-Profiler
 - Version 16.0, neue Funktion,44
 - Version 16.0, neue Funktion, MobiLink,40
- MobiLink-Server
 - 32-Bit-Unterstützung, Version 16.0, Verhaltensänderung,50
 - Upgrade,334
- MobiLink-Server, 32-Bit-Unterstützung
 - Version 12.0.1, Verhaltensänderung, MobiLink,101
- MobiLink-Serverkompatibilität
 - Version 16.0, neue Funktion, MobiLink,40
- MobiLink-Systemobjekte
 - Upgrade,326
- MobiLink-Systemprozeduren
 - Version 11.0.0, Erweiterung,263
- Monitor
 - Ressourcen migrieren,342
 - Upgrade,342
 - Version 11.0.1, neue Funktion,217
 - Version 12.0.0, neue Funktion,195
 - Version 12.0.0, Verhaltensänderung,196
 - Version 12.0.1, neue Funktion,114
 - Version 12.0.1, Verhaltensänderung,116
 - Version 16.0, neue Funktion,70
 - Version 16.0, Verhaltensänderung,71
- Monitor, Verhaltensänderungen
 - Version 12.0.0,196
 - Version 12.0.1,116
- MSACCESSODBC-Serverklasse
 - Version 11.0.0, neue Funktion,247
- Multiprogramming-Stufe
 - Version 12.0.0, Erweiterung,125
 - Version 12.0.0, Verhaltensänderungen,148
 - Version 12.0.1, Verhaltensänderungen,94
- MySQL Server 5.5.X-Unterstützung
 - Version 16.0, neue Funktion, MobiLink,42
- MySQL, konsolidierte Datenbank
 - Version 11.0.0, neue Funktion,263
- MySQL-Unterstützung im MobiLink-Modellmodus
 - Version 11.0.1, Erweiterung,211
- MYSQLODBC-Serverklasse
 - Version 11.0.0, neue Funktion,247

N

- Nachbarschaftssuchen
 - Version 11.0.1, Verhaltensänderung,206
 - Version 16.0, Erweiterung,27
- Nachrichten-Steuerungsparameter
 - Version 12.0.1, Erweiterung,104
- Name, Eigenschaft
 - Version 12.0.0, Verhaltensänderung,150
- NCHAR, Datentyp
 - Version 12.0.0, Erweiterung,140
- network_conn_name, Variable
 - Version 12.0.1, neue Funktion, MobiLink,98
- NetworkData-Klasse
 - Version 12.0.1, neue Funktion, MobiLink,83
- Netzwerk, Registerkarte
 - Version 11.0.0, neue Funktion,277
- Netzwerkserver
 - Version 12.0.0, Erweiterung,125
 - Version 12.0.0, Verhaltensänderung,148
- Neu aufbauen
 - Datenbanken der Version 9 oder früher mit dbunload,309
 - Datenbanken der Version 9 und früher,307
 - Datenbanken für Version 16,298,310
- Neu erstellen, Datenbank der Version 9 oder früher
 - Info,303
- Neu startbare Downloads unter Android
 - Version 16.0, Erweiterung, UltraLiteJ,64
- Neu startbare HTTP-Verbindung
 - Version 12.0.1, neue Funktion,82
- Neuaufbau
 - Sicherheitsvorkehrungen für Upgrade,297
- Neuaufbau, Datenbanken
 - Version 11.0.0, Erweiterung,223
- Neuaufbau, Dienstprogramm
 - Version 12.0.0, nicht unterstützt,155
- Neue Funktionen
 - Version 11.0.0,219
 - Version 11.0.1,199
 - Version 12.0.0,117
 - Version 12.0.1,73
 - Version 16.0,1
 - Version 9.0.2 und früher,vii
- Neue Funktionen in Version 11.0.0
 - Info,219
- Neue Funktionen in Version 11.0.1
 - Info,199

- Neue Funktionen in Version 12.0.0
 - Info,117
 - Neue Funktionen in Version 12.0.1
 - Info,73
 - Neue Funktionen in Version 16.0
 - Info,1
 - Neuerstellung
 - Einschränkungen,304
 - Fehlersuche,305
 - Neuerstellung von Datenbanken
 - fehlgeschlagene Neuerstellung,305
 - new_remote_id, Systemparameter
 - Version 12.0.1, neue Funktion, MobiLink,97
 - new_username, Systemparameter
 - Version 12.0.1, neue Funktion, MobiLink,97
 - newdemo
 - Version 12.0.0, neue Funktion,146
 - NewPassword-Verbindungsparameter
 - Version 11.0.0, neue Funktion,224
 - NEWPWD-Verbindungsparameter
 - Version 11.0.0, neue Funktion,224
 - NEXT_CONNECTION-Funktion
 - Version 16.0, Verhaltensänderung,33
 - NEXT_DATABASE-Funktion
 - Version 16.0, Verhaltensänderung,33
 - NEXT_HTTP_RESPONSE_HEADER-Funktion
 - Version 12.0.0, neue Funktion,135
 - Nicht mehr empfohlene Funktionen
 - Version 11.0.0,219
 - Version 11.0.1,199
 - Version 12.0.0,117
 - Version 12.0.1,73
 - Version 16.0,1
 - Version 9.0.2 und früher,vii
 - Nicht synchronisierte Spalten, Aktualisierungen
 - Version 12.0.1, Verhaltensänderung, MobiLink,83
 - Nicht-blockierende Downloadbestätigungsskripten
 - Version 11.0.0, Erweiterung,266
 - Version 12.0.0, Erweiterung,161
 - no_reload_status
 - Version 12.0.0, neue Funktion,141
 - NODE-Verbindungsparameter
 - Version 12.0.0, neue Funktion,123
 - NodeType-Verbindungsparameter
 - Version 12.0.0, neue Funktion,123
 - Version 12.0.1, Erweiterung,73
 - Version 12.0.1, Verhaltensänderung, OnDemand,74
 - non_keywords-Option
 - Version 11.0.0, Verhaltensänderung,255
 - NOTIFY TRACE EVENT-Anweisung
 - Version 16.0, neue Funktion,23
 - NumLocalTempTables-Eigenschaft
 - Version 16.0, neue Funktion,19
 - NVARCHAR, Datentyp
 - Version 12.0.0, Erweiterung,140
- ## O
- ObjectType, Eigenschaft
 - Version 12.0.0, neue Funktion,133
 - OData Producer
 - Version 16.0, neue Funktion,10
 - OData Server
 - Version 16.0, neue Funktion,10
 - ODBC
 - Version 16.0, Verhaltensänderung,36
 - ODBC, Serverklassen
 - Version 11.0.0, Erweiterung,247
 - ODBC-Datenquellen
 - Version 12.0.0, Erweiterung,128
 - ODBC-Datenquellen-Administrator
 - Version 12.0.1, Erweiterung, OnDemand,73
 - ODBC-Treiber
 - Version 12.0.0, Erweiterung,141
 - ODBC-Treibermanager
 - Version 11.0.0, Erweiterung,242
 - ODBC-Verbindungsparameter
 - Version 12.0.0, Erweiterung,128
 - OEM.ini
 - Version 11.0.0, Erweiterung,280
 - Version 12.0.0, Erweiterung,191
 - Version 12.0.1, Erweiterung,75
 - Version 12.0.1, neue Funktion,111
 - Version 16.0, Erweiterung,67,70
 - OFFSET-Klausel
 - Version 12.0.1, Erweiterung,88
 - OLE DB
 - Version 11.0.1, Erweiterung,203
 - Version 12.0.1, Erweiterung,89
 - Version 16.0, Erweiterung,25
 - ON-Klausel
 - Version 16.0, nicht unterstützt,37
 - OnDemand Edition, Unterstützung
 - Version 12.0.1, neue Funktion,73
 - Open Client

- Version 11.0.0, Erweiterung,226
- OpenString-Algorithmus
 - Version 11.0.0, neue Funktion,223
- OPENSTRING-Klausel
 - Version 11.0.0, neue Funktion,223
 - Version 12.0.1, Erweiterung,77
 - Version 16.0, Erweiterung,24
- OpenTableEx-Funktion [UL C++]
 - Version 11.0.0, Erweiterung, UltraLite,274
- OPENXML-Operator
 - Version 12.0.0, Erweiterung,134
 - Version 12.0.0, Verhaltensänderungen,152
- Operatorvorrang, Volltextsuche
 - Version 11.0.1, Verhaltensänderung,206
- Optimierer
 - Version 11.0.0, Erweiterung,221
 - Version 11.0.1, Erweiterung,204
 - Version 12.0.1, Erweiterung,91
 - Version 16.0, Erweiterung,9
- optimistic_wait_for_commit-Option
 - Version 11.0.0, nicht unterstützt,260
- OPTION-Klausel
 - Version 11.0.0, Erweiterung,239
 - Version 11.0.0, Verhaltensänderung,257
- Optionen, Fenster
 - Version 16.0, Erweiterung,70
- Optionen-Überwachungsliste
 - Version 11.0.0, neue Funktion,234
- OptionWatchAction-Eigenschaft
 - Version 11.0.0, neue Funktion,231
- OptionWatchList-Eigenschaft
 - Version 11.0.0, neue Funktion,231
- OR REPLACE-Klausel
 - Version 11.0.1, Erweiterung,202
 - Version 12.0.0, neue Funktion,139
- Oracle
 - Version 11.0.0, Erweiterung,264
 - Version 12.0.0, Erweiterung,159
 - Version 16.0, neue Funktion, MobiLink,42
- Oracle, Treiber
 - Version 11.0.0, Erweiterung,264
 - Version 12.0.0, Erweiterung,159
- Oracle-Varray
 - Version 11.0.1, neue Funktion,211
- OSGi-Deployment-Bundle
 - Version 12.0.1, Erweiterung,77
- OSUser-Eigenschaft
 - Version 11.0.0, neue Funktion,230

- Outbound Enabler
 - Version 12.0.0, Erweiterung, Relay Server,166
- OUTER APPLY-Klausel
 - Version 11.0.0, neue Funktion,239
- Outer Joins
 - Fehlerbehandlung, Upgrades auf Version 16,306
- Outer-Joins
 - Version 12.0.0, Erweiterung,126
- OUTPUT-Anweisung
 - Version 11.0.0, Erweiterung,284
 - Version 11.0.0, Verhaltensänderung ,259,284
 - Version 11.0.1, Verhaltensänderung,209
 - Version 12.0.0, Erweiterung,194

P

- PacketsReceived, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- PacketsReceivedUncomp, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- PacketsSent, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- PacketsSentUncomp, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- Paketgröße
 - Version 11.0.0, Standardwertänderung,247
- Palm OS
 - Version 12, nicht mehr empfohlen, MobiLink,275
- Parallele Sicherungen
 - Version 11.0.1, Erweiterung,205
- ParentConnection, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- Partieller Index-Scan
 - Version 12.0.1, Erweiterung,91
- PartnerState-Eigenschaft
 - Version 12.0.1, Erweiterung,86
 - Version 16.0, Verhaltensänderung,12
- pause_after_failure-Steuerungsparameter
 - Version 12.0.1, Erweiterung,104
- PBUF-Verbindungsparameter
 - Version 11.0.0, Erweiterung,224
 - Version 12.0.0, Verhaltensänderung,152
- percent_as_comment-Option
 - Version 11.0.0, Verhaltensänderung,260
- Performance
 - Upgrade von Datenbankdateien,303
 - Version 12.0.0, Erweiterung,145
 - Version 12.0.1, Erweiterungen,90

- Version 16.0, Erweiterung,26
- Performancedaten
 - Version 16.0, Erweiterung,67
- Performancestatistik-Dienstprogramm (dbstats)
 - Version 12.0.0, neue Funktion,145
- Performancewarnungen
 - Version 12.0.0, Erweiterung,197
- Personal Server
 - Version 12.0.0, Verhaltensänderung,149
- PHP
 - Version 12.0.0, Erweiterung,143
 - Version 16.0, Erweiterung,25
 - Version 16.0, nicht unterstützte Funktionen,38
- PHP-Modul
 - Version 11.0.0, Verhaltensänderung-Funktionsnamensänderungen,252
- PID-Dateien
 - Version 12.0.0, neue Funktion,145
- Plan-Caching
 - Version 11.0.0, Erweiterung,247
 - Version 16.0, Verhaltensänderung,30
- Plananzeige
 - Version 11.0.0, Erweiterung,282
 - Version 11.0.0, nicht mehr empfohlen, Funktionen,282
- Pläne
 - Version 12.0.1, Erweiterung,77
- Platten-Lesezugriffe, Arbeitstabelle, Statistik
 - Version 11.0.0, neue Funktion,232
- Plattformen
 - Version 11.0.0, Erweiterung, UltraLite,273
 - Version 12.0.0, Erweiterung, UltraLite,173
 - Version 12.0.1, Erweiterung, UltraLite,106
 - Version 16.0, Erweiterung, UltraLite,60
- Plug-Ins
 - unterstützte Versionen,289
- Pocket PC 2003
 - Version 12.0.1, neue Funktion,83
 - Version 16.0, Erweiterung, UltraLite,60
- Port, Protokolloption
 - Version 12.0.0, Verhaltensänderung,150
- post_login_procedure-Option
 - Version 11.0.0, Verhaltensänderung,255
- precision-Option
 - Version 11.0.0, Verhaltensänderung,255
- Prefetch
 - Version 11.0.0, Erweiterung,224
- PrefetchBuffer (PBUF)-Verbindungsparameter
 - Version 12.0.0, Verhaltensänderung,152
- PrefetchBuffer-Verbindungsparameter
 - Version 11.0.0, Erweiterung,224
- PREFILTER-Klausel
 - Version 12.0.0, neue Funktion,126
- PREPARE-Anweisung
 - Version 12.0.0, Verhaltensänderung,151
- Prepares-Eigenschaft
 - Version 11.0.1, neue Funktion,200
- PrepStmt, Eigenschaft
 - Version 12.0.1, neue Funktion,92
- PreserveSource-Eigenschaft
 - Version 11.0.0, nicht unterstützt,259
- priority-Eigenschaft
 - Version 11.0.0, neue Funktion,230
- priority-Option
 - Version 11.0.0, neue Funktion,228
- Privilegien
 - Version 16.0, neue Funktion,1
- ProcessID, Eigenschaft
 - Version 12.0.1, neue Funktion,76
- ProcessorAffinity-Eigenschaft
 - Version 16.0, neue Funktion,19
- PROFILE-Berechtigung
 - Version 16.0, nicht mehr empfohlen,37
- Progress, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- progress_messages, Option
 - Version 12.0.0, neue Funktion,130
- progress_messages-Eigenschaft
 - Version 12.0.0, neue Funktion,132
- promotion_time-Option
 - Version 16.0, neue Funktion,12
- PROPERTY_NAME-Funktion
 - Version 11.0.0, Erweiterung,232
- Protokollierung
 - Version 16.0, neue Funktion, MobiLink,41
- Protokollierung von Datenbankänderungen starten
 - Version 12.0.0, neue Funktion,188
- Proxy-Tabellen
 - Version 12.0.0, Erweiterung,146
- proxy_host, Option
 - Version 12.0.1, neue Funktion, Relay Server,101
- proxy_port, Option
 - Version 12.0.1, neue Funktion, Relay Server,101
- Prüfsummen
 - Version 11.0.0, Erweiterung,245
 - Version 11.0.0, Verhaltensänderung,254

- Version 12.0.0, Erweiterung,127
- Version 12.0.0, Verhaltensänderung,148
- publication-Synchronisationsprofil
 - Version 12.0.0, nicht mehr empfohlen,
 - MobiLink,171

Python

- Version 11.0.0, neue Funktion, Python-Datenbank-API,240
- Version 12.0.0, neue Funktion,143

Q

QAnywhere

- nicht unterstützt für Version 16.0,53

query_mem_timeout-Eigenschaft

- Version 11.0.0, neue Funktion,230

query_mem_timeout-Option

- Version 11.0.0, neue Funktion,228

query_plan_on_open-Option

- Version 11.0.0, nicht unterstützt,260

QueryBypassedCosted-Eigenschaft

- Version 11.0.1, neue Funktion,
Datenbankeigenschaft,200
- Version 11.0.1, neue Funktion,
Verbindungseigenschaft,199

QueryBypassedHeuristic-Eigenschaft

- Version 11.0.1, neue Funktion,
Datenbankeigenschaft,200
- Version 11.0.1, neue Funktion,
Verbindungseigenschaft,199

QueryBypassedOptimized-Eigenschaft

- Version 11.0.1, neue Funktion,
Datenbankeigenschaft,200
- Version 11.0.1, neue Funktion,
Verbindungseigenschaft,199

QueryDescribedBypass-Eigenschaft

- Version 11.0.1, neue Funktion,
Datenbankeigenschaft,200
- Version 11.0.1, neue Funktion,
Verbindungseigenschaft,199

QueryDescribedOptimizer-Eigenschaft

- Version 11.0.1, neue Funktion,
Datenbankeigenschaft,200
- Version 11.0.1, neue Funktion,
Verbindungseigenschaft,199

QueryHeapPages, Eigenschaft

- Version 12.0.1, neue Funktion,92

QueryMemActiveCurr-Eigenschaft

- Version 11.0.0, neue Funktion,230

- Version 12.0.1, neue Funktion,92

QueryMemActiveEst-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemActiveMax-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemExtraAvail-Eigenschaft

- Version 11.0.0, neue Funktion,230

QueryMemGrantBase-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemGrantBaseMI-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemGrantExtra-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemGrantFailed-Eigenschaft

- Version 11.0.0, neue Funktion,230
- Version 12.0.1, neue Funktion,92

QueryMemGrantGranted-Eigenschaft

- Version 11.0.0, neue Funktion,230
- Version 12.0.1, neue Funktion,92

QueryMemGrantRequested-Eigenschaft

- Version 11.0.0, neue Funktion,230
- Version 12.0.1, neue Funktion,92

QueryMemGrantWaited, Eigenschaft

- Version 12.0.1, neue Funktion,92

QueryMemGrantWaiting-Eigenschaft

- Version 11.0.0, neue Funktion,230

QueryMemPages-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemPercentOfCache-Eigenschaft

- Version 11.0.0, neue Funktion,231

QueryMemWaited-Eigenschaft

- Version 11.0.0, neue Funktion,230

QueryOpened-Eigenschaft

- Version 11.0.1, neue Funktion,
Datenbankeigenschaft,200
- Version 11.0.1, neue Funktion,
Verbindungseigenschaft,199

QueryReused-Eigenschaft

- Version 11.0.0, neue Funktion,232

QueryRowsBufferFetch, Eigenschaft

- Version 12.0.0, nicht unterstützt,155

QueryRowsFetched, Eigenschaft

- Version 12.0.0, neue Funktion,132

quoted_identifier-Option

- Version 11.0.0, Verhaltensänderung,255

R

- RAISERROR-Anweisung
 - Version 12.0.1, Erweiterung,88
- RAISERROR-Klausel
 - Version 12.0.1, Erweiterung,88
- Räumliche Daten
 - Version 12.0.0, neue Funktion,119
 - Version 12.0.1, Laden von Erweiterungen,85
- Räumliche Daten, Synchronisation
 - Version 12.0.0, neue Funktion, MobiLink,159
- Räumliche Datentypen
 - Version 12.0.0, neue Funktion, MobiLink,157
- Räumliche Vorschau Tabulator
 - Version 12.0.0, neue Funktion,120
- RAW-Verschlüsselung
 - Version 16.0, neue Funktion,14
- READ-Anweisung
 - Version 12.0.0, Erweiterung,113
- READ_CLIENT_FILE-Funktion
 - Version 11.0.0, neue Funktion,232
- readcert-Dienstprogramm
 - Version 11.0.0, entfernt,288
- READCLIENTFILE-Berechtigung
 - Version 16.0, nicht mehr empfohlen,37
- READFILE-Berechtigung
 - Version 16.0, nicht mehr empfohlen,37
- ReadHint-Eigenschaft
 - Version 11.0.0, neue Funktion,230
 - Version 11.0.0, umbenannt,258
- ReadHintScatter-Eigenschaft
 - Version 11.0.0, neue Funktion,230
 - Version 11.0.0, umbenannt,258
- ReadHintScatterLimit-Eigenschaft
 - Version 11.0.0, neue Funktion,231
 - Version 11.0.0, umbenannt,258
- ReadPK, Sperren
 - Version 12.0.0, neue Funktion,145
- ReceiveBufferSize, Protokolloption
 - Version 12.0.1, Erweiterung,75
- Redirector
 - Version 11.0.0, nicht mehr empfohlen,266
 - Version 12.0.0, entfernt,171
- REFRESH MATERIALIZED VIEW-Anweisung
 - Version 11.0.0, Erweiterung,222
 - Version 11.0.0, Verhaltensänderung,252
- REFRESH TEXT INDEX-Anweisung
 - Version 11.0.0, neue Funktion,237
- REGBIN-Protokolloption
 - Version 11.0.0, nicht unterstützt,260
- REGEXP-Suchbedingung
 - Version 11.0.0, neue Funktion,220
 - Version 11.0.1, Verhaltensänderung,207
- REGEXP_SUBSTR-Funktion
 - Version 11.0.0, neue Funktion,220,233
 - Version 11.0.1, Verhaltensänderungen,207
- RegisterBindery-Protokolloption
 - Version 11.0.0, nicht unterstützt,260
- Reguläre Ausdrücke
 - Version 11.0.0, neue Funktion,220
 - Version 11.0.1, Verhaltensänderung,207
- Reine Download-Tabellen
 - Version 12.0.1 neue Funktion, UltraLite,105
- Relative Pfade
 - Version 16.0, Verhaltensänderung,29
- Relay Server
 - Version 12.0.1, neue Funktionen,101
 - Version 16.0, 32-Bit-Version nicht mehr weiterentwickelt (Linux),56
 - Version 16.0, 32-Bit-Version nicht mehr weiterentwickelt (Windows),56
- Relay Server, neue Funktionen
 - Identitäts-Weiterleitung,84
 - Version 16.0,53
- Relay Server, Verhaltensänderungen
 - Proxy-Unterstützung mit Durchreichen,54
 - Schnellsetup für Apache unter Linux,54
 - Schnellsetup für IIS 7, 7.5 und 8.0,53
 - Speicherverwaltung, verbessert,55
 - überlapptes Cookie erneuern,54
 - up_pad_size,54
 - Version 16.0,55
 - Verwaltung des virtuellen Speichers, verbessert,54
- Relay Server-Plug-In
 - Version 12.0.0, neue Funktion, MobiLink,158
- Relay Server-Plug-In, Verhaltensänderungen
 - Erweitert, Registerkarte, Eigenschaften der Backend-Serverfarm,56
 - Netzwerk, Registerkarte ,55
 - Netzwerk, Registerkarte, Eigenschaften der Relay Server-Farm,55
- Relay Server-Statusseite, verbessert
 - Version 16.0, Verhaltensänderung,55
- reload.sql
 - Version 11.0.0, Erweiterung,248
- REMOTE DBA-Berechtigung

Version 16.0, nicht mehr empfohlen,37
 remote_id-Systemparameter
 Version 12.0.0, neue Funktion, MobiLink,161
 RemoteCapability-Eigenschaft
 Version 11.0.0, neue Funktion,231
 REORGANIZE TABLE-Anweisung
 Version 11.0.0, Verhaltensänderungen,252
 replicate_all, Option
 in Version 12.0.0 nicht unterstützt,118
 Replication Server
 in Version 12.0.0 nicht unterstützt,117
 ReqCountActive, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqCountBlockContention, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqCountBlockIO, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqCountBlockLock, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqCountUnscheduled, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqTimeActive, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqTimeBlockContention, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqTimeBlockIO, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqTimeBlockLock, Eigenschaft
 Version 12.0.1, neue Funktion,92
 ReqTimeUnscheduled, Eigenschaft
 Version 12.0.1, neue Funktion,92
 reqtool-Dienstprogramm
 Version 11.0.0, entfernt,288
 RequestsReceived, Eigenschaft
 Version 12.0.1, neue Funktion,92
 reserved_keywords, Option
 Version 12.0.0, neue Funktion,131
 reserved_keywords-Option
 Version 12.0.1, Verhaltensänderung,94
 Reservierte Wörter
 Fehlerbehandlung, Upgrades auf Version 16,306
 Version 12.0.0, Erweiterung,134
 Version 12.0.0, Verhaltensänderungen,153
 Version 16.0, Erweiterung,28
 Reservierte Wörter, Option
 Version 12.0.0, neue Funktion,131
 reset.sql
 Tabellen mit Autoinkrement-Spalten neu laden,300
 RESOURCE-Berechtigung
 Version 16.0, nicht mehr empfohlen,37
 RESTART-Klausel
 Version 16.0, neue Funktion,13
 RESTORE DATABASE-Anweisung
 Version 12.0.0, Erweiterung,128
 Version 16.0, Erweiterung,24
 ResultSet.afterLast-Methode [UltraLiteJ]
 Version 12.0.1, neue Funktion,79
 ResultSet.beforeFirst-Methode [UltraLiteJ]
 Version 12.0.1, neue Funktion,79
 ResultSet.first-Methode [UltraLiteJ]
 Version 12.0.1, neue Funktion,79
 ResultSet.getBlobInputStream-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getBoolean-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getBytes-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getClobReader-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getDate-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getDecimalNumber-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getDouble-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getFloat-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getLong-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getRowCount-Methode [UltraLiteJ]
 Version 12.0.1, neue Funktion,79
 ResultSet.getSize-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.getString-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80,82
 ResultSet.getUUIDValue-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.isNull-Methode [UltraLiteJ]
 Version 12.0.1, Erweiterung,80
 ResultSet.last-Methode [UltraLiteJ]
 Version 12.0.1, neue Funktion,79
 ResultSet.relative-Methode [UltraLiteJ]
 Version 12.0.1, neue Funktion,79
 ResultSetMetaData.getAliasName-Methode
 [UltraLiteJ]
 Version 12.0.1, Erweiterung,81

ResultSetMetaData.getCorrelationName-Methode
[UltraLiteJ]
Version 12.0.1, Erweiterung,81

ResultSetMetaData.getDomainName-Methode
[UltraLiteJ]
Version 12.0.1, Erweiterung,81

ResultSetMetaData.getQualifiedName-Methode
[UltraLiteJ]
Version 12.0.1, Erweiterung,81

ResultSetMetaData.getTableColumnName-Methode
[UltraLiteJ]
Version 12.0.1, Erweiterung,81

ResultSetMetaData.getTableName-Methode
[UltraLiteJ]
Version 12.0.1, Erweiterung,81

ResultSetMetaData.getWrittenName-Methode
[UltraLiteJ]
Version 12.0.1, Erweiterung,81

RESUME-Anweisung
Version 12.0.0, Erweiterung,143

REVERSE-Funktion
Version 16.0, neue Funktion,9

REVOKE PUBLISH-Anweisung
Version 16.0, Verhaltensänderung,4

REVOKE ROLE-Anweisung
Version 16.0, neue Funktion,3

REVOKE-Anweisung
Version 12.0.0, Erweiterung,124
Version 16.0, Erweiterung,3

REWRITE-Funktion
Version 16.0, Verhaltensänderung,31

ri_trigger_time-Option
Version 11.0.0, nicht unterstützt,260

RIM
Version 11.0.0, neue Funktion, UltraLite-
Unterstützung für BlackBerry,275

Rlbk, Eigenschaft
Version 12.0.1, neue Funktion,92

ROLLBACK-Anweisung
Version 12.0.0, Erweiterung,191

Rollback-Log
Version 11.0.0, Verhaltensänderung,257

Rollen
Version 16.0, neue Funktion,1
Version 16.0, neue Funktion, MobiLink-Client,45

Root-Login-Richtlinie
Version 16.0, Erweiterung,6

root_auto_lock_time-Option

Version 16.0, neue Funktion,2

ROW LOG-Klausel
Version 16.0, neue Funktion,35

ROW-Konstruktor
Version 16.0, neue Funktion,9

RSA TLS und HTTPS für Pocket PC 2003
Version 16.0, Erweiterung, UltraLite,60

RSOE
Version 12.0.0, Erweiterung, Relay Server,166

Ruby
Version 11.0.1, neue Funktion,203

Rückgabecodes
Version 16.0, Verhaltensänderung,34

S

sa_audit_string-Systemprozedur
Version 16.0, Verhaltensänderung,28

sa_certificate_info-Systemprozedur
Version 16.0, neue Funktion,20

sa_char_terms-Systemprozedur
Version 11.0.0, neue Funktion,233

sa_clean_database-Systemprozedur
Version 12.0.1, Verhaltensänderung,90
Version 16.0, Verhaltensänderung,28

sa_column_stats-Systemprozedur
Version 16.0, Verhaltensänderung,28

sa_config.csh
Version 12.0.1, Erweiterung, Unix,87

sa_config.sh
Version 12.0.1, Erweiterung, Unix,87

sa_conn_activity-Systemprozedur
Version 16.0, Verhaltensänderung,28,33

sa_conn_compression_info-Systemprozedur
Version 16.0, Verhaltensänderung,28,33

sa_conn_info-Systemprozedur
Version 12.0.0, Erweiterung,128
Version 16.0, Verhaltensänderung,28,33

sa_conn_list-Systemprozedur
Version 16.0, Verhaltensänderung,28,33

sa_conn_options-Systemprozedur
Version 16.0, Verhaltensänderung,28,33

sa_conn_properties-Systemprozedur
Version 11.0.0, Erweiterung,235
Version 16.0, Verhaltensänderung,28,33

sa_copy_cursor_to_temp_table-Systemprozedur
Version 12.0.0, neue Funktion,133

sa_cpu_topology-Systemprozedur

Version 16.0, neue Funktion,20
sa_db_info-Systemprozedur
Version 16.0, Verhaltensänderung,28,33
sa_db_list-Systemprozedur
Version 16.0, Verhaltensänderung,28,33
sa_db_option-Systemprozedur
Version 16.0, neue Funktion,17,20
sa_db_properties-Systemprozedur
Version 16.0, Verhaltensänderung,28,33
sa_describe_cursor-Systemprozedur
Version 12.0.0, neue Funktion,133
sa_describe_shapefile-Systemprozedur
Version 12.0.0, neue Funktion,120
sa_disable_auditing_type-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_disk_free_space-Systemprozedur
Version 11.0.0, Erweiterung,234
Version 16.0, Verhaltensänderung,28
sa_external_library_unload-Systemprozedur
Version 11.0.0, neue Funktion,234
Version 16.0, Verhaltensänderung,28
sa_flush_cache-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_flush_statistics-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_get_dtt_groupleads-Systemprozedur
Version 11.0.0, neue Funktion,232
sa_get_histogram-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_get_ldapservers_status-Systemprozedur
Version 16.0, neue Funktion,6
sa_get_request_profile-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_get_request_times-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_get_server_messages-Systemprozedur
Version 11.0.0, nicht mehr empfohlen,262
sa_get_table_definition-Systemprozedur
Version 11.0.1, neue Funktion, dbunload,201
Version 16.0, Verhaltensänderung,28
sa_get_user_status-Systemprozedur
Version 11.0.0, neue Funktion,233
Version 16.0, neue Funktion,6
sa_index_density-Systemprozedur
Version 11.0.0, Erweiterung,234
Version 12.0.0, Verhaltensänderung,152
Version 16.0, Verhaltensänderung,28
sa_index_levels-Systemprozedur

Version 16.0, Verhaltensänderung,28
sa_install_feature-Systemprozedur
Version 12.0.0, neue Funktion,133
Version 16.0, Verhaltensänderung,28
sa_internal_text_index_posting-Systemprozedur
Version 11.0.0 neue Funktion, nur für den internen
Gebrauch,233
sa_java_loaded_classes-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_list_cursors-Systemprozedur
Version 12.0.0, neue Funktion,133
sa_load_cost_model-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_locks-Systemprozedur
Version 12.0.0, Erweiterung,145
sa_make_object-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_materialized_view_can_be_immediate-
Systemprozedur
Version 11.0.0, neue Funktion,234
Version 16.0, Verhaltensänderung,28
sa_materialized_view_info-Systemprozedur
Version 11.0.0, Erweiterungen,234
sa_mirror_server_status-Systemprozedur
Version 12.0.0, neue Funktion,133
sa_nchar_terms-Systemprozedur
Version 11.0.0, neue Funktion,233
sa_parse_json-Systemprozedur
Version 16.0, neue Funktion,20
sa_post_login_procedure-Systemprozedur
Version 11.0.0, neue Funktion,234
sa_procedure_profile-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_procedure_profile_summary-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_recompile_views-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_refresh_index_stats-Systemprozedur
Version 11.0.0, neue Funktion,233
sa_refresh_materialized_views-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_refresh_text_indexes-Systemprozedur
Version 11.0.0, neue Funktion,233
Version 16.0, Verhaltensänderung,28
sa_remove_tracing_data-Systemprozedur
Version 16.0, Verhaltensänderung,28
sa_reserved_words-Systemprozedur
Version 12.0.0, neue Funktion,134

- sa_reset_identity-Systemprozedur
 - Tabellen mit Autoinkrement-Spalten neu laden,300
 - Version 16.0, Verhaltensänderung,28
- sa_save_trace_data-Systemprozedur
 - Version 16.0, Verhaltensänderung,28
- sa_send_udp-Systemprozedur
 - Version 16.0, Verhaltensänderung,28
- sa_server_messages-Systemprozedur
 - Version 11.0.0, neue Funktion,262
- sa_server_option-Systemprozedur
 - Version 11.0.0, Erweiterung,234,244
 - Version 12.0.0, Erweiterung,124,134
 - Version 16.0, Erweiterung,17,20
 - Version 16.0, Verhaltensänderung,28
- sa_server_properties-Systemprozedur
 - Version 11.0.0, Erweiterung,235
- sa_set_http_option-Systemprozedur
 - Version 11.0.1, Erweiterung,201
- sa_set_tracing_level-Systemprozedur
 - Version 16.0, Verhaltensänderung,28
- sa_table_fragmentation-Systemprozedur
 - Version 16.0, Verhaltensänderung,28
- sa_table_page_usage-Systemprozedur
 - Version 12.0.0, Erweiterung,134
 - Version 16.0, Verhaltensänderung,28
- sa_table_stats-Systemprozedur
 - Version 16.0, Verhaltensänderung,28
- sa_text_index_handles-Systemprozedur
 - Version 11.0.0 neue Funktion, nur für den internen Gebrauch,233
- sa_text_index_postings-Systemprozedur
 - Version 11.0.0 neue Funktion, nur für den internen Gebrauch,233
- sa_text_index_vocab-Systemprozedur
 - Version 11.0.0, neue Funktion,233
 - Version 12.0.0, neue Funktion,153
- sa_text_index_vocab_nchar-Systemprozedur
 - Version 12.0.0, neue Funktion,133
 - Version 16.0, Verhaltensänderung,28
- sa_unload_cost_model-Systemprozedur
 - Version 16.0, Verhaltensänderung,28
- sa_user_defined_counter_add-Systemprozedur
 - Version 12.0.1, neue Funktion,87
 - Version 16.0, Verhaltensänderung,28
- sa_user_defined_counter_set-Systemprozedur
 - Version 12.0.1, neue Funktion,87
 - Version 16.0, Verhaltensänderung,28
- sa_validate-Systemprozedur
 - Version 11.0.0, Verhaltensänderungen,252
 - Version 16.0, Verhaltensänderung,28
- sa_verify_password-Systemprozedur
 - Version 16.0, Verhaltensänderung,28,32
- saidbc.jar
 - Version 12.0.1, nicht unterstützt,96
- SAJDBC-Serverklasse
 - Version 12.0.0, nicht mehr empfohlen,155
 - Version 16.0, nicht unterstützt,39
- sajdbc.jar
 - Version 16.0, nicht unterstützt,39
- sajdbc4.jar
 - Version 12.0.1, Verhaltensänderung,96
- sample_config.csh
 - Version 12.0.1, Erweiterung, Unix,87
- sample_config.sh
 - Version 12.0.1, Erweiterung, Unix,87
- Sandboxing
 - Version 16.0, neue Funktion,16
- SAP HANA
 - Version 16.0, neue Funktion,11
 - Version 16.0, neue Funktion, MobiLink,42
- SAP Sybase IQ
 - Version 12.0.0, neue Funktion,112
- SAP Sybase IQ 15.2-Unterstützung
 - Version 16.0, entfernt, MobiLink,52
- SAP Sybase IQ 15.4-Unterstützung
 - Version 16.0, neue Funktion, MobiLink,42
- SAP Sybase IQ-Unterstützung
 - Version 12.0.1, neue Funktion, MobiLink,96
- SASpxOptionsBuilder-Klasse [SA .NET 2.0]
 - Version 11.0.0, nicht unterstützt,260
- SATMP-Umgebungsvariable
 - Version 11.0.0, Erweiterung,245
- SAVE OPTION VALUES-Klausel
 - Version 12.0.0, neue Funktion,136
- scale-Option
 - Version 11.0.0, Verhaltensänderung,255
- Scale-Out
 - Version 12.0.0, neue Funktion,123
- Scale-Out, schreibgeschützt
 - Version 12.0.0, neue Funktion,123
- Schattentabelle, Index
 - Version 12.0.0, neue Funktion, MobiLink,157
- SCHEMATA-Zeilengruppe, OLE DB
 - Version 11.0.1, neue Funktion,203
- Schlüsselpaargenerator-Dienstprogramm (createkey)
 - Version 11.0.0, neue Funktion,267

Schnelllader
 Version 11.0.0, Erweiterung,277
 Version 12.0.0, Erweiterung,187
 Version 16.0, Erweiterung,70
 Schnellsetup für Apache unter Linux
 Version 16.0, neue Funktion, Relay Server,54
 Schnellsetup für IIS 7, 7.5 und 8.0
 Version 16.0, neue Funktion, Relay Server,53
 Schnellstart
 Datenbank-Upgrade auf Version 16,298
 Schreib-Prüfsummen
 Version 12.0.0, neue Funktion,127
 Schreibgeschützt
 Version 11.0.0, neue Funktion,
 Spiegeldatenbankzugriff,223
 Schreibgeschützte Datenbank
 Version 11.0.1, Verhaltensänderung,216
 Schreibgeschützte Datenbanken
 Version 12.0.1, Erweiterung,75
 script_full_path-Feld
 Version 16.0, entfernt, MobiLink,52
 scvwen1600.jar
 Version 16.0, nicht mehr empfohlen,69
 SearchBindery-Protokolloption
 Version 11.0.0, nicht unterstützt,260
 secure_feature_key-Option
 Version 16.0, nicht mehr empfohlen,38
 Seitengrößen
 Version 12.0.0, Erweiterung,129
 SELECT-Anweisung
 Version 11.0.0, Erweiterung,220,238
 Version 11.0.1, Erweiterung,202
 Version 12.0.0, Erweiterung,126,139
 Version 12.0.1, Erweiterung,88
 Selektivitätsschätzungen
 Version 12.0.0, Erweiterung,146
 SELinux, Richtlinien
 Version 11.0.0, neue Funktion,245
 send_column_names-Synchronisationsparameter
 Version 16.0, Verhaltensänderung, UltraLite,9
 SendBufferSize-Protokolloption
 Version 12.0.1, Erweiterung,75
 SendColumnNames, erweiterte Option
 Version 16.0, nicht mehr empfohlen, MobiLink,50
 Version 16.0, Verhaltensänderung, MobiLink,8
 Sequenzen
 Version 12.0.0, neue Funktion,124
 Server-Verbindungsparameter
 Version 12.0.0, Verhaltensänderung,150
 ServerEdition-Eigenschaft
 Version 11.0.1, neue Funktion,200
 Servereigenschaften
 Version 11.0.0, neue Funktion,231
 Serverfarm
 Version 12.0.0, Erweiterung, MobiLink,158
 Serverinitiierte Synchronisation
 Version 12.0.1, Erweiterung,82
 Serverinitiierte Synchronisation, in Serverfarm
 Version 11.0.0, Erweiterung,267
 Serverinitiierte Synchronisation, Listener
 Version 11.0.0, Erweiterung,267
 Serverlizenzierung, Dienstprogramm (dblic)
 Version 12.0.0, Erweiterung,129
 Version 12.0.1, Erweiterung,95
 Servermeldungen
 Version 11.0.0, Erweiterung,247
 Servermeldungen, Fenster
 Version 12.0.0, Erweiterung,147
 ServerName-Verbindungsparameter
 Version 12.0.0, Verhaltensänderung,150,155
 Version 12.0.1, Verhaltensänderung,
 OnDemand,74
 ServerNodeAddress-Eigenschaft
 Version 11.0.0, neue Funktion,230
 ServerPort, Protokolloption
 Version 12.0.0, Verhaltensänderung,150
 Serverseitige Sicherungen
 Version 11.0.1, Erweiterung,205
 SET MIRROR OPTION-Anweisung
 Version 12.0.0, neue Funktion,123
 Version 16.0, Erweiterung,12
 SET OPTION-Anweisung
 Version 11.0.0, Verhaltensänderung, Interactive
 SQL,285
 Version 12.0.0, Erweiterung,139
 Version 12.0.0, Verhaltensänderung,156
 Version 12.0.0, Verhaltensänderung, Interactive
 SQL,193
 Version 12.0.1, Verhaltensänderung, UltraLite,110
 SET REMOTE OPTION
 Version 12.0.1, Verhaltensänderung,78
 SET REMOTE OPTION-Anweisung
 Version 12.0.1, Erweiterung,104
 SET TEMPORARY OPTION-Anweisung
 Version 16.0, Verhaltensänderung,32
 setAutoCheckpoint-Methode

- Version 16.0, entfernte API, UltraLite ,66
- setClob
 - PreparedStatement,89
- setDatabaseKey-Methode
 - Version 16.0, Verhaltensänderung, UltraLite ,66
- setSendColumnNames-Methode
 - Version 16.0, Verhaltensänderung, UltraLite,9
- SetupVSPackage
 - Version 12.0.1, Erweiterung,77
- Shadow Paging
 - Version 16.0 entfernte Funktion, UltraLite ,66
- SHAPEFILE-Klausel
 - Version 12.0.0, neue Funktion, FROM-Klausel,119
 - Version 12.0.0, neue Funktion, LOAD TABLE-Anweisungsklausel ,119
 - Version 12.0.1, neue Funktion, INPUT-Anweisung,85
- Shared Memory
 - Version 12.0.0, Erweiterung, Relay Server,166
 - Version 12.0.0, Verhaltensänderung,150
- SHARED_DIR-Umgebungsvariable
 - Version 11.0.0, nicht unterstützt,261
- showPerformanceDataUI
 - Version 16.0, neue Funktion,67
- Sicherheit
 - Version 11.0.0, Erweiterung, UltraLite,273
- Sicherung und Wiederherstellung
 - Version 11.0.0, Erweiterung,224
 - Version 12.0.0, Erweiterung,128
 - Version 16.0, Erweiterung,14
- Sicherungen
 - Version 11.0.1, Erweiterung,205
 - Version 12.0.0, Erweiterung,128
- Sicherungsdienstprogramm (dbbackup)
 - Version 16.0, Erweiterung,14
- Sierra Wireless Aircards
 - Version 11.0.0, nicht unterstützt,268
- SIMILAR TO-Suchbedingung
 - Version 11.0.0, neue Funktion,220
 - Version 11.0.1, Verhaltensänderung,207
- Skripten
 - Upgrade,335
- Skriptgesteuerter Upload
 - Version 16.0, Verhaltensänderung, MobiLink,51
- SMALLINT UNSIGNED-Datentyp
 - Version 16.0, neue Funktion,24
- Smartphone
 - Version 11.0.0, Erweiterung,287
- Version 11.0.0, neue Funktion, UltraLite,275
- Version 12.0.1, Erweiterung, UltraLite,106
- SMS-Listener
 - Version 16.0, Verhaltensänderung,50
- SMTP
 - Version 12.0.0, Erweiterung,134
 - Version 16.0, Verhaltensänderung,34
- SNMP
 - Version 11.0.0, Erweiterung,248
- Sofortansichten
 - Version 11.0.0, neue Funktion,222
 - Version 16.0, Erweiterung,22
- Software-Updates suchen
 - Version 12.0.0, Erweiterung,187
- Softwarekompatibilität
 - UltraLite-Upgrades,338
- sp_alter_secure_feature_key-Systemprozedur
 - Version 16.0, neue Funktion,15
 - Version 16.0, Verhaltensänderung,32
- sp_auth_sys_role_info-Systemprozedur
 - Version 16.0, neue Funktion,2
- sp_copy_directory-Systemprozedur
 - Version 16.0, neue Funktion,20
 - Version 16.0, Verhaltensänderung,28
- sp_copy_file-Systemprozedur
 - Version 16.0, neue Funktion,20
 - Version 16.0, Verhaltensänderung,28
- sp_create_directory-Systemprozedur
 - Version 16.0, neue Funktion,20
 - Version 16.0, Verhaltensänderung,28
- sp_create_secure_feature_key-Systemprozedur
 - Version 16.0, neue Funktion,15
 - Version 16.0, Verhaltensänderung,32
- sp_delete_directory-Systemprozedur
 - Version 16.0, neue Funktion,20
 - Version 16.0, Verhaltensänderung,28
- sp_delete_file-Systemprozedur
 - Version 16.0, neue Funktion,20
 - Version 16.0, Verhaltensänderung,28
- sp_displayroles-Systemprozedur
 - Version 16.0, neue Funktion,2
- sp_drop_secure_feature_key-Systemprozedur
 - Version 16.0, neue Funktion,15
- sp_forward_to_remote_server-Systemprozedur
 - Version 12.0.1, neue Funktion,87
 - Version 16.0, Verhaltensänderung,28
- sp_get_last_synchronize_result-Systemprozedur
 - Version 16.0, Verhaltensänderung,28

sp_has_role-Systemprozedur	Version 16.0, Erweiterung,27
sp_list_directory-Systemprozedur	Spaltenattribute
Version 16.0, neue Funktion,2	nächsten verfügbaren Wert in neu erstellter
sp_list_secure_feature_keys-Systemprozedur	Datenbank abfragen,300
Version 16.0, Verhaltensänderung,28	Spaltennamen
sp_move_directory-Systemprozedur	Version 16.0, Verhaltensänderung,8
Version 16.0, neue Funktion,20	Version 16.0, Verhaltensänderung, UltraLite,66
Version 16.0, Verhaltensänderung,28	Spaltennamen senden
sp_move_file-Systemprozedur	Version 16.0, Verhaltensänderung,8,50
Version 16.0, neue Funktion,21	Version 16.0, Verhaltensänderung, UltraLite,66
sp_objectpermission-Systemprozedur	Spaltenzuordnung
Version 16.0, neue Funktion,2	Version 16.0, Verhaltensänderung,49
sp_password-Systemprozedur	Spatial Viewer
Version 16.0, Verhaltensänderung,32	Version 12.0.0, neue Funktion,120
sp_proc_priv-Systemprozedur	SpatialUtilities-Klasse
Version 16.0, neue Funktion,2	Version 12.0.0, neue Funktion, MobiLink,161
sp_remote_columns-Systemprozedur	Speicher
Version 16.0, Verhaltensänderung,28	Version 12.0.0, Erweiterung, MobiLink,158
sp_remote_exported_keys-Systemprozedur	Speichermodus
Version 16.0, Verhaltensänderung,28	Version 11.0.0, neue Funktion,228
sp_remote_imported_keys-Systemprozedur	Sperren
Version 16.0, Verhaltensänderung,28	Version 11.0.0, neue Funktion, Interactive
sp_remote_primary_keys-Systemprozedur	SQL ,279
Version 16.0, Verhaltensänderung,28	Version 11.0.0, Verhaltensänderung,258
sp_remote_procedures-Systemprozedur	Version 11.0.1, Verhaltensänderung, Interactive
Version 16.0, Verhaltensänderung,28	SQL ,216
sp_remote_tables-Systemprozedur	Version 12.0.0, Erweiterung,145
Version 16.0, Verhaltensänderung,28	Version 12.0.0, Verhaltensänderung,151
sp_sys_priv_role_info-Systemprozedur	Version 16.0, Erweiterung, MobiLink,41
Version 16.0, neue Funktion,2	Spiegeldatenbanken
sp_trace_event_fields-Systemprozedur	Version 11.0.0, Erweiterung,223
Version 16.0, neue Funktion,10	Spiegeln
sp_trace_event_session_events-Systemprozedur	Upgrade,314
Version 16.0, neue Funktion,10	Version 11.0.0, Erweiterung,222
sp_trace_event_session_target_options-Systemprozedur	Spiegelung
Version 16.0, neue Funktion,10	Upgrade (Datenbankserver) auf Hauptversion,314
sp_trace_event_session_targets-Systemprozedur	Version 16.0, neue Funktion,11
Version 16.0, neue Funktion,10	Version 16.0, Verhaltensänderung,30
sp_trace_event_sessions-Systemprozedur	SPX
Version 16.0, neue Funktion,10	Version 11.0.0, nicht unterstützt,260
sp_trace_events-Systemprozedur	SpxOptionsBuilder-Eigenschaft [SA .NET 2.0]
Version 16.0, neue Funktion,10	Version 11.0.0, nicht unterstützt,260
sp_use_secure_feature_key-Systemprozedur	SpxOptionsString-Eigenschaft [SA .NET 2.0]
Version 16.0, neue Funktion,15	Version 11.0.0, nicht unterstützt,260
Version 16.0, Verhaltensänderung,32	SQL Anywhere
Spalten	Version 11.0.1, neue Funktion,214
	SQL Anywhere 11, CustDB-Datenquelle
	Version 11.0.0, Verhaltensänderung,288

- SQL Anywhere 11, Demo-Datenquelle
 - Version 11.0.0, Verhaltensänderung,288
- SQL Anywhere 11.0.1-Unterstützung
 - Version 16.0, entfernt, MobiLink,52
- SQL Anywhere 16
 - Upgrade,289
 - Version 16.0, neue Funktion, MobiLink,42
- SQL Anywhere Explorer
 - Version 12.0.0, nicht mehr unterstützt,155
- SQL Anywhere for MS Access Migration, Dienstprogramm
 - Version 11.0.0, nicht unterstützt,247
- SQL Anywhere ODBC-Treiber
 - Version 12.0.0, Erweiterung,141
- SQL Anywhere OnDemand, Unterstützung
 - Version 12.0.1, neue Funktion,73
- SQL Anywhere, C-API
 - Version 11.0.0, neue Funktion,240
- SQL Anywhere, MIB
 - Version 11.0.0, Erweiterung,248
- SQL Anywhere, neue Funktionen
 - Version 11.0.0,220
 - Version 12.0.0,119
 - Version 12.0.1,84
 - Version 16.0,9
- SQL Anywhere, nicht mehr empfohlene Funktionen
 - Version 11.0.0,259
 - Version 11.0.1,210
 - Version 12.0.0,154
 - Version 12.0.1,96
 - Version 16.0,37
- SQL Anywhere, nicht mehr weiterentwickelte Funktionen
 - Version 11.0.0,259
 - Version 11.0.1,210
 - Version 12.0.0,154
 - Version 12.0.1,96
 - Version 16.0,37
- SQL Anywhere, Passthrough-Skripten
 - Version 11.0.0, Erweiterung, UltraLite,270
- SQL Anywhere, SNMP-Erweiterungsagent
 - Version 11.0.0, Erweiterung,248
- SQL Anywhere, Verhaltensänderungen
 - Version 11.0.0,250
 - Version 11.0.1,206
 - Version 12.0.0,148
 - Version 12.0.1,94
 - Version 16.0,28
- SQL Anywhere-Monitor
 - Version 12.0.0, Erweiterung, Relay Server,166
 - Version 16.0, Verhaltensänderung,71
- SQL Anywhere-Monitor, Verhaltensänderungen
 - Version 12.0.0,196
 - Version 12.0.1,116
- SQL Anywhere-ODBC-Treiber
 - Version 12.0.0, Verhaltensänderung,152
- SQL Anywhere-Plug-In
 - unterstützte Versionen,289
- SQL Flagger
 - Version 12.0.0, Erweiterung,148
- SQL Remote
 - fehlende Nachrichten verarbeiten,104
 - Upgrade,341
 - Version 12.0.1, Verhaltensänderung,78
- SQL Remote, neue Funktionen
 - Version 11.0.0,269
 - Version 12.0.0,172
 - Version 12.0.1,104
- SQL Remote, Verhaltensänderungen
 - Version 11.0.0,269
 - Version 12.0.1,104
- SQL Remote-Nachrichten
 - Erweiterung in Version 12.0.1,104
- SQL Remote-Nachrichtenagent (dbremote)
 - Version 12.0.0, Verhaltensänderung,118
- SQL SECURITY-Klausel
 - Version 11.0.0, neue Funktion,233
- SQL, Passthrough
 - Version 11.0.0, neue Funktion,265
- SQL-Anweisungen aus der Ergebnismenge generieren
 - Version 11.0.0, Erweiterung,282
- SQL-Passthrough
 - Version 12.0.0 nicht mehr empfohlen, MobiLink,170
- SQL-Schlüsselwörter
 - Version 12.0.0, Erweiterung,134
- SQL-Unterstützung
 - Version 11.0.1, Erweiterung, UltraLiteJ,213
- sqlanydb
 - Version 11.0.0, neue Funktion,240
 - Version 12.0.0, neue Funktion,143
- SQLANYSAMPI2-Umgebungsvariable
 - Version 12.0.1, Erweiterung, Unix,87
- sqlanywhere_commit-Funktion (nicht mehr empfohlen)
 - Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_connect-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_data_seek-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_disconnect-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_error-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_errorcode-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_execute-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_fetch_array-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_fetch_field-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_fetch_object-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_fetch_row-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_free_result-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_identity-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_num_fields-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_num_rows-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_pconnect-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_query-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_result_all-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_rollback-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

sqlanywhere_set_option-Funktion (nicht mehr empfohlen)

Version 11.0.0, nicht mehr empfohlen,252

SQLGetConnectAttr

Version 11.0.0, Verhaltensänderung,253

SQLTables

Version 12.0.0, Verhaltensänderung,152

Version 12.0.1, Verhaltensänderung,95

Version 16.0, Verhaltensänderung,36

SRID-Klausel

Version 12.0.1, neue Funktion, INPUT-Anweisung,85

SSL

Version 11.0.0, Verhaltensänderung,254

st_geometry_asbinary_format, Eigenschaft

Version 12.0.0, neue Funktion,121

st_geometry_asbinary_format, Option

Version 12.0.0, neue Funktion,121

st_geometry_astext_format, Eigenschaft

Version 12.0.0, neue Funktion,121

st_geometry_astext_format, Option

Version 12.0.0, neue Funktion,121

st_geometry_asxml_format, Eigenschaft

Version 12.0.0, neue Funktion,121

st_geometry_asxml_format, Option

Version 12.0.0, neue Funktion,121,131

ST_GEOMETRY_COLUMNS

Version 12.0.0, neue Funktion, konsolidierte Ansicht,121

st_geometry_describe_type, Eigenschaft

Version 12.0.0, neue Funktion,121

st_geometry_describe_type, Option

Version 12.0.0, neue Funktion,121

st_geometry_dump-Systemprozedur

Version 12.0.0, neue Funktion,120

st_geometry_interpolation, Option

Version 12.0.1, neue Funktion,85

st_geometry_load_shapefile-Systemprozedur

Version 12.0.1, neue Funktion,85

st_geometry_on_invalid, Eigenschaft

Version 12.0.0, neue Funktion,121

st_geometry_on_invalid, Option

Version 12.0.0, neue Funktion,121

st_geometry_predefined_srs-Systemprozedur

- Version 12.0.0, neue Funktion, für interne Verwendung,120
- Version 16.0, Verhaltensänderung,28
- st_geometry_predefined_uom-Systemprozedur
 - Version 12.0.0, neue Funktion, für interne Verwendung,120
 - Version 16.0, Verhaltensänderung,28
- ST_SPATIAL_REFERENCE_SYSTEMS
 - Version 12.0.0, neue Funktion, konsolidierte Ansicht,121
- ST_UNITS_OF_MEASURE
 - Version 12.0.0, neue Funktion, konsolidierte Ansicht,121
- ST_WithinDistanceFilter
 - Methode, Version 12.0.1 Erweiterung,86
- Stack-Größe
 - Version 12.0.0, Erweiterung, Windows Mobile,145
- Standard-Vorsichtsmaßnahmen für Upgrade Info,297
- Standarddatenbankserver
 - Version 11.0.1, Erweiterung,205
- START AT-Klausel
 - Version 12.0.1, Erweiterung,88
- START DATABASE-Anweisung
 - Version 12.0.0, Erweiterung,127
 - Version 16.0, Erweiterung,12,16,24
 - Version 16.0, Verhaltensänderung,32,37,70
- START LOGGING-Anweisung
 - Version 12.0.0, Erweiterung,191
- START SERVER-Anweisung
 - Version 12.0.0, neue Funktion,140
- START SYNCHRONIZATION SCHEMA CHANGE-Anweisung
 - Version 12.0.0, neue Funktion,158
 - Version 12.0.0, neue Funktion, MobiLink,163
 - Version 12.0.1, Erweiterung, OnDemand,74
 - Version 16.0, Erweiterung,47
- StatementDescribes-Eigenschaft
 - Version 11.0.1, neue Funktion, Datenbankeigenschaft,200
 - Version 11.0.1, neue Funktion, Verbindungseigenschaft,199
- StatementPostAnnotates-Eigenschaft
 - Version 11.0.1, neue Funktion, Datenbankeigenschaft,200
 - Version 11.0.1, neue Funktion, Verbindungseigenschaft,199
- StatementPostAnnotatesSimple-Eigenschaft
 - Version 11.0.1, neue Funktion, Datenbankeigenschaft,200
 - Version 11.0.1, neue Funktion, Verbindungseigenschaft,199
- StatementPostAnnotatesSkipped-Eigenschaft
 - Version 11.0.1, neue Funktion, Datenbankeigenschaft,200
 - Version 11.0.1, neue Funktion, Verbindungseigenschaft,199
- StatisticsCleaner, Eigenschaft
 - Version 12.0.0, neue Funktion,134
- Statistik
 - Version 12.0.0, Erweiterung,124
- Statistiken aufräumen
 - Version 12.0.0, neue Funktion,124
- Statistikwächter
 - Version 12.0.0, neue Funktion,124
- Statusinformationsdateien
 - Version 12.0.0, Verhaltensänderung,151
- Sternchen
 - Version 11.0.1, Verhaltensänderung, Volltextsuche ,206
- STOP ENGINE-Anweisung
 - Version 12.0.0, nicht mehr empfohlen,155
- STOP SERVER-Anweisung
 - Version 12.0.0, neue Funktion,140
- STOP SYNCHRONIZATION SCHEMA CHANGE-Anweisung
 - Version 12.0.0, neue Funktion,158
 - Version 12.0.0, neue Funktion, MobiLink,163
- Stoppen von Datenbanken
 - Version 12.0.0, Verhaltensänderung,148
- Stoppen, Datenbanken
 - Version 11.0.0, Verhaltensänderung,254
- StreamErrorParameters-Eigenschaft
 - Version 11.0.1, Erweiterung, UltraLite,212
- StreamHTTPParms.getE2eePublicKey-Methode [UltraLiteJ]
 - Version 12.0.1, Erweiterung,79
- StreamHTTPParms.getExtraParameters-Methode [UltraLiteJ]
 - Version 12.0.1, neue Funktion,81
- StreamHTTPParms.isRestartable-Methode [UltraLiteJ]
 - Version 12.0.1, neue Funktion,82
- StreamHTTPParms.setE2eePublicKey-Methode [UltraLiteJ]
 - Version 12.0.1, Erweiterung,79

StreamHTTPParams.setExtraParameters-Methode [UltraLiteJ]	Sybase Adaptive Server Enterprise 15.0-Unterstützung
Version 12.0.1, neue Funktion,81	Version 16.0, entfernt, MobiLink,52
StreamHTTPParams.setRestartable-Methode [UltraLiteJ]	Sybase Central
Version 12.0.1, neue Funktion,82	ältere Datenbanken verwalten,295
StreamHTTPParams.setURLSuffix-Methode [UltraLiteJ]	Datenbanken der Version 10 und höher neu aufbauen,300
Version 12.0.1, Erweiterung,82	Datenbanken der Version 9 und früher neu aufbauen,307
StreamHTTPParams.setZlibCompression-Methode [UltraLiteJ]	Upgrade einer Datenbank,311
Version 12.0.1, Erweiterung,78	Version 12.0.0, Verhaltensänderung,189
StreamHTTPParams.setZlibDownloadWindowSize-Methode [UltraLiteJ]	Version 16.0, Verhaltensänderung,68
Version 12.0.1, Erweiterung,78	Versionsunterstützung,289
StreamHTTPParams.setZlibUploadWindowSize-Methode [UltraLiteJ]	Sybase Central Plug-In
Version 12.0.1, Erweiterung,78	Version 16.0, neue Funktionen,68
StreamHTTPParams.setZlibUploadWindowSize-Methode [UltraLiteJ]	Sybase Central, neue Funktionen
Version 12.0.1, Erweiterung,78	Version 11.0.0,277
StreamHTTPParams.zlibCompressionEnabled-Methode [UltraLiteJ]	Version 11.0.1,214
Version 12.0.1, Erweiterung,78	Version 12.0.0,186
StreamHTTPParams.setTrustedCertificates-Methode [UltraLiteJ]	Version 12.0.1,111
Version 12.0.1, Erweiterung,79	Version 16.0,68
StreamsUsed-Eigenschaft	Sybase Central, Verhaltensänderungen
Version 11.0.0, neue Funktion,231	Version 11.0.0,280
STRIP-Klausel	Version 11.0.1,215
Version 16.0, nicht unterstützt,37	Version 12.0.0,189
Stromausfall	Version 16.0,68
Version 12.0.0, Erweiterung,147	Sybase IQ
Suchbedingungen	Version 16.0, neue Funktionen,68
Version 12.0.0, Erweiterung,139	Sybase Relay Server, Hosting-Dienst
Version 12.0.0, neue Funktion,136	Version 11.0.0, neue Funktion ,264
support, Dienstprogramm	Sybase Replication Server
Version 12.0.0, Verhaltensänderung,118	in Version 12.0.0 nicht unterstützt,117
Supportanfragen-Dienstprogramm	Symbolleisten
Version 11.0.0, Erweiterung,287	Version 11.0.0, Erweiterung, Interactive SQL ,282
Supportanfragen-Dienstprogramm (dbsupport)	SYNC_PROFILE_OPTION_VALUE-Funktion
Version 12.0.0, Erweiterung,130	Version 11.0.1, Erweiterung, UltraLite,212
Supportpakete	Synchronisationsdatenstrom, Sicherheit
Datenbankspiegelung,318	Version 11.0.0, neue Funktion,267
SuppressInfoForDataTypes, ODBC-Verbindungsparameter	Synchronisationsmodell löschen
Version 16.0, neue Funktion,14	Version 16.0, neue Funktion, MobiLink-Plug-In,48
SUSER_NAME-Funktion	Synchronisationsmodell testen
Version 16.0, Verhaltensänderung,35	Version 16.0, neue Funktion, MobiLink-Plug-In,47
SWITCHOFFSET-Funktion	Synchronisationsmodelle
Version 12.0.0, neue Funktion,135	Version 12.0.0, Verhaltensänderung, MobiLink,169
	Version 12.0.1, neue Funktion, MobiLink,99
	Synchronisationsparameter
	Version 16.0, Verhaltensänderung, UltraLite,9

- Synchronisationsprofile
 - Version 11.0.0, Erweiterung, UltraLite,270
 - Version 11.0.0, neue Funktion,266
- Synchronisationsprofiloptionen
 - Version 16.0, Verhaltensänderung, UltraLite,9
- SynchronizationSchemaChangeActive, Eigenschaft
 - Version 12.0.0, neue Funktion,132
- SYNCHRONIZE-Anweisung
 - Version 12.0.0, neue Funktion,158
- SyncResult-Schnittstelle
 - Version 16.0, Erweiterung, UltraLiteJ,61
- SYS_REPLICATION_ADMIN_ROLE-Systemrolle
 - Version 16.0, neue Funktion, MobiLink-Client,45
- SYS_RUN_REPLICATION_ROLE-Systemrolle
 - Version 16.0, neue Funktion, MobiLink-Client,45
- SYS_SPATIAL_ADMIN_ROLE, Gruppe
 - Version 12.0.0, neue Funktion,129
- SYSCERTIFICATE
 - Version 16.0, neue Funktion,25
- SYSSTAT
 - Version 16.0, Erweiterung,25
- SYSDATETIMEOFFSET-Funktion
 - Version 12.0.0, neue Funktion,135
- SYSEXTERNENVOBJECT
 - Version 16.0, Erweiterung,25
- SYSROLEGRANT
 - Version 16.0, neue Funktion,4
- SYSROLEGRANTEXT
 - Version 16.0, neue Funktion,4
- SYSROLEGRANTS
 - Version 16.0, neue Funktion,4
- SYSGROUP
 - Version 16.0, Verhaltensänderung,4
- SYSGROUPS
 - Version 16.0, nicht mehr empfohlen,4
- SYSHISTORY
 - Version 16.0, Erweiterung,25
- SYSINDEXES
 - Version 11.0.0, Verhaltensänderung,250
- SYSJAR
 - Version 16.0, Erweiterung,25
- SYSJAVACLASS
 - Version 16.0, Erweiterung,25
- SYSLDAPSERVER
 - Version 16.0, neue Funktion,7
- SYSLOGINMAP
 - Version 11.0.0, neue Funktion,250
- SYSLOGINPOLICY
 - Version 11.0.0, neue Funktion,250
- SYSLOGINPOLICYOPTION
 - Version 11.0.0, neue Funktion,250
- SYSMIRROROPTION, Systemansicht
 - Version 12.0.0, neue Funktion,123
- SYSMIRRORSERVER
 - Version 12.0.0, neue Funktion, Systemansicht,123
- SYSMIRRORSERVEROPTION
 - Version 12.0.0, neue Funktion, Systemansicht,123
- SYSOBJECT
 - Version 11.0.0, Verhaltensänderung,250
 - Version 12.0.0, Erweiterung,145
 - Version 16.0, Erweiterung,25
- SYSROCPARM
 - Version 12.0.0, Erweiterung, Systemansicht,144
- SYSREMOTE USER
 - Version 16.0, Erweiterung,25
- SYSSEQUENCE
 - Version 12.0.0, neue Funktion,144
- SYSSEQUENCEPERM
 - Version 12.0.0, neue Funktion,144
- SYSPATIALREFERENCESYSTEM
 - Version 12.0.0, neue Funktion, Systemansicht,121
- syssyncresult-Systemtabelle
 - Version 16.0, Erweiterung, UltraLite,60
- SYSTAB
 - Version 16.0, Erweiterung,25
- SYSTABAUTH
 - Version 16.0, Erweiterung,4
- SYSTABCOL
 - Version 12.0.0, Erweiterung, Systemansicht,144
- SYSTABLEPERM
 - Version 16.0, Erweiterung,4
- SYSTEM PROCEDURE AS DEFINER-Klausel
 - ALTER DATABASE, Version 16.0, neue Funktion,29
 - CREATE DATABASE, Version 16.0, neue Funktion,28
 - Version 16.0, neue Funktion, CREATE DATABASE-Anweisung,28
- Systemansichten
 - Version 11.0.0, Verhaltensänderung,250
 - Version 16.0, Erweiterung,25
- Systemereignisse unter Android
 - Version 16.0, Erweiterung, UltraLiteJ,64
- Systemmonitor
 - Version 11.0.0, Erweiterung,229
- Systemobjekte

Version 16.0, Erweiterung, MobiLink-Plug-In,48
 Systempfad
 Dienstprogramme,298
 UltraLite-Dienstprogramme,337
 Systemprozedur sa_enable_auditing_type
 Version 16.0, Verhaltensänderung,28
 Systemprozeduren
 Version 12.0.1, Verhaltensänderung,
 OnDemand,74
 Version 16.0, Verhaltensänderung,28
 Systemschlüssel für gesicherte Funktionen
 Version 16.0, Verhaltensänderung,31
 Systemtabellen
 Version 11.0.0, Verhaltensänderung,250
 Version 16.0, Erweiterung,25
 SYSTEXTCONFIG
 Version 11.0.0, neue Funktion,250
 SYSTEXTIDX
 Version 11.0.0, neue Funktion,250
 Version 16.0, Erweiterung,25
 SYSTEXTIDXTAB
 Version 11.0.0, neue Funktion,250
 SYSUNITOFMEASURE
 Version 12.0.0, neue Funktion, Systemansicht,121
 SYSUSER
 Systemansicht, Version 16.0,
 Verhaltensänderung,37
 Version 16.0, Erweiterung,4,7,25
 SYSUSERAUTHORITY
 Version 16.0, Verhaltensänderung,4
 SYSUSERTYPE
 Version 12.0.0, Erweiterung,144
 SYSVIEW
 Version 16.0, Erweiterung,25

T

Tabellen, nur Download
 Version 12.0.1 neue Funktion, UltraLite,105
 Tabellen-API
 Version 11.0.0, Erweiterung, UltraLite,274
 Tabellenverschlüsselung
 Version 11.0.0, Erweiterung,225
 Tabellenzuordnungseitor
 Version 16.0, Verhaltensänderung,47
 Tabellenzuordnungsrichtung
 Version 12.0.1, Verhaltensänderung, MobiLink,83
 Tastenkürzel

Version 11.0.0, neue Funktion,277,280
 Version 12.0.0, Verhaltensänderung,193
 TCP/IP
 Version 12.0.0, Verhaltensänderung,149
 Version 12.0.1, Verhaltensänderung,75
 TcpIpAddresses-Eigenschaft
 Version 11.0.0, neue Funktion,231
 TDS-Kommunikationsprotokoll
 Version 12.0.0, Erweiterung,144
 Temporäre Dateien
 Version 11.0.0, Erweiterung, Unix,245
 Temporäre Funktionen
 Version 11.0.1, Erweiterung,203
 Temporäre Prozeduren
 Version 11.0.1, Erweiterung,203
 Temporäre Tabellen
 Version 11.0.0, Verhaltensänderung,258
 Version 12.0.0, Erweiterung,133
 Temporäre Verbindungen
 Version 12.0.0, Erweiterung,128
 TERM BREAKER-Klausel
 Version 12.0.0, neue Funktion,126
 Textindizes
 Version 11.0.0, neue Funktion,220
 Version 11.0.1, Verhaltensänderung,206
 Version 12.0.1, Erweiterung,90
 Version 16.0, Erweiterung,22
 Textkonfigurationsobjekte
 Version 11.0.0, neue Funktion,220
 Textpläne
 Version 11.0.0, nicht mehr empfohlen, Interactive
 SQL-Funktionen,282
 Textvervollständigung
 Version 12.0.1, Erweiterungen,112
 Version 16.0, Erweiterung,70
 ThreadDeadlocksAvoided, Eigenschaft
 Version 12.0.0, neue Funktion,133
 ThreadDeadlocksReported, Eigenschaft
 Version 12.0.0, neue Funktion,133
 Threads
 Version 12.0.0, Verhaltensänderung,148
 Timeout-Protokolloption
 Version 12.0.1, Erweiterung,75
 TIMESTAMP WITH TIME_ZONE, Datentyp
 Version 12.0.0, neue Funktion,140
 TIMESTAMP WITH TIME_ZONE-Datentyp
 Version 12.0.0, neue Funktion,159
 timestamp_with_time_zone_format, Option

- Version 12.0.0, neue Funktion,131
- TimestampWithTimeZone-Klasse
 - Version 12.0.0, neue Funktion, MobiLink,161
- TINYINT UNSIGNED-Datentyp
 - Version 16.0, neue Funktion,24
- TLS
 - Version 12.0.0, Erweiterung, MobiLink,158
 - Version 16.0, Erweiterung, MobiLink,41
- TLS-Identitäten, gespeichert in der Datenbank
 - Version 12.0.1, neue Funktion, UltraLite,105
- TLS-Unterstützung
 - Version 16.0, Verhaltensänderung,7
- tls_type-Argument
 - ENCRYPTION-Verbindungsparameter,7
- tls_type-Protokolloption
 - ec, Datenbankserveroption,7
 - Version 16.0, nicht mehr empfohlen, MobiLink,53
- TODATETIMEOFFSET-Funktion
 - Version 12.0.0, neue Funktion,135
- TOP-Klausel
 - Version 12.0.1, Erweiterung,88
- total_download_row_count
 - Version 16.0, Erweiterung, UltraLite C/C++,61
 - Version 16.0, Erweiterung, UltraLite.NET,61
- TRACEBACK-Funktion
 - Version 16.0, Verhaltensänderung,34
- Transact-SQL, Outer Joins
 - Fehlerbehandlung, Upgrades auf Version 16,306
- Transaktionslog
 - Version 11.0.0, Erweiterung,221
- Transaktionslog, Dienstprogramm
 - Version 12.0.0, Verhaltensänderung,118
- Transaktionslog-Offsets
 - Upgrade von Datenbankdateien,303
- TREAT-Funktion
 - Version 12.0.0, neue Funktion,120
- TRIM_ARRAY-Funktion
 - Version 16.0, neue Funktion,9
- TRUNCATE TEXT INDEX-Anweisung
 - Version 11.0.0, neue Funktion,237
- TRUNCATE-Privileg
 - Version 16.0, neue Funktion,1
- truncate_deletes
 - Version 16.0, neue Funktion, UltraLite,57
- truncate_with_auto_commit-Option
 - Version 11.0.0, nicht unterstützt,260
- trusted_certificate_name-Option
 - Version 16.0, neue Funktion,45

- trusted_certificates-Option
 - Version 16.0, neue Funktion, MobiLink,43
- trusted_certificates_file-Eigenschaft
 - Version 16.0, neue Funktion,19
- trusted_certificates_file-Option
 - Version 16.0, neue Funktion,6
- TRY-Anweisung
 - Version 16.0, neue Funktion,23
- TSL
 - Version 11.0.0, Verhaltensänderung,254
- tsql_hex_constant-Option
 - Version 11.0.0, nicht unterstützt,260

U

- Überlapptes Cookie erneuern
 - Relay Server,54
- Überlauffehler
 - Version 12.0.0, Erweiterung,146
- Übersicht, Fensterausschnitt
 - Version 11.0.0, neue Funktion,278
- Übersicht, Registerkarte
 - Version 12.0.0, Erweiterung,189
- Überwachungsliste
 - Version 11.0.0, neue Funktion,234
- UCA-Kollation
 - Version 12.0.0, Erweiterung,147
- UDP-Listener
 - Version 16.0, Verhaltensänderung,50
- ul_column_name_type-Enumeration [UltraLite C++]
 - Version 12.0.1, neue Funktion,83
- ul_stream_error-Struktur
 - Version 11.0.0, Erweiterung, UltraLite,273
- ul_sync_info
 - Version 11.0.0, Verhaltensänderung, UltraLite,276
- ul_sync_stats_download
 - Version 16.0, Erweiterung, UltraLite C/C++,60
- ul_sync_stats_upload
 - Version 16.0, Erweiterung, UltraLite C/C++,60
- ulcpp11.cpp
 - Version 16.0, entfernte API, UltraLite ,67
- uleng-Dienstprogramm
 - Version 12.0.0, Verhaltensänderung,118
- uliface.h
 - Version 16.0, entfernte API, UltraLite ,67
- ULjDbT
 - Version 11.0.1, UltraLiteJ-Dienstprogramm,214
- ULjException-Klasse [UltraLiteJ]

- Version 12.0.1, Erweiterung,82
- ULODBC-Serverklasse
 - Version 11.0.0, neue Funktion,247
- ULResultSetSchema.GetColumnName-Methode [UltraLite C++]
 - Version 12.0.1, Erweiterung,83
- UltraLite
 - Fehlermeldungen, neu,110
 - Upgrading von Datenbanken und Anwendungen,337
- UltraLite C/C++
 - Version 16.0, Erweiterung, UltraLite,60
- UltraLite für M-Business Anywhere
 - nicht mehr empfohlen,107
 - Version 11.0.0, Erweiterung, UltraLite,275
 - Version 12.0.0, Erweiterung, UltraLite,176
 - Version 16.0, entfernte API, UltraLite ,67
- UltraLite Java Edition-Datenbanken
 - neue Funktionen in Version 16.0,56
- UltraLite, ALTER DATABASE SCHEMA FROM FILE-Anweisung
 - Version 11.0.0, Erweiterung, UltraLite,272
- UltraLite, ALTER SYNCHRONIZATION PROFILE-Anweisung
 - Version 11.0.0, neue Funktion, UltraLite,270
- UltraLite, Anwendungen
 - Version 11.0.0, neue Funktion, UltraLite-Zeilen-Traversal,274
- UltraLite, C/C++-API
 - Version 11.0.0, Erweiterung, UltraLite,274
- UltraLite, Datenbanken
 - Version 11.0.0, Erweiterungen,270
- UltraLite, Dienstprogramm zum Entladen von Datenbanken in XML (ulunload)
 - Version 11.0.0, Erweiterung, UltraLite,273
- UltraLite, Dienstprogramm zum Validieren von Datenbanken (ulvalid)
 - Version 11.0.0, neue Funktion, UltraLite,271
- UltraLite, ESQl
 - Version 11.0.0, Erweiterung, UltraLite,275
- UltraLite, isolation_level, Option
 - Version 11.0.0, neue Funktion, UltraLite,271
- UltraLite, LOAD TABLE-Anweisung
 - Version 11.0.0, Erweiterung, UltraLite,272
- UltraLite, neue Funktionen
 - Version 11.0.0,270
 - Version 11.0.1,212
 - Version 12.0.0,173
- Version 12.0.1,104
- Version 16.0,56
- UltraLite, SELECT-Anweisung
 - Version 11.0.0 Erweiterung, UltraLite,270
- UltraLite, SYNCHRONIZE-Anweisung
 - Version 11.0.0, Erweiterung, UltraLite,270
- UltraLite, Tabellen
 - Version 11.0.0, neue Funktion, UltraLite-Zeilen-Traversal,274
- UltraLite, Tabellen-API
 - Version 11.0.0, Erweiterung, UltraLite,274
- UltraLite, Verhaltensänderungen
 - Version 11.0.0,275
 - Version 12.0.0,177
 - Version 12.0.1,110
 - Version 16.0,65
- UltraLite-Clients
 - Version 11.0.0, Erweiterung, UltraLite,272
- UltraLite-Datenbanken
 - neue Funktionen in Version 12.0.0,173
 - neue Funktionen in Version 12.0.1,104
 - neue Funktionen in Version 16.0,56
 - Upgrade von früheren Versionen,337
 - Version 11.0.0, neue Funktion,270
- UltraLite-Dienstprogramm, Datenbank initialisieren (ulinit)
 - Version 11.0.0, Erweiterung, UltraLite,273
- UltraLite.NET
 - asynchrone Synchronisation,106
 - Version 11.0.0, Erweiterung, UltraLite,275
 - Version 12.0.1, Erweiterung, UltraLite,106
 - Version 16.0, Erweiterung, UltraLite,61
- UltraLite_Connection-Objekt
 - Version 11.0.0, neue Funktion, UltraLite-Zeilen-Traversal,274
- UltraLiteJ
 - Codebeispiel, Speicherorte,106
 - Version 12.0.1, Erweiterung, UltraLite,107
 - Version 16.0, Erweiterung,61
- UltraLiteJ, Datenbanken
 - Version 12.0.0, entfernte und nicht mehr empfohlene Funktionen,184
 - Version 12.0.0, Erweiterung,184
- UltraLiteJ, neue Funktionen
 - Version 11.0.1,213
- UltraLiteJ, Verhaltensänderungen
 - Version 11.0.1,213,214
- UltraLiteJ-Codebeispiel, Speicherorte

- Version 12.0.1, neue Funktion, UltraLite,106
- UltraLiteJ-Datenbanken
 - neue Funktionen in Version 12.0.1,104
- UltraLiteJ-Datenbankübertragung, Dienstprogramm
 - BlackBerry,214
- ulvalid-Dienstprogramm
 - Version 11.0.0, neue Funktion, UltraLite,271
- UNION-Anweisung
 - Version 11.0.0, Erweiterung,239
 - Version 11.0.0, Verhaltensänderung,257
- UniqueIdentifier-Eigenschaft
 - Version 11.0.0, nicht unterstützt,259
- Unix
 - Version 11.0.0, neue Funktion,245
 - Version 12.0.0, Erweiterung,145
 - Version 12.0.0, Verhaltensänderung, Daemon,118
 - Version 16.0, Erweiterung,26
- UNLOAD TABLE-Anweisung
 - Version 11.0.0, Erweiterung,239
 - Version 12.0.1, Verhaltensänderung,94
- UNLOAD-Anweisung
 - Version 11.0.0, Erweiterung,239
 - Version 16.0, Erweiterung,24
- unload-Dienstprogramm (dbunload)
 - Fehlerbehandlung, Upgrades auf Version 16,306
- UNNEST-Operator
 - Version 16.0, neue Funktion,9
- Unterstützte Plattformen
 - Version 11.0.0, Verhaltensänderung,257
- Unterstützung für konsolidierte Datenbanken
 - Version 12.0.0, neue Funktion, MobiLink,159
- up_pad_size
 - Relay Server,54
- UPDATE-Anweisung
 - Version 11.0.0, Erweiterung,239,247
 - Version 11.0.0, Verhaltensänderung,257
 - Version 11.0.1, Erweiterung,204
 - Version 12.0.0, Verhaltensänderung,149
 - Version 12.0.1, Erweiterung,88
- Upgrade
 - Datenbanken der Version 10 und höher mit ALTER DATABASE,313
 - Datenbanken der Version 10 und höher mit dbupgrad,312
 - Datenbanken der Version 10 und höher mit Sybase Central,311
 - Datenbanken mit materialisierten Ansichten,296
 - Datenbankspiegelung,314
 - Fehlerbehandlung,322
 - Info zum Upgrade auf Version 16,289
 - konsolidierte MobiLink-Datenbanken,326
 - MobiLink,325
 - MobiLink, entfernte Datenbanken,335
 - MobiLink-Server,334
 - MobiLink-Systemobjekte,326
 - Monitor,342
 - Neuerstellung Datenbanken für Version 16,303,310
 - SQL Anywhere,295
 - SQL Remote,341
 - Synchronisationsskripten,335
 - UltraLite, Überblick für Datenbanken und Anwendungen,337
 - Vorsichtsmaßnahmen,297
- Upgrade einer Datenbank
 - aus Sybase Central,311
- Upgrade einer Datenbank, Assistent
 - verwenden,311
- Upgrade von Adaptive Server Enterprise, Oracle, MySQL oder Microsoft SQL Server
 - MobiLink, konsolidierte Datenbanken,328
- Upgrade von Datenbanken
 - Einschränkungen,304
- Upgrade von konsolidierten SQL Anywhere-Datenbanken vor Version 10.0.0
 - MobiLink, konsolidierte Datenbanken,331
- Upgrade von SQL Anywhere
 - Info,289
- Upgrade von SQL Anywhere 10.0.0 und höher
 - MobiLink, konsolidierte Datenbanken,327
- Upgrade-Dienstprogramm (dbupgrad)
 - Version 11.0.1, Verhaltensänderung,209
- Upgrades
 - Version 16.0, Erweiterung,13
- Upload-Zeilenzahl unter Android
 - Version 16.0, Erweiterung, UltraLiteJ,63
- upload_update
 - Konflikterkennung, Version 12.0.0, nicht mehr empfohlen,172
- user_estimates-Option
 - Version 11.0.0, Erweiterung,239
- USER_NAME-Funktion
 - Version 16.0, Verhaltensänderung,35
- UserDefinedCounterRate01, Eigenschaft
 - Version 12.0.1, neue Funktion,91
- UserDefinedCounterRate02, Eigenschaft

- Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRate03, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRate04, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRate05, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRaw01, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRaw02, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRaw03, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRaw04, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - UserDefinedCounterRaw05, Eigenschaft
 - Version 12.0.1, neue Funktion,91
 - username-Systemparameter
 - Version 12.0.0, neue Funktion, MobiLink,161
 - UTC TIMESTAMP-Spezialwert
 - Version 12.0.0, Verhaltensänderungen,149
 - UTCTimestampCatalog-Eigenschaft
 - Version 16.0, neue Funktion,19
 - UTF-16-Kodierung
 - Version 11.0.0, Erweiterung, CSCONVERT-Funktion,246
 - Version 11.0.0, Erweiterung, LOAD TABLE-Anweisung,246
 - Version 11.0.0, Erweiterung, UNLOAD-Anweisung,246
 - UTF-8
 - Version 11.0.0, Verhaltensänderung, Unterstützung in URLs,241
 - UUID.toString-Methode [UltraLite für M-Business Anywhere]
 - Version 12.0.1, neue Funktion,83
 - uuid_has_hyphens, Eigenschaft
 - Version 12.0.0, wieder eingesetzte Funktion,130
 - uuid_has_hyphens, Option
 - Version 12.0.0, wieder eingesetzte Funktion,130
 - uuid_has_hyphens-Option
 - Version 11.0.0, nicht unterstützt,260
- ## V
- V_\$TRANSACTION, Oracle-Systemansicht
 - Version 12.0.0, Verhaltensänderung, MobiLink,167
 - VALIDATE DATABASE-Anweisung
 - Version 12.0.0, Verhaltensänderung,152
 - VALIDATE LDAP SERVER-Anweisung
 - Version 16.0, neue Funktion,6
 - VALIDATE TEXT INDEX-Anweisung
 - Version 12.0.1, neue Funktion,88
 - VALIDATE-Anweisung
 - Version 12.0.1, Erweiterung,88
 - VALIDATE-Berechtigung
 - Version 16.0, nicht mehr empfohlen,37
 - Validierung
 - Version 11.0.0, Verhaltensänderung,227
 - Version 12.0.0, Erweiterung,146
 - Version 12.0.0, Verhaltensänderung,152
 - Validierungs-Dienstprogramm (dbvalid)
 - Version 11.0.0, Erweiterung,227
 - Validierungsdienstprogramm (dbvalid)
 - Version 12.0.0, Verhaltensänderung,152
 - VARCHAR, Datentyp
 - Version 12.0.0, Erweiterung,140
 - varray (Oracle)
 - Version 11.0.1, neue Funktion,211
 - Varray, Unterstützung
 - Version 12.0.0, Erweiterung,159
 - Verbinden, Fenster
 - Version 11.0.0, Erweiterung,277
 - Version 11.0.0, neue Funktion,277
 - Version 12.0.0, Erweiterung,186
 - Version 12.0.0, neue Funktion,112
 - Version 12.0.1, Erweiterung, OnDemand,73
 - Version 16.0, Verhaltensänderung,67
 - Verbindungen
 - Version 11.0.0, Verhaltensänderung,254
 - Verbindungsassistent
 - in Version 12.0.0 entfernt,186
 - Version 11.0.0, neue Funktion,277
 - Verbindungseigenschaften
 - Version 11.0.0, neue Funktion,230
 - Version 12.0.0, neue Funktion,132
 - Version 12.0.1, Erweiterung,76
 - Version 12.0.1, neue Funktion,91
 - Version 16.0, Erweiterungen,19
 - Verbindungspooling
 - Version 12.0.0, Erweiterung,141
 - Version 12.0.0, neue Funktion,128
 - Verbindungsstatus unter Android
 - Version 16.0, Erweiterung, UltraLiteJ,63
 - Verfügbarkeits-Timeout

- Version 16.0, Erweiterung, MobiLink (mlreplay),44
- Verhaltensänderungen
 - Version 11.0.0,219
 - Version 11.0.1,199
 - Version 12.0.0,117
 - Version 12.0.1,73
 - Version 16.0,1
 - Version 9.0.2 und früher,vii
- verify_password_function-Option
 - Version 16.0, Erweiterung,13
- Verschleiern von Dateien, Dienstprogramm, (dbfhide)
 - Version 12.0.0, Erweiterung,129
- Verschlüsselung
 - Version 11.0.0, Erweiterung,225,245
 - Version 12.0.0, Erweiterung,135
 - Version 16.0, Erweiterung,26
 - Version 16.0, neue Funktion, UltraLite,59
- Verschlüsselungssteuerung
 - Version 16.0, entfernte API, UltraLite,65
- Version 11.0.0
 - neue Funktionen,219
 - nicht mehr empfohlene Funktionen,219
 - Verhaltensänderungen,219
- Version 11.0.0, Erweiterung
 - LEN-Funktion,246
 - LENGTH-Funktion,246
- Version 11.0.1
 - neue Funktionen,199
 - nicht mehr empfohlene Funktionen,199
 - Verhaltensänderungen,199
- Version 12.0.0
 - Interactive SQL, neue Funktionen,186
 - MobiLink, neue Funktionen,156
 - neue Funktionen,117
 - SQL Anywhere, neue Funktionen,119
 - Sybase Central, neue Funktionen,186
 - UltraLite, neue Funktionen,173
 - Verhaltensänderungen,117
- Version 12.0.0, Verhaltensänderung
 - Index-Hints,154
- Version 12.0.1
 - Erweiterung, UltraLite,106
 - Interactive SQL, neue Funktionen,111
 - MobiLink, neue Funktionen,96
 - neue Funktionen,73
 - nicht mehr empfohlene Funktionen, MobiLink,103
 - nicht mehr weiterentwickelte Funktionen, MobiLink,103
 - SQL Anywhere, neue Funktionen,84
 - Sybase Central, neue Funktionen,111
 - UltraLite, neue Funktionen,104
 - Verhaltensänderungen,73
 - Verhaltensänderungen, MobiLink,101
 - Verhaltensänderungen, UltraLite ,110
- Version 12.0.1, Erweiterung
 - UltraLiteJ,107
- Version 12.0.1, neue Funktionen
 - MobiLink-Clients,100
 - Relay Server,101
- Version 12.0.1, Verhaltensänderungen
 - MobiLink-Client,102
 - MobiLink-Plug-In,103
- Version 16.0
 - bekannte Upgrade-Probleme,306
 - entfernte APIs, UltraLite ,67
 - entfernte Funktionen, UltraLite,65
 - entfernte Plattformen, UltraLite,66
 - Interactive SQL, neue Funktionen,69
 - MobiLink, neue Funktionen,40
 - neue Funktionen,1
 - Relay Server, neue Funktionen,53
 - Relay Server, Verhaltensänderungen,55
 - SQL Anywhere, neue Funktionen,9
 - Sybase Central, neue Funktionen,68
 - UltraLite, neue Funktionen,56
 - Upgrade,289
 - Verhaltensänderungen,1
 - Verhaltensänderungen, UltraLite ,65
- Version 16.0, Erweiterung
 - UltraLite C/C++,60
 - UltraLite.NET,61
 - UltraLiteJ,61
- Version 5
 - Upgrade von SQL Remote-Installationen,341
- Versionen
 - Upgrade von Synchronisationsskripten,335
- Versionen 12.0.0
 - nicht mehr weiterentwickelte Funktionen,117
- Versionen 12.0.1
 - nicht mehr weiterentwickelte Funktionen,73
- Versionen 16.0
 - nicht mehr empfohlene Funktionen,1
- Vertrauenswürdige Zertifikate
 - Version 12.0.1, Erweiterung,79

viewcert, Dienstprogramm
 Version 16.0, Erweiterung,17
 VIM, Nachrichtentyp
 Version 11.0.0, nicht unterstützt,269
 Vista
 Version 11.0.0, Erweiterung,224
 Visual Studio
 Version 11.0.0, Verhaltensänderung,288
 Visual Studio .NET
 Version 11.0.0, Verhaltensänderung,288
 Visual Studio 2005-Projektdateien
 Version 16.0 entfernte Funktion, UltraLite ,65
 Visual Studio 2012-Unterstützung
 Version 16.0, neue Funktion,24
 Volltextsuche
 Version 11.0.0, neue Funktion,220
 Version 11.0.1, Verhaltensänderung,206
 Version 16.0, Erweiterung,27
 Vorsichtsmaßnahmen
 Upgrade,297
 VSS
 Version 11.0.0, neue Funktion,224

W

WaitStartTime-Eigenschaft
 Version 11.0.1, neue Funktion,
 Verbindungseigenschaft,199
 WaitType-Eigenschaft
 Version 11.0.1, neue Funktion,
 Verbindungseigenschaft,199
 Warnmeldungen unter Android
 Version 16.0, Erweiterung, UltraLiteJ,63
 Wartungspläne
 Version 12.0.0, Erweiterung,190
 WebClientLogFile-Eigenschaft
 Version 11.0.0, neue Funktion,244
 WebClientLogging-Eigenschaft
 Version 11.0.0, neue Funktion,244
 Webdienstclients
 Version 11.0.1, Verhaltensänderung,210
 Webdienste
 Version 11.0.0, Verhaltensänderung,254
 Version 12.0.0, Erweiterung,141
 Version 12.0.1, Erweiterung,89
 Webserver
 Version 11.0.0, Verhaltensänderung,254
 webservice_sessionid_name-Eigenschaft

Version 16.0, neue Funktion,19
 webservice_sessionid_name-Option
 Version 16.0, neue Funktion,18
 win32, Verzeichnis
 Version 11.0.0, Verhaltensänderung,288
 Windows
 Version 11.0.0, Verhaltensänderung,257
 Windows 2000
 Version 12.0.0, nicht unterstützt,155
 Windows CE nun Windows Mobile
 Version 11.0.0, Verhaltensänderung,287
 Windows IIS-Unterstützung
 Version 12.0.0, Erweiterung, Relay Server,166
 Windows Mobile
 Version 11.0.0, Erweiterung,245
 Version 12.0.0, Erweiterung,145
 Windows Vista
 Version 11.0.0, Erweiterung,224
 Windows, Dienste
 Version 12.0.0, Verhaltensänderung,118
 Windows-Systemmonitor
 Version 11.0.0, Erweiterung,229
 WITH CONTENT LOGGING-Klausel
 Version 11.0.0, neue Funktion,235
 WITH FILE NAME LOGGING-Klausel
 Version 11.0.0, neue Funktion,236
 WITH NULLS NOT DISTINCT-Klausel
 Version 12.0.0, neue Funktion,136,188
 WITH ROW LOGGING-Klausel
 Version 11.0.0, neue Funktion,236
 WITH SHARE MODE-Klausel, REFRESH
 MATERIALIZED VIEW-Anweisung
 Version 11.0.0, neue Funktion,223
 Word-Dokumente, vollständige Textindizierung
 Version 12.0.0, neue Funktion,126
 Worker-Threads
 Version 12.0.0, Verhaltensänderung,148
 Write-at-end-beständige Speicher
 Version 16.0, entfernte Funktion, UltraLite,66
 WriteChecksums-Eigenschaft
 Version 12.0.0, neue Funktion,132
 WRITECLIENTFILE-Berechtigung
 Version 16.0, nicht mehr empfohlen,37
 WriteNoPK, Sperren
 Version 12.0.0, neue Funktion,145

X

X64, Verzeichnis

Version 11.0.0, Verhaltensänderung,288

Xcode 4.2-Beispiele

Version 12.0.1, neue Funktion,83

XMLTYPE-Datentyp

Version 16.0, neue Funktion,40

Version 16.0, neue Funktion, MobiLink-Plug-In,49

xp_cmdshell-Systemprozedur

Version 16.0, Verhaltensänderung,28

xp_get_mail_error_code-Systemprozedur

Version 16.0, neue Funktion,21

xp_get_mail_error_text-Systemprozedur

Version 16.0, neue Funktion,21

xp_getenv-Systemprozedur

Version 16.0, neue Funktion,21

Version 16.0, Verhaltensänderung,28

xp_move_file-Systemprozedur

Version 16.0, Verhaltensänderung,28

xp_read_file-Systemprozedur

Version 12.0.0, Erweiterung,134

Version 16.0, Verhaltensänderung,28

xp_sendmail-Systemprozedur

Version 16.0, Verhaltensänderung,28,34

xp_startmail-Systemprozedur

Version 16.0, Verhaltensänderung,28,32,34

xp_startsmtp-Systemprozedur

Version 12.0.0, Erweiterung,134

Version 16.0, Erweiterung,21

Version 16.0, Verhaltensänderung,28,32,34

xp_stopmail-Systemprozedur

Version 16.0, Verhaltensänderung,28,34

xp_stopsmtpt-Systemprozedur

Version 16.0, Verhaltensänderung,28,34

xp_write_file-Systemprozedur

Version 16.0, Verhaltensänderung,28

Version 16.0, Erweiterung, UltraLiteJ,61

Zeilenbeschränkung

Version 12.0.1, Erweiterung,82

Zeilennummern

Version 11.0.0, Erweiterung, Interactive SQL,282

Zeitstempel

Version 12.0.0, Erweiterung,131

Zentrale Administration von entfernten Datenbanken

Version 12.0.0, Erweiterung, Relay Server,166

Version 12.0.0, neue Funktion, MobiLink,156

Version 12.0.1, Erweiterung, MobiLink,98

Version 12.0.1, neue Funktion, MobiLink,100

Zertifikate

Version 11.0.0, Verhaltensänderung,288

Version 16.0, Verhaltensänderung,7

ZLIB-Komprimierung

Version 16.0, neue Funktion, UltraLite,57

Zugriffspläne

Version 12.0.1, Erweiterung,77

Zusammengesetzte Datentypen

Version 16.0, neue Funktion,9

Zustand

in Version 12.0.0 nicht mehr weiterentwickelt,196

Z

Zeichen, Datentypen

Version 12.0.0, Erweiterung,140

Zeilen-Caching

Version 16.0, Erweiterung, UltraLite,60

Zeilen-Traversal

Version 11.0.0, Erweiterung, UltraLite,274

Zeilenanzahl

Version 12.0.0, Erweiterung,143

Zeilenberichte