

## **Der ORACLE-Treiber**

### **Für alle Benutzer**

Die folgenden Themen beschreiben den ORACLE-Treiber und erklären, wie der Treiber installiert werden muß, um von einem Anwendungsprogramm verwendet werden zu können.

[Überblick](#)

[Die Hardware- und Software-Anforderungen](#)

[Einrichten des ORACLE-Treibers](#)

[Hinzufügen, Bearbeiten und Löschen von ORACLE-Datenquellen](#)

[Herstellen einer Verbindung zu einer ORACLE-Datenquelle](#)

[Fehlerbehebung](#)

### **Für fortgeschrittene Benutzer**

Diese Themen erläutern, wie der ORACLE-Treiber direkt verwendet werden kann.

[Verbindungszeichenfolgen \(fortgeschritten\)](#)

[SQL-Anweisungen \(fortgeschritten\)](#)

[Datentypen \(fortgeschritten\)](#)

[Fehlermeldungen \(fortgeschritten\)](#)

### **Für Programmierer**

Die folgenden Themen beschreiben, wie der ORACLE-Treiber über Code verwendet werden kann. Diese Themen sind für Programmierer von Anwendungsprogrammen vorgesehen und erfordern die Kenntnis der zu ODBC-(Open Database Connectivity) gehörenden Schnittstelle für die Anwendungsprogrammierung (Application Programming Interface - [API](#)).

[Rückgabewerte von SQLGetInfo \(Programmierung\)](#)

[ODBC-API-Funktionen \(Programmierung\)](#)

[Probleme aufgrund der Realisierung \(Programmierung\)](#)

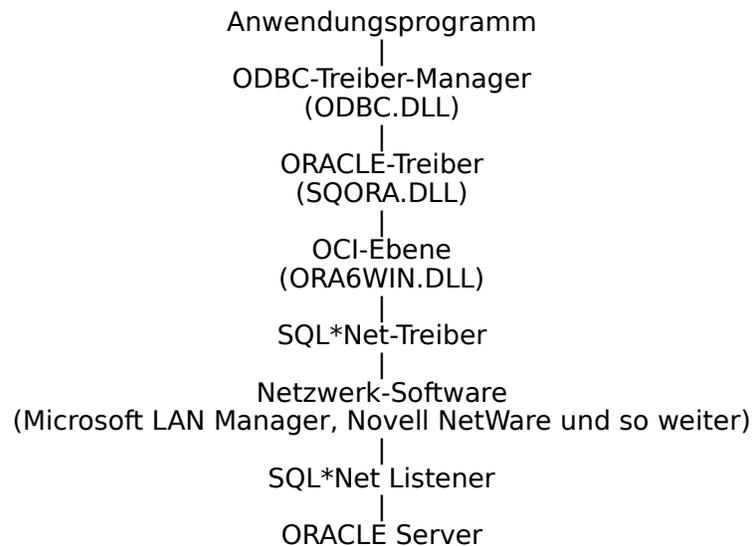
## Überblick

Siehe auch

ORACLE Server ist ein Mehrbenutzer-RDBMS (relationales Datenbank-Managementsystem, DBMS), das unter einer Vielzahl von Betriebssystemen, die für Arbeitsstationen oder Minicomputer entwickelt wurden, ausgeführt werden kann. Der Zugriff auf Daten erfolgt im ORACLE Server mit Hilfe der Strukturierten Abfragesprache (Structured Query Language, SQL). IBM-PCs sowie IBM-kompatible PCs, auf denen Microsoft Windows ausgeführt wird, kommunizieren mit ORACLE-Datenbankservern über ein Netzwerk, wie zum Beispiel den Microsoft LAN Manager, Novell NetWare, Banyan VINES, DECnet oder ein beliebiges TCP/IP-Netzwerk.

Mit Hilfe des ORACLE-Treibers können Anwendungsprogramme über die Schnittstelle ODBC (Open Database Connectivity) auf die in einer ORACLE-Datenbank abgelegten Daten zugreifen. Der Treiber greift auf SQL\*Net zurück, um mit dem jeweiligen Netzwerk zu kommunizieren.

Die Anwendungsprogramm/Treiber-Architektur sieht wie folgt aus:



**Siehe auch**

Für alle Benutzer

[Hinzufügen, Bearbeiten und Löschen von ORACLE-Datenquellen](#)

[Herstellen einer Verbindung zu einer ORACLE-Datenquelle](#)

[Die Hardware- und Software-Anforderungen](#)

[Einrichten des ORACLE-Treibers](#)

## Die Hardware- und Software-Anforderungen

Siehe auch

Um auf ORACLE-Daten zugreifen zu können, müssen Ihnen folgende Komponenten zur Verfügung stehen:

- der ORACLE-Treiber;
- eine ORACLE Server-Datenbank;
- ein Netzwerk, über das die Rechner miteinander verbunden sind, auf denen sich die benötigten Dateien befinden, sowie eine über das Netzwerk per SQL\*Net hergestellte Verbindung.

In den folgenden Abschnitten wird beschrieben, welche Hardware und Software diese Komponenten jeweils erfordern.

### Der ORACLE-Treiber

Der ORACLE-Treiber erfordert die folgende Hardware:

- einen ISA-Rechner (Industry Standard Architecture), zum Beispiel einen IBM PC/AT oder einen dazu kompatiblen Rechner, oder
- einen MCA-Rechner (Micro Channel Architecture), zum Beispiel einen IBM PS/2 oder einen dazu kompatiblen Rechner, oder
- einen EISA-Rechner (Extended Industry Standard Architecture) mit einem der Mikroprozessoren 80286, 80386 oder 80486;
- einen Arbeitsspeicher (RAM) von mindestens 2 MB; 4 MB RAM werden empfohlen;
- eine Festplatte, auf der für den ORACLE-Treiber noch ungefähr 350 KB frei sind.

Der ORACLE-Treiber erfordert die folgende Software:

- MS-DOS ab Version 3.3;
- Microsoft Windows ab Version 3.0;
- den ODBC-Treiber-Manager, Version 1.0, (ODBC.DLL);
- ORA6WIN.DLL, eine DLL-Datei (Dynamic-Link Library), die ORACLE Call Interface (OCI)-Dienste zur Verfügung stellt.

---

**Anmerkung** Der ORACLE-Treiber erfordert ORA6WIN.DLL unabhängig davon, ob eine spätere Version des OCI installiert wurde. ORA6WIN.DLL wird zusammen mit dem ORACLE-Treiber geliefert.

---

### ORACLE Server

Sie können nur dann mit dem ORACLE-Treiber auf Daten einer ORACLE-Datenbank zugreifen, wenn ORACLE Server in der Version 6.0.34 oder höher vorliegt. Da der ORACLE-Treiber für ORACLE Server, Version 6, entwickelt wurde, ist es möglich, daß Sie beim Zugriff auf Daten über ORACLE Server, Version 7, auf kleinere Unstimmigkeiten stoßen. Informationen darüber, welche Hardware und Software ORACLE Server erfordert, finden Sie in der ORACLE-Dokumentation.

### Die Netzwerk-Software

Es wird ein Netzwerk benötigt, um die Plattformen miteinander zu verbinden, auf denen sich ORACLE Server sowie der ORACLE-Treiber befinden. Informationen zur Hardware und Software, die das jeweilige Netzwerk erfordert, finden Sie in der Netzwerk-Dokumentation.

---

**Anmerkung** Wenn Sie den Microsoft LAN Manager einsetzen und mit dem Named Pipes-Protokoll arbeiten, muß der LAN Manager in der Version 2.0 oder höher vorliegen. Wenn Sie den Microsoft LAN Manager einsetzen und mit dem TCP/IP-Protokoll arbeiten, muß er in der Version 2.2 oder höher vorliegen.

---

### SQL\*Net

Der ORACLE-Treiber und ORACLE Server setzen SQL\*Net als gemeinsame Netzwerkschnittstelle ein; Sie sollten den ORACLE-Treiber mit jeder SQL\*Net-Verbindung zwischen einem Computer, auf dem Windows ausgeführt wird, und einem ORACLE-Datenbank-Server einsetzen können. Der ORACLE-Treiber arbeitet nur im Zusammenhang

mit SQL\*Net-Dynamic-Link-Libraries (DLL-Dateien).

Ausführliche Informationen zu SQL\*Net finden Sie in der SQL\*Net-Dokumentation.

---

**Anmerkung** SQL\*Net-Komponenten sind nur von der Oracle Corporation erhältlich.

---

**Siehe auch**

Für alle Benutzer

[Einrichten des ORACLE-Treibers](#)

**SQL\*Net**

SQL\*Net ist ein Netzwerk-Schnittstellenpaket von der Oracle Corporation, mit dessen Hilfe Anwendungsprogramme über eine Vielzahl von Netzwerken auf Daten zugreifen können, die mit ORACLE Server verwaltet werden.

## Einrichten des ORACLE-Treibers

Siehe auch

### So richten Sie den ORACLE-Treiber ein:

- 1 Sofern Sie dies noch nicht erledigt haben, müssen Sie die SQL\*Net for Windows-DLL-Dateien auf Ihrem Rechner installieren. Wenn Sie Ihre SQL\*Net-Verbindung testen möchten, sollten Sie Ihren Rechner als ORACLE-Client-Arbeitsstation einrichten und SQL\*DBA oder SQL\*Plus ausführen. Informationen hierzu finden Sie in der SQL\*Net-Dokumentation.

---

**Wichtig** Der ORACLE-Treiber ist für die Verwendung mit ORACLE Server, Version 6 vorgesehen; SQL\*Net for Windows-DLL-Dateien sind vorgesehen für ORACLE Server, Version 7. Weitere Informationen über das Konfigurieren Ihres Systems zur Verwendung der SQL\*Net for Windows-DLL-Dateien mit dem ORACLE-Treiber finden Sie in der Datei ORACLE.TXT, die mit dem ORACLE-Treiber geliefert wird.

---

- 2 Erstellen Sie für jeden ORACLE Server, über den Sie auf Daten zugreifen möchten, eine Datenquelle.

### So installieren Sie eine neue Version des ORACLE-Treibers:

- 1 Doppelklicken Sie in der Hauptgruppe des Programm-Managers auf das Symbol "Systemsteuerung". Doppelklicken Sie innerhalb des Fensters "Systemsteuerung" auf das Symbol "ODBC".

---

**Anmerkung** Wenn Sie mit Microsoft Windows, Version 3.0, arbeiten, starten Sie den ODBC-Administrator, indem Sie auf das zur Gruppe "Microsoft ODBC" gehörende Symbol "Microsoft ODBC-Administrator" doppelklicken.

---

Es wird das Dialogfeld **Datenquellen** angezeigt.

- 2 Wählen Sie aus dem Dialogfeld **Datenquellen** die Schaltfläche "Treiber".  
Es wird das Dialogfeld **Treiber** angezeigt.
- 3 Wählen Sie aus dem Dialogfeld **Treiber** die Schaltfläche "Hinzufügen".  
Es wird das Dialogfeld **Treiber hinzufügen** angezeigt.
- 4 Geben Sie den Pfad (Laufwerk und Verzeichnis) des ORACLE-Treibers in das Textfeld ein. Sie können auch Sie auf die Schaltfläche "Durchsuchen" klicken und dann einen Pfad auswählen.
- 5 Wählen Sie "OK" im Dialogfeld **Treiber hinzufügen**.  
Es wird das Dialogfeld **Treiber installieren** angezeigt.
- 6 Markieren Sie in der Liste "Verfügbare ODBC-Treiber" den Eintrag "ORACLE".
- 7 Wählen Sie "OK".  
Der ORACLE-Treiber wird installiert.

### So löschen Sie den ORACLE-Treiber:

- 1 Doppelklicken Sie in der Hauptgruppe des Programm-Managers auf das Symbol "Systemsteuerung". Doppelklicken Sie innerhalb des Fensters "Systemsteuerung" auf das Symbol "ODBC".

---

**Anmerkung** Wenn Sie mit Microsoft Windows, Version 3.0, arbeiten, starten Sie den ODBC-Administrator, indem Sie auf das zur Gruppe "Microsoft ODBC" gehörende Symbol "Microsoft ODBC-Administrator" doppelklicken.

---

Es wird das Dialogfeld **Datenquellen** angezeigt.

- 2 Wählen Sie aus dem Dialogfeld **Datenquellen** die Schaltfläche "Treiber".  
Es wird das Dialogfeld **Treiber** angezeigt.
- 3 Markieren Sie in der Liste "Installierte ODBC-Treiber" den Eintrag "ORACLE".
- 4 Wählen Sie die Schaltfläche "Löschen".

Eine Meldung fordert Sie dazu auf, das Löschen des Treibers sowie aller Datenquellen, denen der Treiber zugeordnet ist, zu bestätigen.

5 Wählen Sie die Schaltfläche "Ja".

**Siehe auch**

Für alle Benutzer

[Hinzufügen, Bearbeiten und Löschen von ORACLE-Datenquellen](#)

[Die Hardware- und Software-Anforderungen](#)

### **Datenquelle (ORACLE)**

Eine Datenquelle umfaßt sowohl die Daten, auf die ein Benutzer zugreifen möchte, als auch die Informationen, die benötigt werden, um die Daten abzurufen. Für den ORACLE-Treiber ist eine Datenquelle eine bestimmte Kopie von ORACLE Server, der Rechner, auf dem sich ORACLE befindet, das Netzwerk, über das auf den Rechner zugegriffen werden kann, sowie die SQL\*Net-Komponenten, die als Schnittstelle zu dem Netzwerk dienen.

## Hinzufügen, Bearbeiten und Löschen von ORACLE-Datenquellen

Siehe auch

Bevor Sie mit dem ORACLE-Treiber auf Daten zugreifen können, müssen Sie für jede Ihrer Kopien von ORACLE Server eine Datenquelle hinzufügen. Der ORACLE-Treiber verwendet die von Ihnen beim Hinzufügen der Datenquelle eingegebenen Informationen, um auf die Daten zuzugreifen. Sie können eine Datenquelle jederzeit ändern oder löschen.

### So fügen Sie eine ORACLE-Datenquelle hinzu:

- 1 Doppelklicken Sie in der Hauptgruppe des Programm-Managers auf das Symbol "Systemsteuerung". Doppelklicken Sie innerhalb des Fensters "Systemsteuerung" auf das Symbol "ODBC".

---

**Anmerkung** Wenn Sie mit Microsoft Windows, Version 3.0, arbeiten, starten Sie den ODBC-Administrator, indem Sie auf das zur Gruppe "Microsoft ODBC" gehörende Symbol "Microsoft ODBC-Administrator" doppelklicken.

---

- 2 Wählen Sie aus dem Dialogfeld **Datenquellen** die Schaltfläche "Hinzufügen"  
Es wird das Dialogfeld **Datenquelle hinzufügen** angezeigt.
- 3 Markieren Sie innerhalb der Liste "Installierte ODBC-Treiber" den Eintrag "ORACLE" und wählen Sie die Schaltfläche "OK".  
Es wird das Dialogfeld **ODBC-Oracle-Setup** angezeigt.
- 4 Wählen Sie im Dialogfeld ODBC-Oracle-Setup die gewünschten Optionen aus, und wählen Sie "OK".

### So bearbeiten Sie eine ORACLE-Datenquelle:

- 1 Doppelklicken Sie in der Hauptgruppe des Programm-Managers auf das Symbol "Systemsteuerung". Doppelklicken Sie innerhalb des Fensters "Systemsteuerung" auf das Symbol "ODBC".

---

**Anmerkung** Wenn Sie mit Microsoft Windows, Version 3.0, arbeiten, starten Sie den ODBC-Administrator, indem Sie auf das zur Gruppe "Microsoft ODBC" gehörende Symbol "Microsoft ODBC-Administrator" doppelklicken.

---

- 2 Markieren Sie im Dialogfeld **Datenquellen** in der Liste "Datenquellen" die zu bearbeitende Datenquelle und wählen Sie die Schaltfläche "Setup".  
Es wird das Dialogfeld **ODBC-Oracle-Setup** angezeigt.
- 3 Wählen Sie im Dialogfeld ODBC-Oracle-Treiber die gewünschten Optionen aus, und wählen Sie "OK".

### So löschen Sie eine ORACLE-Datenquelle:

- 1 Doppelklicken Sie in der Hauptgruppe des Programm-Managers auf das Symbol "Systemsteuerung". Doppelklicken Sie innerhalb des Fensters "Systemsteuerung" auf das Symbol "ODBC".

---

**Anmerkung** Wenn Sie mit Microsoft Windows, Version 3.0, arbeiten, starten Sie den ODBC-Administrator, indem Sie auf das zur Gruppe "Microsoft ODBC" gehörende Symbol "Microsoft ODBC-Administrator" doppelklicken.

---

- 2 Markieren Sie im Dialogfeld **Datenquellen** in der Liste "Datenquellen" diejenige Datenquelle, die Sie löschen möchten.
- 3 Wählen Sie die Schaltfläche "Löschen", und wählen Sie anschließend die Schaltfläche "Ja", um Ihre Löschabsicht zu bestätigen.

**Siehe auch**

Für alle Benutzer

[Herstellen einer Verbindung zu einer ORACLE-Datenquelle](#)

[Einrichten des ORACLE-Treibers](#)

## **Herstellen einer Verbindung zu einer ORACLE-Datenquelle**

Siehe auch

Der ORACLE-Treiber kann nur dann eine Verbindung zu einer Datenquelle herstellen, wenn auf Ihrem Rechner ein SQL\*Net-Treiber geladen ist und auf dem ORACLE-Datenbank-Server der entsprechende SQL\*Net Listener ausgeführt wird. Weitere Informationen zu SQL\*Net finden Sie in der SQL\*Net-Dokumentation.

Bei der Herstellung einer Verbindung ist es möglich, daß ein Anwendungsprogramm Sie zur Eingabe bestimmter Informationen auffordert. Wenn ein Anwendungsprogramm Sie dazu auffordert, Informationen zu einer ORACLE-Datenquelle einzugeben, sollten Sie folgende Schritte ausführen:

### **So stellen Sie eine Verbindung zu einer ORACLE-Datenquelle her**

- 1 Geben Sie in das Feld "Benutzername" den Namen ein, den Sie in ORACLE Server verwenden.
- 2 Geben Sie in das Feld "Kennwort" das Kennwort ein, das Sie in ORACLE Server verwenden.
- 3 Wählen Sie "OK".

**Siehe auch**

Für alle Benutzer

[Hinzufügen, Bearbeiten und Löschen von ORACLE-Datenquellen](#)

Für fortgeschrittene Benutzer

[Verbindungszeichenfolgen \(fortgeschritten\)](#)

Für Programmierer

[Die Realisierung von SQLDriverConnect \(Programmierung\)](#)

## Fehlerbehebung

In den folgenden Abschnitten wird erläutert, wie Sie Fehler beheben können, die beim Einsatz des ORACLE-Treibers eventuell auftreten.

### **Ein Anwendungsprogramm funktioniert nicht, wenn speicherresidente SQL\*Net-Treiber eingesetzt werden**

Der ORACLE-Treiber arbeitet nur in Zusammenhang mit SQL\*Net-Dynamic-Link-Libraries (DLL-Dateien).

Ausführliche Informationen zu SQL\*Net finden Sie in der SQL\*Net-Dokumentation.

---

**Anmerkung** SQL\*Net-Komponenten sind nur von der Oracle Corporation erhältlich.

---

### **"Message send failure"-Fehler, wenn Microsoft LAN Manager mit TCP/IP arbeitet**

Wenn Microsoft LAN Manager mit dem Protokoll TCP/IP arbeitet, gibt das in dem Abschnitt [sockets] der Datei TCPUTILS.INI stehende Schlüsselwort MAXSENDSIZE an, wie viele Datenbytes auf einmal an ORACLE Server gesendet oder von dort abgerufen werden können. Tritt ein "Message send failure"-Fehler auf, sollten Sie diesen Wert vergrößern. Dieser Fehler tritt meist dann auf, wenn SQL\_LONGVARCHAR- oder SQL\_LONGVARIABLE-Daten gesendet werden. Weitere Informationen finden Sie in der ORACLE-Dokumentation.

## Das Dialogfeld "ODBC-Oracle-Setup"

Das Dialogfeld **ODBC-Oracle-Setup** enthält die folgenden Optionen.

### Datenquellename

Ein Name, mit dem Sie die Datenquelle kennzeichnen. Beispiel: "Personaldaten".

### Beschreibung

Eine Beschreibung der in der Datenquelle abgelegten Daten. Beispiel: "Einstellungsdatum, Gehaltsentwicklung und aktuelle Beurteilung aller Mitarbeiter".

### SQL\*Net-Verbindungszeichenfolge

Eine SQL\*Net-Verbindungszeichenfolge, die angibt, wo sich ORACLE Server befindet, aus dem der Treiber Daten abrufen soll. Eine ORACLE-Verbindungszeichenfolge hat folgendes Format:

*Netz\_Präfix:Hostname[:SID]*

Die zu diesem Format gehörenden Argumente haben folgende Bedeutungen:

Argument	Bedeutung
<i>Netz_Präfix</i>	Gibt an, welcher SQL*Net-Treiber eingesetzt werden soll. Einige gängige <i>Netz_Präfixe</i> sind: P: Named pipes T: TCP/IP X: SPX/IPX
<i>Hostname</i>	Der Name oder Alias-Name des Servers, auf dem sich ORACLE Server befindet.
<i>SID</i>	Die Systemkennung (System-ID) der Datenbank, die sich auf dem Server befindet und mit der Sie arbeiten möchten (optional).

Weitere Informationen finden Sie in Ihrer SQL\*Net-Dokumentation.

Wenn Sie die folgenden Felder ausfüllen möchten, klicken Sie auf die Schaltfläche "Optionen".

### Umwandlung

Es wird die Beschreibung des aktuellen Umwandlungsprogramms angezeigt. Wenn Sie ein anderes Umwandlungsprogramm einsetzen möchten, wählen Sie die Schaltfläche "Auswählen". Sie können dann ein anderes Umwandlungsprogramm aus der Liste im Dialogfeld **Umwandler auswählen** auswählen.

### OEM- in ANSI-Zeichen umwandeln

Wenn der ORACLE-Treiber und ORACLE Server denselben Nicht-ANSI-Zeichensatz verwenden, aktivieren Sie das Kontrollkästchen "OEM-Zeichen in ANSI-Zeichen umwandeln".

Wenn der ORACLE-Treiber und ORACLE Server mit unterschiedlichen Zeichensätzen arbeiten, müssen Sie ein für die jeweiligen Zeichensätze geeignetes Umwandlungsprogramm angeben.

## Verbindungszeichenfolgen (fortgeschritten)

Siehe auch

In einer für den ORACLE-Treiber verwendeten Verbindungszeichenfolge werden die folgenden Schlüsselwörter eingesetzt:

<b>Schlüssel wort</b>	<b>Erläuterung</b>
<b>DSN</b>	Akronym für Data Source Name; der Name der Datenquelle.
<b>DBQ</b>	Eine ORACLE-Verbindungszeichenfolge der Form: <i>Netz_Präfix:Hostname[:SID]</i> <i>Netz_Präfix</i> gibt an, welcher SQL*Net-Treiber verwendet werden soll; <i>Hostname</i> gibt den Namen oder Alias-Namen des Servers an, auf dem sich ORACLE Server befindet; <i>SID</i> ist die optionale Systemkennung (System-ID) der Datenbank, die sich auf dem Server befindet und mit der Sie arbeiten möchten. Einige gängige <i>Netz_Präfixe</i> sind: P: Named Pipes T: TCP/IP X: SPX/IPX Weitere Informationen finden Sie in der SQL*Net-Dokumentation.
<b>UID</b>	Akronym für User Login ID; die Kennung, mit der sich der Benutzer anmeldet.
<b>PWD</b>	Abkürzung für Password; das vom Benutzer angegebene Kennwort.

Um beispielsweise zu der auf dem Server "HRSRVR" befindlichen Datenquelle "Personal (Arbeiter/Angestellte)" eine Verbindung herzustellen und dafür den NetBIOS SQL\*Net-Treiber, die Anmeldekennung (Login ID) "Schmidt" und das Kennwort "Sesam" zu verwenden, müssten Sie folgende Verbindungszeichenfolge angeben:

```
DSN=Personal (Arbeiter/Angestellte);DBQ=b:HRSRVR;UID=Schmidt;PWD=Sesam
```

**Siehe auch**

Für alle Benutzer

[Herstellen einer Verbindung zu einer ORACLE-Datenquelle](#)

Für Programmierer

[Die Realisierung von SQLDriverConnect \(Programmierung\)](#)

## **SQL-Anweisungen (fortgeschritten)**

Siehe auch

Der ORACLE-Treiber unterstützt vollständig die ODBC SQL-Grammatikstufe Core. Darüber hinaus unterstützt er fast alle SQL-Anweisungen, die zur ODBC SQL-Grammatikstufe Extended gehören. Wie vom ODBC-Konzept her vorgesehen, wird der ORACLE-Treiber SQL-spezifische Grammatik an ORACLE Server weitergeben.

Unter den folgenden Hilfethemen ist die SQL-Grammatik beschrieben, die der ORACLE-Treiber realisiert.

Für fortgeschrittene Benutzer

Beschränkungen der ODBC SQL-Grammatik (fortgeschritten)

Nicht unterstützte ODBC SQL-Grammatik (fortgeschritten)

Für Programmierer

Die Realisierung der ODBC SQL-Grammatik (fortgeschritten)

**Siehe auch**

Für fortgeschrittene Benutzer

[Datentypen \(fortgeschritten\)](#)

Für Programmierer

[Rückgabewerte von SQLGetInfo \(Programmierung\)](#)

## **Beschränkungen der ODBC SQL-Grammatik (fortgeschritten)**

Die einzige Beschränkung, die der ORACLE-Treiber und ORACLE Server bezüglich der ODBC SQL-Grammatik bedingen, ergibt sich für die Länge einer Anweisung.

Die Länge einer Anweisung ist auf etwa 2000 Byte beschränkt. Der ORACLE-Treiber wandelt Parametermarken in ":cn" um, wobei *n* die bei 0 beginnende Positionsnummer der jeweiligen Marke ist. Außerdem wandelt der ORACLE-Treiber auch ODBC-Escape-Klauseln in SQL-spezifische Grammatik um. Eine sich ergebende Anweisung muß kürzer als 2000 Byte sein. Diese Einschränkung ist durch das SQL\*Net-Modul bedingt, das unter Microsoft Windows ausgeführt wird.

## Nicht unterstützte ODBC SQL-Grammatik (fortgeschritten)

Der ORACLE-Treiber unterstützt einschließlich IEF (Integrity Enhancement Facility) alle zu den ODBC-Grammatiken Core und Extended gehörenden SQL-Anweisungen und -Klauseln ohne Einschränkungen. Lediglich für die nachstehend aufgeführten Anweisungen sind einige Einschränkungen zu beachten:

Betroffene Anweisung	Erläuterung
DELETE	Die Klausel WHERE CURRENT OF <i>Cursor-Name</i> wird nicht unterstützt (positionierte DELETE-Anweisung).
IEF	Die Klauseln CASCADE und RESTRICT der Anweisungen DROP TABLE, DROP VIEW und REVOKE.
Prozeduren	Aufrufe von Prozeduren.
UPDATE	Die Klausel WHERE CURRENT OF <i>Cursor-Name</i> wird nicht unterstützt (positionierte UPDATE-Anweisung).

## Die Realisierung der ODBC SQL-Grammatik (Programmierung)

Besondere Beachtung hinsichtlich der Art und Weise, wie die ODBC SQL-Grammatik ausgeführt ist, erfordert nur die Realisierung der Vergleichsprädikate.

Ist ein Vergleichsprädikat so aufgebaut, daß der zweite Ausdruck des Vergleichs eine Parametermarke ist, und ist dieser Parameter mit Hilfe der Funktion **SQLSetParam** mit SQL\_NULL\_DATA belegt, wird der Vergleich mißlingen. Dies entspricht der NULL-Prädikat-Grammatik in ODBC SQL.

## Datentypen (fortgeschritten)

Siehe auch

Der ORACLE-Treiber setzt die ORACLE Server-Datentypen in Beziehung zu den entsprechenden ODBC SQL-Datentypen (er bildet die ORACLE-Datentypen auf die entsprechenden ODBC SQL-Datentypen ab). In der folgenden Tabelle sind sowohl alle ORACLE Server-Datentypen als auch die ihnen entsprechenden ODBC SQL-Datentypen zusammengestellt.

<b>ORACLE Server SQL-Datentypen</b>	<b>ODBC SQL- Datentypen</b>
CHAR	SQL_VARCHAR*
DATE	SQL_TIMESTAMP
FLOAT	SQL_FLOAT
LONG	SQL_LONGVARCHAR
LONG RAW	SQL_LONGVARBINARY
NUMBER	SQL_NUMERIC
NUMBER(5,0)	SQL_SMALLINT
NUMBER(10,0)	SQL_INTEGER
RAW	SQL_VARBINARY
VARCHAR	Nicht unterstützt*
VARCHAR2	Nicht unterstützt*

\* Dem Datentyp CHAR ist bei ORACLE Server in der Version 6.0 eine variable Länge und bei ORACLE Server in der Version 7.0 eine feste Länge zugeordnet. Die Datentypen VARCHAR und VARCHAR2 werden von ORACLE Server nur in der Version 7.0 unterstützt. Der ORACLE-Treiber arbeitet mit den Datentypen von ORACLE Server, Version 6.0.

---

**Anmerkung** Alle Konvertierungen, die im Anhang D des Handbuchs *Microsoft ODBC SDK Programmer's Reference* aufgeführt sind, werden für die oben aufgeführten ODBC SQL-Datentypen unterstützt.

---

Unter den folgenden Hilfethemen sind die Datentypen beschrieben, die im ORACLE-Treiber realisiert sind.

Für fortgeschrittene Benutzer

[Die Realisierung der Datentypen \(fortgeschritten\)](#)

[Beschränkungen der Datentypen \(fortgeschritten\)](#)

Für Programmierer

[Die Realisierung der Datentypen \(Programmierung\)](#)

**Siehe auch**

Für fortgeschrittene Benutzer

[SQL-Anweisungen \(fortgeschritten\)](#)

## **Die Realisierung der Datentypen (fortgeschritten)**

Besondere Beachtung hinsichtlich der Art und Weise, wie die Datentypen ausgeführt sind, muß der fortgeschrittene Benutzer lediglich der Realisierung des Datentyps SQL\_TIMESTAMP schenken.

Der in ORACLE Server definierte Datentyp DATE umfaßt sowohl das Datum als auch die Uhrzeit. In der von ORACLE Server verwendeten SQL-spezifischen Grammatik wird eine in einer DATE-Spalte festgehaltene Uhrzeit nur dann zurückgegeben, wenn sie von einer Abfrage explizit angefordert wird. Da der ORACLE-Treiber den Datentyp DATE auf den Datentyp SQL\_TIMESTAMP abbildet, liefert der Treiber aber trotzdem sowohl die in einer DATE-Spalte festgehaltenen Datumsangaben als auch die Uhrzeiten.

## Beschränkungen der Datentypen (fortgeschritten)

Der ORACLE-Treiber und ORACLE Server bedingen bezüglich der Datentypen die folgenden Beschränkungen.

Betroffener Datentyp	Erläuterung
Literale	ORACLE Server beschränkt Literale in SQL-Anweisungen auf 255 Byte.
<u>Anzahl an Datenbyte</u>	Wird mit Microsoft LAN Manager das Protokoll TCP/IP eingesetzt, kann nur eine begrenzte Anzahl an Byte an ORACLE Server gesendet beziehungsweise von dort abgerufen werden.
SQL_LONGVARCHAR	Der ORACLE-Treiber begrenzt SQL_LONGVARCHAR-Daten (LONG-Daten) auf 32.768 Byte.
SQL_LONGVARCHAR und SQL_LONGVARBINARY	ORACLE Server läßt pro Tabelle nur eine Spalte zu, die mit einem der LONG-Datentypen definiert ist. Die LONG-Datentypen sind SQL_LONGVARCHAR (LONG) und SQL_LONGVARBINARY (LONG RAW).

## Die Realisierung der Datentypen (Programmierung)

Bezüglich der Art und Weise, wie die Datentypen ausgeführt sind, ist für Programmierer nur die Realisierung der Datentypen CHAR, VARCHAR und VARCHAR2 wichtig.

Der ORACLE-Treiber arbeitet mit den Datentypen der Version 6.0 von ORACLE Server. Da sich die alphanumerischen Datentypen der Versionen 6.0 und 7.0 voneinander unterscheiden, können die von der Funktion **SQLGetTypeInfo** gelieferten Informationen irreführend sein, wenn die Datenquelle in der Version 7.0 vorliegt.

Der Datentyp CHAR besitzt innerhalb der Version 6.0 eine variable Länge und wird vom ORACLE-Treiber daher auf SQL\_VARCHAR abgebildet. Innerhalb der Version 7.0 hat der Datentyp CHAR dagegen eine feste Länge, wird aber weiterhin auf SQL\_VARCHAR abgebildet. Die Datentypen VARCHAR und VARCHAR2 gibt es nur innerhalb der Version 7.0. Sie werden deshalb vom ORACLE-Treiber auf keinen ODBC SQL-Datentyp abgebildet.

Bei einem Wert *fSqlType* für SQL\_VARCHAR liefert **SQLGetTypeInfo** den ORACLE Server-Datentyp CHAR. Bei einem Wert von *fSqlType* für SQL\_CHAR liefert **SQLGetTypeInfo** keinen ORACLE Server-Datentyp. Dieses Verhalten entspricht den Datentypen der Version 6.0, für die Datentypen der Version 7.0 ist dies dagegen irreführend.

## Fehlermeldungen (fortgeschritten)

Wenn ein Fehler auftritt, liefert der ORACLE-Treiber die Nummer des Datenquellenfehlers, den SQL-STATUS (SQLSTATE, ein ODBC-Fehlercode) und eine Fehlermeldung. Der Treiber ermittelt diese Informationen sowohl aus den Fehlern, die er selbst feststellt, als auch aus den Fehlern, die ORACLE Server meldet.

### Datenquellenfehler (Native Error)

Für Fehler, die innerhalb einer Datenquelle auftreten, liefert der ORACLE-Treiber den Datenquellenfehler, den ihm ORACLE Server gemeldet hat. Für Fehler, die der ORACLE-Treiber oder der Treiber-Manager festgestellt hat, liefert der ORACLE-Treiber den Datenquellenfehler Null.

### SQL-STATUS (SQLSTATE)

Für Fehler, die innerhalb einer Datenquelle auftreten, bildet der ORACLE-Treiber den gemeldeten Datenquellenfehler auf den entsprechenden SQL-STATUS ab. Wenn der ORACLE-Treiber einen Fehler feststellt, erzeugt er den entsprechenden SQL-STATUS (SQLSTATE). Wenn der Treiber-Manager einen Fehler feststellt, erzeugt dieser den entsprechenden SQL-STATUS (SQLSTATE).

### Fehlermeldung

Für Fehler, die innerhalb einer Datenquelle auftreten, liefert der ORACLE-Treiber eine Fehlermeldung, die auf der von ORACLE Server gelieferten Meldung basiert. Für Fehler, die innerhalb des ORACLE-Treibers oder des Treiber-Managers auftreten, liefert der ORACLE - Treiber eine Fehlermeldung, die auf dem Text basiert, der dem SQL-STATUS (SQLSTATE) zugeordnet ist.

Eine Fehlermeldung hat folgendes Format:

*[Hersteller][ODBC-Komponente][Datenquelle]Fehlermeldung*

Die in eckigen Klammern ([ ]) stehenden Präfixe geben die Fehlerquelle an. In der folgenden Tabelle sind die vom ORACLE-Treiber gelieferten Werte dieser Präfixe zusammengestellt. Wenn der jeweilige Fehler innerhalb einer Datenquelle auftritt, kennzeichnen die Präfixe *[Hersteller]* und *[ODBC-Komponente]* den Hersteller und Namen der ODBC-Komponente, die die Fehlermeldung von der Datenquelle empfangen hat.

Fehlerquelle	Präfix	Wert
Treiber-Manager	<i>[Hersteller]</i>	[Microsoft]
	<i>[ODBC-Komponente]</i>	[ODBC DLL]
	<i>[Datenquelle]</i>	Nicht angegeben
ORACLE-Treiber	<i>[Hersteller]</i>	[Microsoft]
	<i>[ODBC-Komponente]</i>	[ODBC Oracle Driver]
	<i>[Datenquelle]</i>	Nicht angegeben
ORACLE Server	<i>[Hersteller]</i>	[Microsoft]
	<i>[ODBC-Komponente]</i>	[ODBC Oracle Driver]
	<i>[Datenquelle]</i>	[Oracle OCI]

## Rückgabewerte von SQLGetInfo (Programmierung)

In der folgenden Tabelle sind die für das Argument *flInfoType* in der Sprache C formulierten #define-Konstanten sowie die entsprechenden von **SQLGetInfo** gelieferten Werte zusammengestellt. Ein Anwendungsprogramm kann diese Informationen abrufen, indem es die mit der C-Anweisung #define definierten Konstanten im Argument *flInfoType* an **SQLGetInfo** übergibt.

<i>flInfoType</i> -Wert (#define)	Gelieferter Wert
SQL_ACCESSIBLE_TABLES	Nein.
SQL_ACCESSIBLE_PROCEDURES	Nein.
SQL_ACTIVE_CONNECTIONS	15
SQL_ACTIVE_STATEMENTS	Keine Beschränkung.

SQL_CONCAT_NULL_BEHAVIOR	Non-NULL.
SQL_CONVERT_FUNCTIONS	Keiner.
SQL_CONVERT_Typ, wobei Typ der SQL-Datentyp ist (zum Beispiel CHAR)	Keiner.
SQL_CURSOR_COMMIT_BEHAVIOR	Cursor werden aufrechterhalten.
SQL_CURSOR_ROLLBACK_BEHAVIOR	Cursor werden aufrechterhalten.
SQL_DEFAULT_TXN_ISOLATION	Transaktion 2 kann Zeilen, die mit den Auswahlkriterien von Transaction 1 (SQL_TXN_REPEATABLE_READ) übereinstimmen, einfügen (jedoch nicht aktualisieren oder löschen).
SQL_DBMS_NAME	Oracle
SQL_DBMS_VER	06.00.0000 oder 07.00.0000 (Beachten Sie, daß ORACLE Server in der Version 6.0.34 oder höher vorliegen muß, um mit dem ORACLE-Treiber verwendet werden zu können.).
SQL_DRIVER_NAME	SQORA.DLL
SQL_DRIVER_VER	01.00.nnnn, wobei nnnn das Erstellungsdatum darstellt.
SQL_EXPRESSIONS_IN_ORDERBY	Ja.
SQL_FETCH_DIRECTION	Nächste.
SQL_IDENTIFIER_CASE	Bei Bezeichnern findet die Groß-/Kleinschreibung keine Beachtung (not case-sensitive).
SQL_IDENTIFIER_QUOTE_CHAR	" (Anführungszeichen)
SQL_MAX_COLUMN_NAME_LEN	30
SQL_MAX_CURSOR_NAME_LEN	18
SQL_MAX_OWNER_NAME_LEN	30
SQL_MAX_PROCEDURE_NAME_LEN	0
SQL_MAX_QUALIFIER_NAME_LEN	0
SQL_MAX_TABLE_NAME_LEN	30
SQL_MULT_RESULT_SETS	Nein.
SQL_MULTIPLE_ACTIVE_TXN	Ja.
SQL_NUMERIC_FUNCTIONS	ABS, CEILING, FLOOR, MOD, SIGN, SQRT
SQL_ODBC_API_CONFORMANCE	Ebene 1 (Level 1).
SQL_ODBC_SAG_CLI_CONFORMANCE	Stimmt mit der SAG-Spezifikation überein.
SQL_ODBC_SQL_CONFORMANCE	Core.
SQL_ODBC_SQL_OPT_IEF	Nein.
SQL_OUTER_JOINS	Ja.
SQL_OWNER_TERM	Owner
SQL_PROCEDURE_TERM	Nicht unterstützt.

SQL_PROCEDURES	Nein.
SQL_QUALIFIER_NAME_SEPARATOR	. (Punkt)
SQL_QUALIFIER_TERM	Nicht unterstützt.
SQL_ROW_UPDATES	Nein.
SQL_SCROLL_CONCURRENCY	Schreibgeschützt.
SQL_SCROLL_OPTIONS	Nur vorwärts.
SQL_SEARCH_PATTERN_ESCAPE	\ (Umgekehrter Schrägstrich)
SQL_STRING_FUNCTIONS	ASCII, CHAR, CONCAT, INSERT, LCASE, LEFT, LENGTH, LOCATE, LTRIM, REPEAT, REPLACE, RIGHT, RTRIM, SUBSTRING, UCASE
SQL_SYSTEM_FUNCTIONS	DBNAME, IFNULL, USERNAME
SQL_TABLE_TERM	Table
SQL_TIMEDATE_FUNCTIONS	NOW, CURDATE, DAYOFMONTH, DAYOFWEEK, DAYOFYEAR, MONTH, QUARTER, WEEK, YEAR, CURTIME, HOUR, MINUTE, SECOND
SQL_TXN_CAPABLE	Transaktionen dürfen nur <u>DML</u> -Anweisungen (SELECT, INSERT, UPDATE und DELETE) enthalten.
SQL_TXN_ISOLATION_OPTION	Transaktion 2 kann Zeilen, die mit den Auswahlkriterien von Transaktion 1 (SQL_TXN_REPEATABLE_READ) übereinstimmen, einfügen (jedoch nicht aktualisieren oder löschen).

## Die ODBC API-Funktionen (Programmierung)

Siehe auch

Der ORACLE-Treiber unterstützt alle Funktionen der Stufe Core und der Stufe 1 (Level 1) sowie die folgenden Funktionen der Stufe 2 (Level 2):

SQLDataSources  
SQLMoreResults  
SQLNativeSql

Außerdem unterstützt der ORACLE-Treiber Umwandlungs-DLL-Dateien.

Unter den folgenden Hilfethemen sind die im ORACLE-Treiber realisierten ODBC API-Funktionen erläutert.

Für Programmierer

[Erweiterungen der ODBC API-Funktionen \(Programmierung\)](#)

[Die Realisierung der ODBC API-Funktionen \(Programmierung\)](#)

[Beschränkungen der ODBC API-Funktionen \(Programmierung\)](#)

**Siehe auch**

Für fortgeschrittene Benutzer

[Fehlermeldungen \(fortgeschritten\)](#)

## **Erweiterungen der ODBC API-Funktionen (Programmierung)**

**SQLGetData** ist die einzige Funktion der zum ORACLE-Treiber gehörenden Funktionen, die über die im Handbuch *Microsoft ODBC SDK Programmer's Reference* angegebenen Spezifikationen hinausgeht.

**SQLGetData** kann aus jeder beliebigen Spalte Daten abrufen, unabhängig davon, ob sich rechts von der Spalte verknüpfte Spalten befinden.

## Die Realisierung der ODBC API-Funktionen (Programmierung)

In der folgenden Tabelle ist beschrieben, wie der ORACLE-Treiber bestimmte Funktionen realisiert.

<b>Funktion</b>	<b>Erläuterung</b>
<b>SQLConnect</b>	Steht in der Konfigurationsdatei einer ORACLE-Arbeitsstation (CONFIG.ORA) das Schlüsselwort LOCAL, ist für <b>SQLConnect</b> nur eine Benutzerkennung bzw. ein Anmelde-name (user ID) und ein Kennwort (password) erforderlich.
<b>SQLDriverConnect</b>	<b>SQLDriverConnect</b> verwendet die Schlüsselwörter DSN, DBQ, UID und PWD.
<b>SQLMoreResults</b>	Da ORACLE nicht in der Lage ist, mehrere Ergebnisgruppen zu unterstützen, liefert <b>SQLMoreResults</b> immer SQL_NO_DATA_FOUND und stellt den Status der <i>hstmt</i> auf allocated/zugeordnet (wenn die SQL-Anweisung mit <b>SQLExecDirect</b> ausgeführt wurde) oder prepared/vorbereitet ein (wenn die SQL-Anweisung mit <b>SQLExecute</b> ausgeführt wurde).
<b>SQLSpecialColumns</b>	Wird die Funktion <b>SQLSpecialColumns</b> mit der Option SQL_BEST_ROWID aufgerufen, liefert sie immer die ROWID-Spalte.

## Die Realisierung von SQLDriverConnect (Programmierung)

Steht in der Konfigurationsdatei einer ORACLE-Arbeitsstation (CONFIG.ORA) der Parameter LOCAL, ist für **SQLDriverConnect** nur eine Benutzerkennung bzw. ein Anmelde-name (user ID) und ein Kennwort (password) erforderlich.

Eine **SQLDriverConnect**-Verbindungszeichenfolge verwendet die folgenden Schlüsselwörter:

<b>Schlüsselwort</b>	<b>Erläuterung</b>
<b>DSN</b>	Akronym für Data Source Name. Der Name der Datenquelle, wie er in der Datei ODBC.INI steht.
<b>DBQ</b>	Eine ORACLE-Verbindungszeichenfolge der Form: <i>Netz_Präfix:Hostname[:SID]</i> <i>Netz_Präfix</i> gibt an,

welcher SQL\*Net-Treiber verwendet werden soll; *Hostname* gibt den Namen oder Alias-Namen des Servers an, auf dem sich ORACLE Server befindet; *SID* ist die optionale Systemkennung (System-ID) der Datenbank, die sich auf dem Server befindet und mit der Sie arbeiten möchten. Einige gängige *Netz\_Präfixe* sind:

P: Named Pipes  
 T: TCP/IP  
 X: SPX/IPX

Weitere Informationen finden Sie in der SQL\*Net-Dokumentation.

**UID** Akronym für User Login ID; die Kennung, mit der sich der Benutzer anmeldet.

**PWD** Abkürzung für Password; das vom Benutzer angegebene Kennwort.

## Beschränkungen der ODBC API-Funktionen (Programmierung)

Die folgenden zum ORACLE-Treiber gehörenden Funktionen halten die im Handbuch *Microsoft ODBC SDK Programmer's Reference* aufgeführten Spezifikationen nicht ein.

Funktion	Erläuterung
<b>SQLCancel</b>	Da der ORACLE-Treiber die Schnittstelle OCI (ORACLE Call Interface) verwendet und OCI asynchrone Verarbeitung nicht unterstützt, ist <b>SQLCancel</b> mit <b>SQLFreeStmt(<i>hstmt</i>,SQL_CLOSE)</b> identisch.
<b>SQLGetConnectOption</b> und <b>SQLSetConnectOption</b>	Diese Funktionen unterstützen nur die Optionen SQL_ACCESS_MODE, SQL_AUTOCOMMIT, SQL_TRANSLATE_DLL und SQL_TRANSLATE_OPTION.
<b>SQLGetCursorName</b> und <b>SQLSetCursorName</b>	Diese Funktionen lesen und setzen den Cursor-Namen wie im Handbuch <i>Microsoft ODBC SDK Programmer's Reference</i> angegeben. Trotzdem ist es Ihnen nicht möglich, den Cursor-Namen in einer SQL-Anweisung zu verwenden, da der ORACLE-Treiber keine positionierten UPDATE- und DELETE-Anweisungen

**SQLGetStmtOption**  
und  
**SQLSetStmtOption**

unterstützt.  
Diese Funktionen unterstützen  
nur die Optionen  
SQL\_MAX\_LENGTH,  
SQL\_MAX\_ROWS und  
SQL\_NOSCAN.

## Probleme aufgrund der Realisierung (Programmierung)

Die folgenden realisierungsspezifischen Probleme können sich auf den Einsatz des ORACLE-Treibers auswirken.

<b>Problemursache</b>	<b>Erläuterung</b>
ORACLE Call Interface (OCI)	Der ORACLE-Treiber greift über OCI auf die Daten einer ORACLE-Datenquelle zu.
Setup-DLL-Datei	Der ODBC-Administrator ruft die Funktion <b>ConfigDSN</b> auf, wenn Benutzer Datenquellen konfigurieren. Für den ORACLE-Treiber befindet sich diese Funktion in einer Setup-DLL-Datei (SQORASTP.DLL).
Transaktionen	Arbeitet der Treiber im Modus Manual-Commit, können bei ORACLE Server in der Version 6.0 nur <u>DML</u> -Anweisungen (Data Manipulation Language = Datenbearbeitungssprache) explizit übernommen (commit) oder zurückgesetzt (roll back) werden. <u>DDL</u> -Anweisungen (Data Definition Language = Datenbearbeitungssprache) werden unabhängig davon, welcher Commit-Modus aktiv ist, automatisch übernommen (commit). Sowohl im Modus Manual-Commit als auch im Modus Auto-Commit können DML- und DDL-Anweisungen gemischt werden. Bei ORACLE Server in der Version 7.0 können einige DDL-Anweisungen explizit übernommen (commit) oder zurückgesetzt (roll back) werden. Weitere Informationen finden Sie in der ORACLE-Dokumentation.

**API**

Akronym für Application Programming Interface (Schnittstelle für die Anwendungsprogrammierung). Eine Zusammenstellung von Routinen, die ein Anwendungsprogramm (zum Beispiel Microsoft Access) dazu einsetzt, Basisfunktionen aufzurufen und auszuführen.

## **Zeichensatz**

Ein Zeichensatz ist eine Zusammenstellung aus 256 Zeichen (Buchstaben, Ziffern und Symbole), die für ein Land oder eine Sprache spezifisch sind. Jeder Zeichensatz ist in einer Tabelle definiert, die als Codeseite bezeichnet wird. Alle von dem ANSI-Zeichensatz abweichenden Zeichensätze werden als OEM-Zeichensätze bezeichnet (OEM ist das Akronym für Original Equipment Manufacturer). Microsoft Windows verwendet den ANSI-Zeichensatz (Codeseite 1007).

### **Übereinstimmungsstufe (conformance level)**

Einige Anwendungsprogramme können nur mit Treibern arbeiten, die bestimmte Funktionalitätsstufen beziehungsweise Übereinstimmungsstufen unterstützen. Beispielsweise kann es sein, daß ein Anwendungsprogramm Treiber benötigt, die den Benutzer dazu auffordern können, das Kennwort einer Datenquelle einzugeben. Diese Fähigkeit ist Bestandteil der Stufe 1 (Level 1) der für die Schnittstelle für die Anwendungsprogrammierung (API) definierten Übereinstimmungsstufen.

Jeder ODBC-Treiber entspricht sowohl einer von drei API-Stufen (Core, Stufe 1 oder Stufe 2) als auch einer von drei SQL-Grammatikstufen (Minimum, Core oder Extended). Einige Treiber unterstützen Funktionen, die nicht der für sie angegebenen Stufe, sondern einer höheren Stufe zugeordnet sind.

Ausführliche Informationen zu den Übereinstimmungsstufen finden Sie in dem Handbuch *Microsoft ODBC SDK Programmer's Reference*.

## **Datenquelle**

Eine Datenquelle umfaßt die Daten, auf die ein Benutzer zugreifen möchte, sowie die Informationen, die erforderlich sind, um auf die Daten zuzugreifen. Beispiele zu Datenquellen:

- Ein Verzeichnis, das dBASE-Dateien enthält, auf die Sie zugreifen möchten.
- Eine SQL Server-Datenbank, der Server, auf dem sich die Datenbank befindet, sowie das Netzwerk, über das auf diesen Server zugegriffen werden kann.

**DBMS**

Akronym für Datenbank-Managementsystem. Die Software, mit der Daten strukturiert, analysiert, gesucht, aktualisiert und abgerufen werden.

**DDL**

Akronym für Data Definition Language (Datendefinitionssprache). Umfaßt alle SQL-Anweisungen, mit denen Datenobjekte und deren Attribute definiert werden können, so zum Beispiel die Anweisungen CREATE TABLE, DROP VIEW und GRANT.

**DLL**

Akronym für Dynamic-Link Library (DLL-Datei). Eine Zusammenstellung von Routinen, die von einem oder mehreren Anwendungsprogrammen aufgerufen werden können, um allgemeine Aufgaben zu erledigen. Die ODBC-Treiber sind DLL-Dateien.

**DML**

Akronym für Data Manipulation Language (Datenbearbeitungssprache). Umfaßt alle SQL-Anweisungen, mit denen Daten bearbeitet werden können, so zum Beispiel die Anweisungen UPDATE, INSERT und DELETE.

**ODBC**

Akronym für Open Database Connectivity, eine Schnittstelle für offenen und herstellerunabhängigen Datenbankzugriff. Ein Treiber-Manager sowie eine Reihe von ODBC-Treibern, die es Anwendungsprogrammen ermöglichen, mit SQL als Standardsprache auf Daten zuzugreifen.

**ODBC-Treiber-Manager**

Eine DLL-Datei (Dynamic-Link Library), mit deren Hilfe auf die ODBC-Treiber zugegriffen werden kann.

**ODBC-Treiber**

Eine DLL-Datei (Dynamic-Link Library), die von einer ODBC-kompatiblen Anwendung, zum Beispiel Microsoft Excel, dazu eingesetzt werden kann, Zugriff auf eine bestimmte Datenquelle zu erhalten. Für jedes Datenbank-Managementsystem (DBMS), beispielsweise SQL Server oder dBASE, ist ein eigener Treiber erforderlich.

**SQL**

Akronym für Structured Query Language ( strukturierte Abfragesprache). Eine Sprache, mit der Daten abgerufen, aktualisiert und verwaltet werden können.

**SQL-Anweisung**

Ein in SQL (Structured Query Language - Stukturierte Abfragesprache) formulierter Befehl; auch als Abfrage bezeichnet. Eine SQL-Anweisung legt fest, welche Operation durchgeführt werden soll (z.B. SELECT, DELETE oder CREATE TABLE), an welchen Tabellen und Spalten die Operation durchgeführt werden soll, und welche Einschränkungen auf diese Operation zutreffen.

**Umwandlungsoption**

Eine Option, die angibt, wie ein Umwandlungsprogramm Daten umwandelt. Beispielsweise kann eine Umwandlungsoption die Zeichensätze angeben, zwischen denen ein Umwandlungsprogramm alphanumerische Daten (Zeichendaten) umwandelt. Eine solche Option kann aber auch den Schlüssel angeben, nach dem Daten verschlüsselt und entschlüsselt werden.

## **Umwandlungsprogramm**

Eine DLL-Datei (Dynamic-Link Library), die alle Daten umwandelt, die zwischen einem Anwendungsprogramm, zum Beispiel Microsoft Access, und einer Datenquelle übertragen werden. Die häufigste Verwendung eines Umwandlungsprogrammes besteht darin, alphanumerische Daten von einem Zeichensatz in einen anderen umzuwandeln. Ein Umwandlungsprogramm kann aber auch Verschlüsselungen und Entschlüsselungen oder Komprimierungen und Dekomprimierungen vornehmen.

