

Novidades

Esta versão do driver Microsoft ODBC para Oracle inclui diversas melhorias no desempenho e estabilidade. Você deverá constatar que o driver tem maior funcionalidade e velocidade devido às seguintes adições:

- Recuperação estendida aperfeiçoada com suporte para ligação baseada em linha. Consulte [Funções de nível 2](#). Para maiores detalhes, consulte *Referências do programador e Guia SDK do Microsoft ODBC 3.0*.
- Integração com o Microsoft® Transaction Server para transações distribuídas. Consulte [Utilizando o Microsoft Transaction Server](#)
- Suporte para a sintaxe do Oracle Packaged Procedure em funções de catálogo. Consulte [Retornando parâmetros de matriz a partir de procedimentos armazenados](#)
- Implementação do SQLDescribeParam para fornecer maior exatidão em descrições de parâmetro para instruções SQL. Consulte [Funções de nível 2](#)
- Capacidade de retornar matrizes a partir de procedimentos armazenados. Consulte [Retornando parâmetros de matriz a partir de procedimentos armazenados](#)
- Suporte para indicadores. Consulte [Funções de nível 2](#) e [Tabela de opções de instrução](#)
- Arquivo estendido da **Ajuda**

Esta versão do driver apresenta maior estabilidade quando testada em outros ambientes como o Microsoft Transaction Server e o Internet Information Server.

Visão geral

O driver Microsoft® ODBC para Oracle permite que você conecte seu aplicativo concordante com a conectividade aberta de banco de dados (ODBC, Open Database Connectivity) a um banco de dados Oracle. Esta versão oferece maior desempenho e recursos adicionais de controle, incluindo acesso a pacotes de PL/SQL, integração com XA/DTC e acesso ao Oracle a partir do Internet Information Server (IIS). O driver ODBC se conforma à especificação da conectividade aberta de banco de dados (ODBC) descrita na *Referência do Programador ODBC (versão 2.0)* para sua plataforma.

Este arquivo da **Ajuda** descreve como instalar, configurar e utilizar o driver ODBC e inclui as seguintes seções:

- [Requisitos do sistema](#)
- [Adicionando e modificando fontes de dados usando o Programa de Instalação](#)
- [Configurando o driver ODBC para Oracle](#)
- [Conectando-se a uma fonte de dados](#)
- [Níveis de conformidade do ODBC](#)
- [Mapeando tipos de dados](#)
- [Utilizando o Microsoft Internet Information Server](#)
- [Utilizando a autenticação do sistema operacional](#)
- [Limitações do uso de cursores controlados por conjunto de teclas](#)
- [Retornando parâmetros de matriz a partir de procedimentos armazenados](#)
- [Tabela de opções de conexão](#)
- [Tabela de opções de instrução](#)
- [Tabela de tipo de cursor e de combinações de concorrência](#)
- [Mensagens de erro](#)
- Funções API
 - [Observações de segurança de segmento sobre funções API](#)
 - [Observações sobre funções API](#)
 - [Funções em nível do núcleo](#)
 - [Funções de nível 1](#)
 - [Funções de nível 2](#)

Requisitos do sistema

Para utilizar o driver Microsoft ODBC para Oracle, você precisa ter o Windows 95 ou o Windows NT e o software cliente do Oracle, versão 7.3 ou mais recente instalados em seu sistema Windows. O driver Microsoft ODBC para Oracle suporta somente o SQL*Net 2.3 ou mais recente. Para obter maiores informações sobre os produtos Oracle, consulte seu conjunto de documentação do Oracle.

Adicionando e modificando fontes de dados utilizando o Programa de Instalação

Uma fonte de dados identifica um caminho até os dados, que podem incluir uma biblioteca de rede, um servidor, um banco de dados e outros atributos — neste caso, a fonte de dados é o caminho até um banco de dados do Oracle. Para se conectar a uma fonte de dados, o **Gerenciador de drivers** verifica o Registro do Windows em busca de informações específicas da conexão.

A entrada de Registro criada pelo **Administrador de fonte de dados ODBC** é utilizada pelo **Gerenciador de drivers ODBC** e drivers ODBC. Esta entrada contém informações sobre cada fonte de dados e seu driver associado. Antes que você possa se conectar a uma fonte de dados, suas informações de conexão devem ser adicionadas ao Registro.

Para adicionar e configurar fontes de dados, acesse o Administrador ODBC através do **ODBC de 32 bits** no **Painel de controle** do Windows. O Administrador ODBC atualizará as informações da conexão com sua fonte de dados. À medida que você adicionar as fontes de dados, o Administrador ODBC atualizará as informações do Registro para você.

► **Adicionando uma fonte de dados para Windows**

- 1 Para iniciar o Administrador ODBC, clique duas vezes no ícone **ODBC** no **Painel de controle** do Windows.
- 2 Ao ver a caixa de diálogo **Administrador de fonte de dados ODBC**, clique no botão **Adicionar**. A caixa de diálogo **Criar nova fonte de dados** aparece.
- 3 Selecione o **Driver ODBC** e, em seguida, clique em **Concluir**. A caixa de diálogo **Configurar Microsoft ODBC para Oracle** aparece.
- 4 Na caixa **Nome da fonte de dados**, digite o nome da fonte de dados que você deseja acessar. Pode ser qualquer nome que você escolher.
- 5 Na caixa **Descrição**, digite a descrição do driver. Este é um campo opcional que descreve o driver do banco de dados ao qual a fonte de dados se conecta. Pode ser qualquer nome que você escolher.
- 6 Na caixa **Nome do usuário**, digite seu nome de usuário do banco de dados. O nome do usuário é sua identificação de usuário do banco de dados.
- 7 Na caixa **Servidor**, digite a seqüência de conexão para o motor do Servidor Oracle. A seqüência de conexão identifica o motor do Servidor Oracle que você deseja acessar.
- 8 Clique em **OK** para adicionar esta fonte de dados.

Observação A caixa de diálogo **Fontes de dados** aparece e o Administrador ODBC atualiza as informações do Registro. O nome do usuário e a seqüência de conexão que você digitar se tornarão os valores padrão de conexão à fonte de dados para esta fonte de dados. Isto é, quando você se conectar à fonte de dados utilizando uma caixa de diálogo ou uma seqüência de conexão, estes valores se tornarão as entradas padrão para a conexão à fonte de dados.

- 9 Clique em **Adicionar** para adicionar outra fonte de dados ou clique em **Fechar** para sair.

► **Modificando uma fonte de dados para Windows**

- 1 Invoque o Administrador ODBC. A caixa de diálogo **Fontes de dados** aparece.
- 2 Na caixa de diálogo **Fontes de dados**, selecione a fonte de dados do Oracle que você deseja modificar e, em seguida, clique em **Configurar**. A caixa de diálogo **Configurar Microsoft ODBC para Oracle** aparece.
- 3 Modifique os campos aplicáveis da fonte de dados e, em seguida, clique em **OK**.

Quando você terminar de modificar as informações nesta caixa de diálogo, o Administrador ODBC atualizará as informações do Registro.

Configurando o driver ODBC para Oracle

Você pode controlar o desempenho do driver ODBC se conhecer o ambiente de dados e definir corretamente os parâmetros da conexão com a fonte de dados através da caixa de diálogo **Administrador de fonte de dados ODBC** ou através dos parâmetros de seqüência de conexão. A caixa de diálogo fornece os controles a seguir para se conectar a uma fonte de dados utilizando a caixa de diálogo ou utilizando seqüências de conexão:

- **Guia NFD do usuário** Exibe uma lista de nomes de fontes de dados que estão localizadas no computador.
- **Guia NFD do sistema** Exibe uma caixa de diálogo que permite que você adicione ou remova uma fonte de dados do sistema. As fontes de dados do sistema são acessíveis a todos os usuários da máquina local.
- **Guia NFD do arquivo** Exibe uma caixa de diálogo que permite que você adicione ou remova uma fonte de dados do arquivo da máquina local. As fontes de dados do arquivo podem ser compartilhadas por todos os usuários que tenham o mesmo driver instalado.
- **Guia Drivers ODBC** Exibe uma lista de drivers ODBC instalados.
- **Guia Rastreamento** Exibe uma caixa de diálogo que permite que você especifique como o **Gerenciador de drivers ODBC** rastreará chamadas às funções ODBC. Você pode configurar separadamente o rastreamento para cada aplicativo ODBC instalado.
- **Guia Sobre** Exibe uma caixa de diálogo que lista os arquivos de componentes ODBC instalados.

Depois de adicionar uma fonte de dados, você poderá utilizar a caixa de diálogo **Administrador de fonte de dados ODBC** para configurar o acesso à sua fonte de dados. Selecione uma fonte de dados, clique em uma guia para editar ou revisar as informações.

Conectando-se a uma fonte de dados

Um aplicativo ODBC pode passar informações de conexão de diversas maneiras. Por exemplo, o aplicativo poderá fazer com que o driver sempre solicite ao usuário as informações sobre a conexão. Ou o aplicativo poderá esperar por uma seqüência de conexão que especifique a conexão com a fonte de dados. O modo como você se conecta a uma fonte de dados depende do método de conexão utilizado pelos seus aplicativos ODBC.

Uma maneira comum de se conectar a uma fonte de dados é através da caixa de diálogo **Fonte de dados**. Se seu aplicativo ODBC estiver configurado para utilizar uma caixa de diálogo, essa caixa de diálogo será exibida e solicitará que você forneça as informações apropriadas sobre a conexão com a fonte de dados.

Outra maneira é através do uso da seqüência de conexão.

► **Conectando-se a uma fonte de dados utilizando uma caixa de diálogo**

- 1 Ao ver a caixa de diálogo **Fonte de dados**, selecione uma fonte de dados do Oracle e, em seguida, clique em **OK**. As caixas de diálogo **Conectar** aparecem.
- 2 Preencha as informações apropriadas na caixa de diálogo **Conectar** e, em seguida, clique em **OK**.

Depois de verificadas as informações de conexão, seu aplicativo poderá acessar as informações contidas na fonte de dados utilizando o driver ODBC.

Atributos da seqüência de conexão

Alguns aplicativos podem requerer uma seqüência de conexão que especifique as informações da conexão com a fonte de dados ao invés de utilizar uma caixa de diálogo para obter estas informações. A seqüência de conexão é constituída por vários atributos que especificam como um driver se conecta a uma fonte de dados. Um atributo identifica informações específicas de que o driver precisa antes de poder fazer a conexão apropriada com a fonte de dados. Cada driver pode ter um conjunto de atributos diferente, mas o formato da seqüência de conexão é sempre o mesmo. A seqüência de conexão tem o seguinte formato:

```
“DSN=data-source-name[;SERVER=value] [;PWD=value] [;UID=value]
[;<Attribute>=<value>]”
```

Observação A versão do driver Microsoft ODBC para Oracle suporta o formato de seqüência de conexão ligeiramente diferente da versão 1.

Você deve especificar o nome da fonte de dados se não especificar os atributos UID, PWD, SERVER (ou CONNECTSTRING) e DRIVER. Contudo, todos os outros atributos são opcionais. Se você não especificar um atributo, esse atributo tomará como padrão aquele que você especificou na guia **NFD** do **Administrador de fonte de dados ODBC**. O valor de atributo poderá diferenciar maiúsculas de minúsculas.

Os atributos para a seqüência de conexão são os seguintes:

Atributo	Descrição	Valor padrão
NFD	O nome da fonte de dados. Este nome está listado na guia Drivers ODBC do Administrador de fonte de dados ODBC .	“”
PWD	A senha para o servidor Oracle que você deseja acessar.	“”
SERVER	A seqüência de conexão para o Servidor Oracle que você deseja acessar.	“”
UID	O nome de usuário do Servidor Oracle. Dependendo do seu sistema, este atributo pode não ser opcional — isto é, determinados bancos de dados e tabelas podem requerer este atributos para fins de segurança. Utilize “/” para usar a autenticação do sistema operacional do Oracle.	“”
BUFFERSIZE	O tamanho ótimo de buffer utilizado ao recuperar colunas. O driver otimiza a recuperação para que uma recuperação do Servidor Oracle retorne linhas suficientes para preencher um buffer deste tamanho. Valores maiores tendem a melhorar o desempenho, se você estiver recuperando um grande volume de dados.	65535
SYNONYM_COLUMNS	Quando este valor for verdadeiro (1), uma chamada de API SQLColumn() retornará informações de coluna. Caso	1

	contrário, SQLColumn() retornará somente colunas em tabelas e visualizações. O Driver ODBC fornece acesso mais rápido quando este valor não é definido.	
REMARKS	Quando este valor for verdadeiro (1), o driver retornará colunas Remarks no conjunto de resultados <u>SQLColumns</u> . O Driver ODBC fornece acesso mais rápido quando este valor não for definido.	0

Por exemplo, uma seqüência de conexão que se conecta à fonte de dados Employees que utiliza o servidor Oracle mickey.world como o Usuário Oracle cindy, seria:

```
“DSN=Employees;UID=cindy;PWD=secret;SERVER=mickey.world”
```

Um outro exemplo, a seqüência de conexão que se conecta à fonte de dados Payroll utilizando a autenticação do sistema operacional e o servidor Oracle moola seria:

```
“DSN=Payroll;UID=/;PWD=;SERVER=moola”
```

Níveis de conformidade do ODBC

O ODBC define dois tipos de padrões de conformidade para drivers — o padrão de conformidade API e o padrão de conformidade da gramática SQL. A conformidade API se refere às funções que um driver suporta. A conformidade SQL se refere à gramática SQL que o driver suporta. Cada padrão de conformidade é constituído de níveis.

Nível de conformidade API

O driver ODBC suporta as Funções API de núcleo e de nível 1. O driver também suporta as seguintes Funções de nível 2:

- SQLBrowseConnect()
- SQLDataSources()
- SQLDescribeParam()
- SQLExtendedFetch()
- SQLForeignKeys()
- SQLMoreResults()
- SQLNativeSql()
- SQLNumParams()
- SQLPrimaryKeys()
- SQLProcedureColumns()
- SQLProcedures()
- SQLSetPos()
- SQLSetScrollOptions()

Opções suportadas

O driver suporta as seguintes opções para as funções de nível 1 SQLGetConnectOption() e SQLSetConnectOption():

- SQL_ACCESS_MODE (somente SQLGetConnectOption())
- SQL_AUTOCOMMIT
- SQL_ODBC_CURSORS
- SQL_OPT_TRACEFILE
- SQL_OPT_TRACE
- SQL_TRANSLATE_DLL
- SQL_TRANSLATE_OPTION
- SQL_TXN_ISOLATION

O driver suporta as seguintes opções para as funções de nível 1 SQLGetStmtOption() e SQLSetStmtOption():

- SQL_BIND_TYPE
- SQL_CONCURRENCY
- SQL_CURSOR_TYPE
- SQL_KEYSET_SIZE
- SQL_MAX_ROW
- SQL_ROWSET_SIZE

Níveis de conformidade SQL

O driver ODBC suporta a gramática SQL mínima e a gramática SQL de núcleo e também suporta as seguintes extensões ODBC para SQL:

- Data, hora e dados de datador
- Associações externas esquerda e direita
- Funções numéricas:

abs	log	round	tan
ceiling	log10	second	truncate
cos	mod	sign	
exp	pi	sin	
floor	power	sqrt	

- Funções de data:

curdate	dayofweek	monthname	second
curtime	dayofyear	minute	week
dayname	hour	now	year
dayofmonth	month	quarter	

- Funções de seqüência de caracteres:

ascii	left	right	ucase
char	length	rtrim	
concat	ltrim	soundex	
lcase	replace	substring	

- Função de conversão de tipo:

convert

- As funções de sistema a seguir:

ifnull	user
--------	------

Mapeando tipos de dados

O Servidor Oracle suporta um conjunto de tipos de dados. O driver ODBC mapeia esses tipos de dados para seus tipos de dados SQL para ODBC apropriados. A tabela a seguir lista o tipo de dados do Servidor Oracle 7.3 e seu tipo de dados SQL para ODBC correspondente.

Tipo de dados do Servidor Oracle	Tipo de dados SQL para ODBC
CHAR	SQL_CHAR
DATE	SQL_TIMESTAMP
FLOAT	SQL_DOUBLE
INTEGER	SQL_DECIMAL
LONG	SQL_LONGVARCHAR
LONG RAW	SQL_LONGVARBINARY
NUMBER	SQL_DECIMAL
RAW	SQL_VARBINARY
VARCHAR2	SQL_VARCHAR

Observação Os tipos de dados SQL para ODBC não suportam o tipo de dados MLSLABEL do Oracle .

Utilizando o Microsoft Transaction Server

Você pode ativar o banco de dados do Oracle para trabalhar com os componentes transacionais do Microsoft Transaction Server (MTS) no Windows NT e no Windows 95. Para ativar o banco de dados do Oracle para trabalhar com os componentes do MTS que suportam transações, os administradores de sistemas devem criar uma visualização denominada V\$XATRANS\$. Para criar este script, você deve executar um script fornecido pelo Oracle. Para obter maiores informações, consulte a **Ajuda** do Microsoft Transaction Server ou a documentação do Oracle.

Utilizando o Microsoft Internet Information Server

Se você encontrar dificuldades em se conectar a partir de um script do Internet Information Server (IIS) (particularmente um erro ORA-12641), adicione a linha a seguir ao arquivo SQLNET.ORA.

```
SQLNET.AUTHENTICATION_SERVICES = (none)
```

Isto desativará o Secure Network Services para que a autenticação anônima possa funcionar.

Utilizando a autenticação do sistema operacional

A autenticação do sistema operacional do Oracle depende do sistema operacional subjacente para controlar o acesso às contas do banco de dados. Os usuários não precisam digitar uma senha quando estiverem usando este tipo de logon.

Para aproveitar este recurso, especifique “/” como a identificação do usuário e não especifique uma senha ao se conectar através de qualquer uma das APIs de conexão, SQLBrowseConnect, SQLConnect e SQLDriverConnect.

Limitações do uso de cursores controlados por conjunto de teclas

Você deve ser capaz de recuperar uma única coluna ROWID para a tabela consultada. Um cursor controlado por conjunto de teclas não pode ser utilizado em associações, consultas ou instruções que contenham as cláusulas DISTINCT, GROUP BY, UNION, INTERSECT ou MINUS.

Retornando parâmetros de matriz a partir de procedimentos armazenados

No Oracle 7.3, não se pode acessar um Tipo de registro PL/SQL, exceto a partir de um programa PL/SQL. Se um procedimento/função em pacote tiver um argumento formal definido como um Tipo de registro PL/SQL, não será possível acoplar esse argumento formal como um parâmetro. Utilize o tipo PL/SQL TABLE no driver Microsoft ODBC para Oracle para invocar os parâmetros de matriz a partir de procedimentos contendo as seqüências de escape corretas.

► Para invocar o procedimento, utilize a sintaxe a seguir

```
{call <package-name>.<proc-or-func>;
 (... , {resultset <max-records-requested> ,<formal-array-param_1>,;
 <formal-array-param_2>, ..., <formal-array-param_n> }, ... ) }
```

Observação

- O parâmetro <max-records-requested> deve ser >= o número de linhas presentes no conjunto de resultados. Caso contrário, o Oracle retornará um erro que será passado a você pelo driver.
- Você não pode utilizar os registros PL/SQL como parâmetros de matriz. Cada parâmetro de matriz pode representar somente uma única coluna de uma tabela do banco de dados.

O exemplo a seguir define um pacote contendo dois procedimentos que retornam diferentes conjuntos de resultados; em seguida, oferece duas maneiras de retornar conjuntos de resultados a partir do pacote:

Definição do pacote:

```
CREATE OR REPLACE PACKAGE SimplePackage AS

TYPE t_id is TABLE of NUMBER(5)
INDEX BY BINARY_INTEGER;

TYPE t_Course is TABLE of VARCHAR2(10)
INDEX BY BINARY_INTEGER;

TYPE t_Dept is TABLE of VARCHAR2(5)
INDEX BY BINARY_INTEGER;

PROCEDURE proc1
(   o_id      OUT   t_id,
    ao_course  OUT   t_Course,
    ao_dept    OUT   t_Dept
);

TYPE t_pk1Type1 IS TABLE OF VARCHAR2(100) INDEX BY BINARY_INTEGER;
TYPE t_pk1Type2 IS TABLE OF NUMBER INDEX BY BINARY_INTEGER;
PROCEDURE proc2
(
  i_Arg1      IN    NUMBER,
  ao_Arg2     OUT   t_pk1Type1,
  ao_Arg3     OUT   t_pk1Type2
);

END SimplePackage;

CREATE OR REPLACE PACKAGE BODY SimplePackage AS
```

```

PROCEDURE procl ( o_id OUT t_id,
ao_course OUT t_Course, ao_dept OUT t_Dept ) AS
BEGIN
  o_id(1) := 200;
  ao_course(1) := 'M101';
  ao_dept(1) := 'EEE' ;

  o_id(2) := 201;
  ao_course(2) := 'PHY320';
  ao_dept(2) := 'ECE' ;

  END procl;
PROCEDURE proc2
(
  i_Arg1      IN    NUMBER,
  ao_Arg2     OUT   t_pk1Type1,
  ao_Arg3     OUT   t_pk1Type2
)
AS
  i NUMBER;
BEGIN
  FOR i IN 1 .. i_Arg1 LOOP
    ao_Arg2(i) := 'Row Number ' || to_char(i);
  END LOOP;
  FOR i IN 1 .. i_Arg1 LOOP
    ao_Arg3(i) := i;
  END LOOP;
END proc2;
END SimplePackage;

```

► Para invocar o procedimento, PROC1

- 1 Retornar todas as colunas em um único conjunto de resultados:

```
{ call SimplePackage.Proc1( {resultset 3, o_id , ao_course, ao_dept } ) }
```

- 2 Retornar cada coluna como um único conjunto de resultados:

```
{call SimplePackage.Proc1( {resultset 3, o_id}, {resultset 3,
ao_course}, {resultset 3, ao_dept} ) }
```

Isto retorna três conjuntos de resultados, um para cada coluna.

► Para invocar o procedimento, PROC2

- 1 Retornar todas as colunas em um único conjunto de resultados:

```
{call SimplePackage.Proc2( 5 , {resultset 5, ao_Arg2, ao_Arg3} ) }
```

- 2 Retornar cada coluna como um único conjunto de resultados:

```
{call SimplePackage.Proc2( 5 , {resultset 5, ao_Arg2}, {resultset 5,
ao_Arg3} ) }
```

Certifique-se de que seus aplicativos recuperam todos os conjuntos de resultados utilizando a API SQLMoreResults. Para obter maiores informações, consulte a documentação SDK do Microsoft ODBC.

Observação No driver Microsoft ODBC para Oracle versão 2.0, você não pode utilizar funções Oracle que retornem matrizes PL/SQL para retornar conjuntos de resultados.

Observações sobre funções API

O driver Microsoft ODBC para Oracle suporta as funções API de Núcleo, de nível 1 e de nível 2. Estas funções estão listadas em Níveis de conformidade do ODBC.

- A conformidade da interface em nível de núcleo (CLI, Core Level Interface) oferece recursos definidos na especificação ISO CLI e os recursos obrigatórios definidos na especificação X/Open CLI.
- A conformidade de nível 1 fornece a funcionalidade da interface de nível de núcleo, bem como recursos adicionais como transações.
- A conformidade de nível 2 fornece a funcionalidade de nível 1, bem como recursos adicionais como indicadores, parâmetros dinâmicos e execução assíncrona das funções ODBC.

Observações de segurança de segmento sobre funções API

O driver Microsoft ODBC para Oracle é seguro para segmento; contudo, o Oracle não permite várias instruções concomitantes ativas em uma única conexão. O driver impõe esta restrição. Em outras palavras, em aplicativos com segmentos múltiplos, embora qualquer segmento possa chamar o driver ODBC para Oracle a qualquer momento, o driver bloqueia qualquer segmento do driver na mesma conexão até que o segmento original saia do driver.

O driver não bloqueará se houver duas instruções em duas conexões diferentes. Contudo, se houver uma única conexão com duas instruções, haverá potencial para bloqueio.

Funções API em nível de núcleo

As funções neste nível abrangem o nível mínimo de conformidade de interface para os drivers ODBC.

Função API	Observações
SQLAllocConnect	Aloca memória para um identificador de conexão, <i>hdbc</i> , no ambiente identificado por <i>henv</i> . O Gerenciador de drivers processa esta chamada e chama o SQLAllocConnect do driver sempre que SQLConnect , SQLBrowseConnect ou SQLDriverConnect for chamado.
SQLAllocEnv	Exibe uma caixa de diálogo especificando o requisito para o software Oracle Client e, em seguida, retorna SQL_NULL_HANDLE. Se o software Oracle Client não estiver instalado, esta função aloca memória para um identificador de ambiente, <i>henv</i> , e inicializa a interface do nível de chamada do ODBC para ser utilizado por um aplicativo.
SQLAllocStmt	Aloca memória para um identificador de instrução e associa o identificador de instrução à conexão especificada por <i>hdbc</i> . O Gerenciador de drivers passa esta chamada para o driver, que aloca a memória para a estrutura <i>hstmt</i> .
SQLBindCol	Atribui espaço de armazenamento para uma coluna de resultados e especifica o tipo do resultado.
SQLCancel	Cancela o processamento sobre um identificador de instrução, <i>hstmt</i> . Em alguns casos, o Oracle não permite que você cancele uma instrução em execução. Isto significa que uma instrução em execução continuará até que o Oracle complete o processo, quando os resultados das instruções serão cancelados pelo driver ODBC.
SQLColAttributes	Retorna informações de descritor de uma coluna em um conjunto de resultados. As informações do descritor são retornadas como uma seqüência de caracteres, um valor de 32 bits dependente de descritor ou um valor inteiro.
SQLConnect	Conecta a uma fonte de dados. Para utilizar a Autenticação do sistema operacional do Oracle, especifique “/” como parâmetro szUID e “” como parâmetro szAuthStr.
SQLDescribeCol	Retorna o nome, tipo, precisão, escala e estado nulo da dada coluna de resultados.
SQLDisconnect	Fecha uma conexão. Se o pool de conexão estiver ativado para um ambiente compartilhado e um aplicativo chamar SQLDisconnect em uma conexão daquele ambiente, a conexão será retornada para o pool de conexões e ainda estará disponível para outros componentes que usem o mesmo ambiente compartilhado.
SQLError	Retorna erro ou informações de status sobre o último erro. O driver mantém uma pilha ou lista de erros que podem ser retornados para os argumentos <i>hstmt</i> , <i>hdbc</i> e <i>henv</i> , dependendo de como é feita a chamada para SQLError . A fila de erros é removida após cada instrução. Geralmente recupera uma mensagem de erro do Oracle e, do contrário,

	est� vazia.
SQLExecDirect	<p>Executa uma nova instru�o SQL prepar�vel. O driver utiliza valores atuais das vari�veis de marcador de par�metro se existir qualquer par�metro na instru�o. Se os nomes de sua tabela, visualiza�o ou campo contiverem espa�os, coloque os nomes entre aspas. Por exemplo, se seu banco de dados contiver uma tabela denominada <i>Minha tabela</i> e o campo <i>Meu campo</i>, coloque cada elemento do identificador entre aspas, como mostrado abaixo:</p> <pre>SELECT "Minha tabela". "Meu campo1", ; "Minha tabela"."Meu campo2" FROM "Minha tabela"</pre>
SQLExecute	Executa uma instru�o SQL preparada (uma instru�o j� preparada por SQLPrepare). O driver utiliza os valores atuais das vari�veis de marcador de par�metro se existir algum par�metro na instru�o.
SQLFetch	Recupera uma linha de um conjunto de resultados para os locais especificados pelas chamadas anteriores para SQLBindCol . Prepara o driver para uma chamada para SQLGetData para as colunas n�o-ligadas.
SQLFreeConnect	Solta um identificador de conex�o e libera toda a mem�ria alocada para o identificador.
SQLFreeEnv	Fecha o Driver ODBC e libera toda a mem�ria associada ao driver.
SQLFreeStmt	P�ra o processamento associado a um <i>hstmt</i> espec�fico, fecha todos os cursores abertos associados ao <i>hstmt</i> , descarta resultados pendentes e, opcionalmente, libera todos os recursos associados ao identificador de instru�o.
SQLGetCursorName	Retorna o nome do cursor associado ao <i>hstmt</i> dado.
SQLNumResultCols	Retorna o n�mero de colunas no cursor de um conjunto de resultados.
SQLPrepare	<p>Prepara uma instru�o SQL planejando como otimizar e executar a instru�o. A instru�o SQL � compilada para execu�o pelo SQLExecDirect.</p> <p>Se os nomes de sua tabela, visualiza�o ou campo contiverem espa�os, coloque os nomes entre aspas ("). Por exemplo, se seu banco de dados contiver uma tabela denominada <i>Minha tabela</i> e o campo <i>Meu campo</i>, coloque cada elemento do identificador como mostrado abaixo:</p> <pre>SELECT "Minha tabela"."Meu campo" FROM "Minha tabela"</pre> <p>Para obter informa�es sobre a utiliza�o de conjuntos de resultados contendo matrizes como par�metros formais, consulte <u>Retornando par�metros de matriz a partir de procedimentos armazenados</u>.</p>
SQLRowCount	O Oracle n�o oferece uma maneira de determinar o n�mero de linhas em um conjunto de resultados at� voc� recuperar a �ltima linha, assim ele retorna -1.

SQLSetCursorName	Associa um nome de cursor a um identificador de instrução ativo, <i>hstmt</i> .
SQLSetParam	Substituído por <u>SQLBindParameter</u> no ODBC 2.x
SQLTransact	Solicita uma operação de confirmação ou repetição para todas as operações ativas em todos os identificadores de instrução (<i>hstmts</i>) associados a uma conexão, ou todas as conexões associadas ao identificador de ambiente, <i>henv</i> . Se uma confirmação falhar quando no modo manual, a transação continuará ativa; você pode escolher repetir a transação ou tentar a operação de confirmação novamente. Se uma operação de confirmação falhar quando no modo de transação automática, a transação será repetida automaticamente; a transação não pode estar inativa.

Funções API de nível 1

As funções neste nível oferecem uma conformidade de interface de núcleo, além de funcionalidades adicionais como o suporte de transação.

Função API	Observações
SQLColumns	Cria um conjunto de resultados para uma tabela que é a lista de colunas da tabela ou tabelas especificadas. Quando você solicita colunas para um sinônimo PUBLIC, você deve ter definido o atributo de conexão SYNONYMCOLUMNS e especificar uma seqüência vazia como o argumento <i>szTableOwner</i> . Ao retornar colunas para sinônimos PUBLIC, o driver define a coluna TABLE NAME como uma seqüência de caracteres vazia. O conjunto de resultados contém uma coluna adicional, ORDINAL POSITION, no fim de cada linha. Este valor é a posição ordinal da coluna na tabela.
SQLDriverConnect	Conecta-se a uma fonte de dados existente. Para maiores detalhes, consulte Atributos de seqüência de conexão .
SQLGetConnectOption	Retorna a definição atual de uma opção de conexão. Esta função é parcialmente suportada: o driver suporta todos os valores para o argumento <i>fOption</i> , mas não suporta alguns dos valores de <i>vParam</i> para o argumento <i>fOption</i> SQL_TXN_ISOLATION . Para obter maiores informações, consulte Tabela de opções de conexão .
SQLGetData	Recupera o valor de um único campo no registro atual do dado conjunto de resultados.
SQLGetFunctions	Retorna TRUE para todas as funções suportadas. Implementado pelo Gerenciador de drivers .
SQLGetInfo	Retorna informações sobre o driver ODBC e a fonte de dados associada a um identificador de conexão, <i>hdbc</i> .
SQLGetStmtOption	Retorna a definição atual de uma opção de instrução. Para obter maiores informações, consulte Tabela de opções de instrução .
SQLGetTypeInfo	Retorna informações sobre os tipos de dados suportados por uma fonte de dados. O driver retorna as informações em um conjunto de resultados SQL.
SQLParamData	Utilizado em conjunção com o SQLPutData para especificar dados de parâmetro durante a execução da instrução.
SQLPutData	Permite que um aplicativo envie dados de um parâmetro ou coluna para o driver durante a execução da instrução.
SQLSetConnectOption	Fornece acesso às opções que governam os aspectos da conexão. Esta função é parcialmente suportada: o driver suporta todos os valores do argumento <i>fOption</i> , mas não suporta alguns dos valores de <i>vParam</i> para o argumento <i>fOption</i> SQL_TXN_ISOLATION . Para obter maiores informações, consulte Tabela de opções de conexão .
SQLSetStmtOption	Define opções relacionadas a um identificador de instrução, <i>hstmt</i> . Para obter maiores informações, consulte Tabela de opções de instrução .

SQLSpecialColumns	Recupera um conjunto ótimo de colunas que identifica uma linha da tabela de uma maneira exclusiva.
SQLStatistics	Recupera uma lista de estatísticas sobre uma única tabela e os índices ou nomes de marca associados à tabela. O driver retorna as informações como um conjunto de resultados.
SQLTables	<p>Retorna a lista de nomes de tabela especificados pelo parâmetro na instrução SQLTables. Se nenhum parâmetro for especificado, retornará os nomes de tabela armazenados na fonte de dados atual. O driver retorna as informações como um conjunto de resultados.</p> <p>As chamadas de tipo enumeração não receberão uma entrada de conjunto de resultados para visualizações remotas ou para visualizações locais parametrizadas. Contudo, uma chamada para SQLTables com um especificador exclusivo de nome de tabela encontrará uma correspondência para tal visualização, se presente com esse nome; isto permite que a API seja usada para verificar conflitos de nomes antes da criação de uma nova tabela.</p> <p>Sinônimos PUBLIC são retornados com um valor de TABLE_OWNER igual a "".</p> <p>As VIEWS pertencentes a SYS ou SYSTEM são identificadas como SYSTEM VIEW.</p>

Funções API de nível 2

Funções neste nível oferecem uma conformidade de interface de nível 1, além de funcionalidade adicional como suporte para indicadores.

Função API	Observações
SQLBindParameter	Associa um buffer a um marcador de parâmetro numa instrução SQL.
SQLBrowseConnect	Retorna níveis sucessivos de atributos e valores de atributo.
SQLDataSources	Lista nomes de fontes de dados. Implementado pelo Gerenciador de drivers
SQLDescribeParam	Retorna a descrição de um marcador de parâmetro associado a uma instrução SQL preparada. Retorna a melhor estimativa do que seja o parâmetro, com base na análise da instrução. Se não for possível determinar o tipo de parâmetro, SQL_VARCHAR retornará com uma extensão 2000.
SQLDrivers	Implementado pelo Gerenciador de drivers .
SQLExtendedFetch	Semelhante a SQLFetch , mas retorna várias linhas utilizando uma matriz para cada coluna. O conjunto de resultados é rolável para frente e é possível torná-lo rolável para trás se o cursor for definido como estável e não somente avanço. Para cursores somente avanço com ligação de coluna padrão, os dados de coluna de conjuntos de dados maiores que o atributo de conexão BUFFERSIZE são recuperados diretamente para os buffers de dados. Não suporta indicadores de tamanho variável e não suporta recuperação de um conjunto de linhas com deslocamento (diferente de 0) em relação a um indicador
SQLForeignKeys	Retorna uma lista de chaves externas em uma única tabela ou uma lista de chaves externas em outras tabelas que se refiram a uma única tabela.
SQLMoreResults	Determina se mais resultados estão pendentes em um identificador de instrução, <i>hstmt</i> , contendo instruções SELECT, UPDATE, INSERT ou DELETE e, em caso afirmativo, inicializa o processamento desses resultados. O Oracle suporta vários conjuntos de resultados somente a partir de procedimentos armazenados, quando utilizando seqüências de escape {resultset... }.
SQLNativeSql	Para obter informações sobre o uso, consulte <u>Retornando parâmetros de matriz a partir de procedimentos armazenados</u> .
SQLNumParams	Retorna o número de parâmetros numa instrução SQL. O número de parâmetros deve ser igual ao número de pontos de interrogação numa instrução SQL passados ao SQLPrepare .
SQLPrimaryKeys	Retorna os nomes de coluna que abrangem a chave primária para uma tabela.
SQLProcedureColumns	Retorna uma lista de parâmetros de entrada e de saída, o

valor de retorno e as colunas no conjunto de resultados de um único procedimento. Para procedimentos em pacote, a coluna PROCEDURE NAME está no formato *packagename.procedurename*.

SQLProcedures

Retorna uma lista de procedimentos na fonte de dados. Para procedimentos em pacote, a coluna PROCEDURE NAME está no formato *packagename.procedurename*.

Como o Oracle não fornece um meio de diferenciar procedimentos em pacote de funções em pacote, o driver retorna SQL_PT_UNKNOWN na coluna PROCEDURE_TYPE.

SQLSetPos

Define a posição do cursor em um conjunto de linhas. Você pode utilizar o **SQLSetPos** com o **SQLGetData** para recuperar linhas de colunas não-ligadas depois de posicionar o cursor em uma linha específica do conjunto de linhas. Linhas adicionadas ao conjunto de resultados utilizando fOption SQL_ADD são adicionadas depois da última coluna do conjunto de resultados.

SQLSetScrollOptions

Define opções que controlam o comportamento dos cursores associados a um identificador de instrução, *hstmt*. Para maiores detalhes, consulte Tabela de tipo de cursor e de combinações de concorrência suportada.

Tabela de opções de conexão

Estas opções permitem a personalização da conexão do banco de dados em um aplicativo.

<u>Opção de conexão</u>	<u>Observações</u>
SQL_AUTOCOMMIT	Se você escolher SQL_AUTOCOMMIT_OFF, seu aplicativo deverá confirmar ou repetir explicitamente as transações com SQLTransact .
SQL_ODBC_CURSORS	Este atributo de conexão é implementado no Gerenciador de drivers .
SQL_OPT_TRACE	Este atributo de conexão é implementado no Gerenciador de drivers .
SQL_OPT_TRACEFILE	Este atributo de conexão é implementado no Gerenciador de drivers .
SQL_TRANSLATE_DLL	Retorna o erro: "Driver sem capacidade".
SQL_TRANSLATE_OPTION	Um valor de 32 bits passado à DLL de conversão.
SQL_TXN_ISOLATION	O driver permite somente: SQL_TXN_READ_COMMITTED Os <i>vParams</i> a seguir não são suportados: SQL_TXN_READ_UNCOMMITTED SQL_TXN_REPEATABLE_READ SQL_TXN_SERIALIZABLE
SQL_ATTR_ENLIST_IN_DTC	Este atributo de conexão do ODBC 3.0 permite que você utilize o driver ODBC para Oracle nas Transações distribuídas coordenadas pelo Microsoft Transaction Server; fornece o Ponteiro de interface, pITransaction, para a transação, como o argumento vParam.
SQL_ATTR_CONNECTION_DEAD	Este atributo de conexão somente leitura do ODBC 3.5 permite que você determine se a conexão com o servidor Oracle falhou. Somente obter, não pode Definir.

Tabela de opções de instrução

Estas opções permitem a personalização de uma instrução de execução específica em um aplicativo.

Opções de instrução	Observações
SQL_BIND_TYPE	Não pode exceder 2.147.483.647 bytes ou a memória disponível.
SQL_CONCURRENCY	Para obter os valores permitidos, consulte <u>Tabela de tipo de cursor e de combinações de concorrência</u> suportada
SQL_CURSOR_TYPE	O driver não permite SQL_CURSOR_DYNAMIC. Consulte SQLSetScrollOptions para obter maiores informações. Para obter os valores permitidos consulte <u>Tabela de tipo de cursor e de combinações de concorrência</u> suportada
SQL_GET_BOOKMARK	Retorna um valor inteiro de 32 bits que é o indicador para o número de registro atual. Somente obter, não pode Definir.
SQL_KEYSET_SIZE	Só pode ser definido como 0.
SQL_MAX_ROWS	O número máximo de linhas a retornar de um conjunto de resultados.
SQL_ROW_NUMBER	Retorna um valor inteiro de 32 bits especificando a posição da linha atual dentro no conjunto de resultados. Somente obter, não pode Definir.
SQL_ROWSET_SIZE	Não pode exceder 4.294.967.296 linhas; contudo, você deve ter memória virtual suficiente em seu computador para manipular o que tiver pedido.
SQL_USE_BOOKMARKS	Suporta definir SQL_USE_BOOKMARKS como SQL_UB_ON e expõe indicadores de tamanho fixo..

Tabela de tipo de cursor e de combinações de concorrência

Os tipos de cursor controlam a funcionalidade do cursor fornecido ao usuário. As opções de concorrência controlam a capacidade de atualização e o comportamento de bloqueio de um conjunto de resultados.

<u>Tipo de cursor</u>	<u>Concorrência (valores permitidos)</u>
SQL_CURSOR_FORWARD_ONLY	SQL_CONCUR_READ_ONLY
SQL_CURSOR_STATIC	SQL_CONCUR_READ_ONLY
SQL_CURSOR_KEYSET_DRIVEN*	SQL_CONCUR_READ_ONLY SQL_CONCUR_LOCK** SQL_CONCUR_VALUES

Observações

* Consulte [Limitações do uso de cursores controlados por conjunto de teclas](#).

** O SQL_CONCUR_LOCK só é suportado quando a opção de conexão **SQL_AUTOCOMMIT** estiver definida como SQL_AUTOCOMMIT_OFF.

Mensagens de erro

Quando ocorre um erro, o driver Microsoft ODBC para Oracle retorna o SQLSTATE (um código de erro de ODBC) e uma mensagem de erro. O driver deriva esta informação tanto de erros detectados pelo driver quanto de erros retornados pelo Servidor Oracle.

Mensagens retornadas pelo driver ODBC para Oracle

Se houver uma mensagem de Erro do Oracle disponível, ela será retornada antecedida pelas marcas [Microsoft], [Driver ODBC para Oracle] e [Oracle]; caso contrário, a mensagem será retornada sem a marca [Oracle], como nos seguintes exemplos:

Mensagem de erro do Oracle:

```
[Microsoft][ODBC driver for Oracle][Oracle]ORA-nnnnn message-text
```

Mensagem de erro do Driver ODBC Oracle

```
[Microsoft][ODBC driver for Oracle]
```

