Informix Product Family
Informix Client Software Development Kit
Version 3.70

IBM Informix OLE DB Provider Programmer's Guide

SC27-3549-01

Informix Product Family
Informix Client Software Development Kit
Version 3.70

IBM Informix OLE DB Provider Programmer's Guide

SC27-3549-01

Note

Before using this information and the product it supports, read the information in “Notices” on page B-1.

This edition replaces SC27-3549-00.

Contents

Introduction

This introduction provides an overview of the information in this publication and describes the conventions it uses.

About this publication

This publication describes the software requirements for using IBM® Informix® OLE DB Provider, shows how to install and configure the provider for your use, and explains how to use IBM Informix OLE DB Provider to enable client applications, such as ActiveX Data Object (ADO) applications and web pages, to access data on an Informix server.

New editions and product names

IBM Informix Dynamic Server editions were withdrawn and new Informix editions are available. Some products were also renamed. The publications in the Informix library pertain to the following products:

- IBM Informix database server, formerly known as IBM Informix Dynamic Server (IDS)
- IBM OpenAdmin Tool (OAT) for Informix, formerly known as OpenAdmin Tool for Informix Dynamic Server (IDS)
- IBM Informix SQL Warehousing Tool, formerly known as Informix Warehouse Feature

For more information about the Informix product family, go to http://www.ibm.com/software/data/informix/.

Types of users

This publication is written for the following users:

- Database administrators who install and configure IBM Informix database servers, databases, and connectivity products
- Developers who write applications using IBM Informix OLE DB Provider

This publication is written with the assumption that you have the following background:

- A working knowledge of your computer, your operating system, and the utilities that your operating system provides
- Some experience with Microsoft OLE DB
- Some experience working with relational databases or exposure to database concepts

You can access the Informix information centers and other technical information such as technotes, white papers, and IBM Redbooks® publications online at http://www.ibm.com/software/data/sw-library/.

Software compatibility

For information about software compatibility, see the IBM Informix OLE DB Provider release notes.

System requirements

To use the IBM Informix OLE DB Provider, you must run on one of these supported operating systems:

- Microsoft Windows NT Version 4.0 with Service Pack 4 or later
- Microsoft Windows 2003 Server, Microsoft Windows XP, or Microsoft Windows Vista

Assumptions about your locale

IBM Informix products can support many languages, cultures, and code sets. All the information related to character set, collation and representation of numeric data, currency, date, and time that is used by a language within a given territory and encoding is brought together in a single environment, called a Global Language Support (GLS) locale.

The IBM Informix OLE DB Provider follows the ISO string formats for date, time, and money, as defined by the Microsoft OLE DB standards. You can override that default by setting an Informix environment variable or registry entry, such as **DBDATE**.

If you use Simple Network Management Protocol (SNMP) in your Informix environment, note that the protocols (SNMPv1 and SNMPv2) recognize only English code sets. For more information, see the topic about GLS and SNMP in the *IBM Informix SNMP Subagent Guide*.

The examples in this publication are written with the assumption that you are using one of these locales: en_us.8859-1 (ISO 8859-1) on UNIX platforms or en_us.1252 (Microsoft 1252) in Windows environments. These locales support U.S. English format conventions for displaying and entering date, time, number, and currency values. They also support the ISO 8859-1 code set (on UNIX and Linux) or the Microsoft 1252 code set (on Windows), which includes the ASCII code set plus many 8-bit characters such as é, è, and ñ.

You can specify another locale if you plan to use characters from other locales in your data or your SQL identifiers, or if you want to conform to other collation rules for character data.

For instructions about how to specify locales, additional syntax, and other considerations related to GLS locales, see the *IBM Informix GLS User's Guide*.

Example code conventions

Examples of SQL code occur throughout this publication. Except as noted, the code is not specific to any single IBM Informix application development tool.

If only SQL statements are listed in the example, they are not delimited by semicolons. For instance, you might see the code in the following example:

```
CONNECT TO stores_demo
...
```

```
DELETE FROM customer
   WHERE customer_num = 121
...

COMMIT WORK
DISCONNECT CURRENT
```

To use this SQL code for a specific product, you must apply the syntax rules for that product. For example, if you are using an SQL API, you must use EXEC SQL at the start of each statement and a semicolon (or other appropriate delimiter) at the end of the statement. If you are using DB–Access, you must delimit multiple statements with semicolons.

Tip: Ellipsis points in a code example indicate that more code would be added in a full application, but it is not necessary to show it to describe the concept being discussed.

For detailed directions on using SQL statements for a particular application development tool or SQL API, see the documentation for your product.

Additional documentation

Documentation about this release of IBM Informix products is available in various formats.

You can access or install the product documentation from the Quick Start CD that is shipped with Informix products. To get the most current information, see the Informix information centers at ibm.com®. You can access the information centers and other Informix technical information such as technotes, white papers, and IBM Redbooks publications online at http://www.ibm.com/software/data/sw-library/.

Compliance with industry standards

IBM Informix products are compliant with various standards.

IBM Informix SQL-based products are fully compliant with SQL-92 Entry Level (published as ANSI X3.135-1992), which is identical to ISO 9075:1992. In addition, many features of IBM Informix database servers comply with the SQL-92 Intermediate and Full Level and X/Open SQL Common Applications Environment (CAE) standards.

The IBM Informix Geodetic DataBlade® Module supports a subset of the data types from the *Spatial Data Transfer Standard (SDTS)—Federal Information Processing Standard 173*, as referenced by the document *Content Standard for Geospatial Metadata*, Federal Geographic Data Committee, June 8, 1994 (FGDC Metadata Standard).

How to provide documentation feedback

You are encouraged to send your comments about IBM Informix user documentation.

Use one of the following methods:

- Send e-mail to docinf@us.ibm.com.

- In the Informix information center, which is available online at http://www.ibm.com/software/data/sw-library/, open the topic that you want to comment on. Click the feedback link at the bottom of the page, fill out the form, and submit your feedback.
- Add comments to topics directly in the information center and read comments that were added by other users. Share information about the product documentation, participate in discussions with other users, rate topics, and more!

Feedback from all methods is monitored by the team that maintains the user documentation. The feedback methods are reserved for reporting errors and omissions in the documentation. For immediate help with a technical problem, contact IBM Technical Support. For instructions, see the IBM Informix Technical Support website at http://www.ibm.com/planetwide/.

We appreciate your suggestions.

Chapter 1. Overview and setup

These topics describe the software you can use with IBM Informix OLE DB Provider and explain how to install and configure it for your use.

Introduction to IBM Informix OLE DB Provider

Microsoft OLE DB is a specification for a set of data access interfaces designed to enable various data stores to work together. OLE DB components are *data providers*, *data consumers*, and service components. Data providers own data and make it available to consumers. Each implementation of a provider is different, but they all expose their data in a tabular form through virtual tables. Data consumers use the OLE DB interfaces to access data.

You use IBM Informix OLE DB Provider to enable client applications, such as ActiveX Data Object (ADO) applications and web pages, to access data on an Informix server.

Informix OLE DB is a component of the IBM Informix Client Software Development Kit.

Tip: This publication describes the characteristics of the IBM Informix OLE DB Provider. It does not describe the architecture of OLE DB providers in general or how to program with OLE DB. For information about OLE DB architecture and programming, go to the Microsoft website (http://www.microsoft.com) and search for "Introduction to OLE DB".

Install and configure Informix OLE DB Provider

IBM Informix OLE DB Provider is distributed with IBM Informix Connect and the IBM Informix Client Software Development Kit (Client SDK).

When you install the Client SDK, IBM Informix OLE DB Provider is installed by default. The installation includes other necessary components and performs required updates to the registry.

After installation, you must run the script `coledbp.sql` on the Informix server against the **sysmaster** database as user **informix**. Informix OLE DB Provider requires the stored procedures added to the server by the `coledbp.sql` script. The script is located in the `INFORMIXDIR\etc` directory. (To remove the stored procedures, you can run the `doledbp.sql` against the **sysmaster** database as user **informix**.)

Manually adding IBM Informix OLE DB Provider to the registry

If you must manually add IBM Informix OLE DB Provider to the registry, you can do it as follows:

1. Using the command prompt, change directory to `INFORMIXDIR\bin`. The file `ifxoledbc.dll` is present in this directory.
2. Type `Regsvr32.exe ifxoledbc.dll` and press **Enter**.

Manually removing IBM Informix OLE DB Provider from the registry

If you must remove IBM Informix OLE DB Provider from the registry, you can do it as follows.

1. Using the command prompt, change directory to `INFORMIXDIR\bin`. The file `ifxoledbc.dll` is present in this directory.
2. Type `Regsvr32.exe /u ifxoledbc.dll` and press **Enter**.

Upgrading from previous versions

To upgrade from previous versions of IBM Informix OLE DB Provider, your database administrator must follow these steps:

1. Run the script `doledbp.sql` against the **sysmaster** database as user **informix**. Ignore any messages about missing database objects.
2. Run the script `coledbp.sql` against the **sysmaster** database as user **informix**.

Version 2.8 of IBM Informix OLE DB Provider changes the way some features used to operate in earlier versions. If you have used a version before 2.8 of the IBM Informix OLE DB Provider, the issues you must be aware of are:

- OLE DB Provider handles the INTERVAL type differently in this release. In versions before 2.8, interval data was returned as decimal numbers with different sections of that number corresponding to year, month, day, and other fields within the value. In order to interpret the decimal number correctly. knowledge of the start and end fields of the interval column was required .

  In this release, the default type is a string with the format as described in the *IBM Informix Guide to SQL: Reference*, with the provision that a conversion to a numeric type is also allowed. If a datetime interval is requested in DB_TYPE_I8 format, the number returned will have 1/100,000 seconds as the unit of measure.
- OLE DB Provider handles complex data types, collections, and row types differently in this release. Data of these types is presented in string format as LVARCHAR data. This method is similar to the method that is used to interact with this data by using the DB-Access tool. As an example of the format being presented, if a column has the definition:

  ```
  my_date     MULTISET(date not null)
  ```

  Data contained in this column is returned to the application in the format:

  ```
  MULTISET{'08/15/2000','02/02/2002','10/11/1999'}
  ```
- Type handling for the DECIMAL, MONEY, and DATETIME types has changed from the previous release.

Related reference

"Data types" on page 2-3

Sample programs

A sample program, **Demo1**, is included in `%INFORMIXDIR%\demo\oledbdemo\Demo1`.

The sample program is a complete project that introduces how to use OLE DB interfaces in a C++ application. It performs the following tasks:

- Connects to IBM Informix by creating a DataSource object
- Creates a Session object
- Creates a Command object

- Executes SQL statements to perform the following tasks:
  - Drop the table **MyTable**, if it exists
  - Create the table **MyTable**
  - Insert records in **MyTable**
- Deletes the Command object
- Deletes the Session object
- Disconnects the database and server connection and deletes the DataSource object

Another sample program is included in `INFORMIXDIR\demo\oledbdemo\DistTxn\`.

Support of OLE DB specifications

The IBM Informix OLE DB Provider supports level 0 of the OLE DB provider specification, including some additional level 1 interfaces.

The IBM Informix OLE DB Provider is built and tested with Microsoft Data Access Components (MDAC) version 2.8.

Related reference

"Supported interfaces"

Support of LDAP Authentication in Windows

You can use LDAP Authentication in Windows with IBM Informix OLE DB Provider, which is similar to the Pluggable Authentication Module (PAM) that is used on UNIX and Linux.

When you want to use an LDAP server to authenticate your system users, use the LDAP Authentication Support module. The module contains source code that you can modify to fit your specific requirements. For information about the LDAP Authentication Support module, see the *IBM Informix Security Guide*.

Supported interfaces

The following interfaces are implemented by IBM Informix OLE DB Provider:

- IAccessor
- IColumnsInfo
- IColumnsRowset
- ICommand
- ICommandPrepare
- ICommandProperties
- ICommandText
- ICommandWithParameters
- IConvertType
- IDBAsynchStatus
- IDBCreateCommand
- IDBCreateSession
- IDBDataSourceAdmin
- IDBInfo

- IDBInitialize
- IDBProperties
- IDBSchemaRowset
- IErrorLookup
- IGetDataSource
- IIndexDefinition
- IOpenRowset
- IPersist
- IRowsetFind
- IRowsetIdentity
- IRowsetInfo
- IRowsetLocate
- IRowsetScroll
- IRowsetUpdate
- ISessionProperties
- ISupportErrorInfo
- ITableDefinition
- ITransactionJoin
- ITransactionLocal
- ITransactionOptions

Related reference

"Support of OLE DB specifications" on page 1-3

Chapter 2. IBM Informix OLE DB Provider

These topics describe the applications you can create using IBM Informix OLE DB Provider and describe how to connect to data sources and manipulate data within your application.

Supported applications

With the IBM Informix OLE DB Provider, you can create the ADO, C, and C++ applications.

You can create the following types of applications:

- ADO applications, including:

  Microsoft Visual Studio C++ applications

  Microsoft Visual Basic applications
- C and C++ applications that access Informix databases directly using the OLE DB interfaces, including ATL applications whose Data Access Consumer Objects were generated by the ATL COM AppWizard

Related reference

"Connect to a data source"

Connect to a data source

IBM Informix OLE DB Provider treats the database (rather than the database server instance) as a data source.

Data source names must be in the following format:

```
[database][@server]
```

The brackets indicate that the enclosed items are optional. If the database name is missing, the client user name is used. If the *@server* name is missing, the default database server is used (corresponding to the value specified by the INFORMIXSERVER registry entry of the client).

To specify ADO connection string keywords, specify keywords in the connection string for the Provider by using the format keyword=*value*. Delimit multiple keywords with a semicolon.

The following table describes the ADO keywords supported by the IBM Informix OLE DB Provider.

Important: These settings take precedence over the settings of environment variables.

Keyword	Value	Description
DSN	Name of the database alias	The Informix database alias in the database directory
UID	User ID	The user ID used to connect to the Informix server
PWD	Password	The password for the user ID
Client_locale	Locale	The client locale for the application

Keyword	Value	Description
Db_locale	Locale	The database locale for the application
UNICODE	True or False	Indicates whether to use IBM Informix GLS Unicode.
RSASWS or REPORTSTRINGASWSTRING	True or False	Enables you to control the data mapping for wide strings.
FBS or FETCHBUFFERSIZE	Numeric	The size in bytes of the buffer size used to send data to or from the database. The range of values is 4096 (default) to 32767. If you want to set the fetch buffer size at 32 K, for example, set the connection string as "FBS=32767" or "FETCHBUFFERSIZE=32767". If the value of "FBS" or "FETCHBUFFERSIZE" is not in the range 4096 - 32767, then by default the value will be changed to 4096 internally and no error message is returned.

Related reference

"Supported applications" on page 2-1

"The UNICODE provider string keyword" on page 2-25

"The REPORTSTRINGASWSTRING provider string keyword" on page 2-26

"Database not found" on page 2-28

"Attempt to use provider from web server or other server fails" on page 2-29

Cursors

IBM Informix OLE DB Provider supports the following ADO cursor types:

- Client-side scrollable cursors (**adUseClient** and **adOpenStatic**)

  Client-side scrollable cursors (**adUseClient** and **adOpenStatic**) support bookmarks and have the following limitation: database updates fail when the rowset includes columns of extended data types.
- Server-side scrollable cursors (**adOpenStatic**)

  Server-side scrollable cursors are faster than client-side cursors. If a server-side scrollable cursor is opened on a table (**adCmdTableDirect**) or on a simple SELECT statement (single table, no aggregates, no GROUP BY clause), the cursor can support bookmarks and, with the Version 9.2, or later IBM Informix server, database updates.
- Server-side nonscrollable cursors (**adUseServer** and **adOpenForwardOnly**)

  Server-side nonscrollable cursors (**adUseServer** and **adOpenForwardOnly**) are the fastest cursors. Like server-side scrollable cursors, nonscrollable cursors support updates when opened on a table or (with the Version 9.2, or later IBM Informix server) when opened on a simple FOR UPDATE-compatible SELECT statement. In addition, if a server side nonscrollable cursor is opened on a table or on a simple SQL statement without an ORDER BY clause, the cursor is able to display changes made to the database by other users (unless transaction isolation precludes it).

The following caveats apply to the use of cursors:

- The only scrollable cursor supported by IBM Informix OLE DB Provider is the static cursor. The Provider accepts requests for other types of scrollable cursors (dynamic and keyset), but it supplies a static cursor regardless of which cursor type is requested.

- Since the scrollable cursor is static, it cannot detect changes made to the database by other users. The DBPROP_OWNINSERT, DBPROP_OTHERINSERT, and DBPROP_OTHERUPDATEDELETE properties for scrollable cursors are read-only VARIANT_FALSE.

  Use a nonscrollable cursor (**adOpenForwardOnly**) if you want the functionality that corresponds to setting these properties to VARIANT_TRUE.
- With IBM Informix servers before version 9.2, the server-side nonscrollable cursor **adUseServer** can update records only when the rowset is opened with **IOpenRowset::OpenRowset()**. The ADO flag corresponding to **IOpenRowset::OpenRowset()** is **adCmdTableDirect**.

  The client-side cursor (**adUseClient**) does not have this limitation.
- Server-side scrollable cursors cannot be opened if the record set includes simple large objects (BYTE and TEXT) or collections.

  You can use a server-side nonscrollable cursor (**adOpenForwardOnly**) or a client-side scrollable cursor (**adUseClient**) with these types.
- The DBPROP_IRowsetScroll property is read-only VARIANT_FALSE for rowsets not opened with **IOpenRowset::OpenRowset()**. It is VARIANT_TRUE for rowsets opened with **IOpenRowset::OpenRowset()** if bookmarks are requested (corresponding ADO flags are **adOpenStatic** and **adOpenKeyset**).
- To support bookmarks and the modification or deletion of records, a data source table must include a ROWID column. (A ROWID column is not needed to insert records.)

  All fragmented and nonfragmented tables created with the WITH ROWIDS clause (or altered with the WITH ROWIDS clause applied) have this column. The ROWID column itself is not visible to consumers unless it is explicitly selected.

  If consumers require a persistent unique ID, create the necessary columns by using the SERIAL, SERIAL8, or BIGSERIAL data types.
- Use of DISTINCT, UNIQUE, ORDER BY, GROUP BY, or aggregates in SQL statements makes the cursor unable to detect changes made on the database by other users.
- Any SELECT statement that cannot be used with FOR UPDATE (for example, because it has joins or aggregates) is incompatible with bookmarks and updatability (but not incompatible with scrolling).
- When you work with ADO client-side cursors, specify the table name in the same text case that is used on the server. Otherwise, the database server returns an error. To work around this issue, use ADO server-side cursors.

Data types

IBM Informix OLE DB Provider supports all built-in and user-defined types. However, see the caveats about using scrollable cursors on data that includes simple large objects and collections in “Cursors” on page 2-2.

Related tasks

“Upgrading from previous versions” on page 1-2

Data type mappings

IBM Informix OLE DB Provider supports data type mappings between Informix data types and OLE DB data types.

The support for data type mappings between Informix and OLE DB data types are shown in the following table.

The data type shown in the column headed **MSDASQL>ODBC 3.80 Type** is the type that an Informix data type maps to when you use the Microsoft OLE DB to ODBC bridge.

Informix data type	OLE DB Provider type before version 2.8	MSDASQL>ODBC 3.80 type	Current OLE DB Provider type
BIGINT	None	DBTYPE_I8	DBTYPE_I8
BIGSERIAL	None	DBTYPE_UI8	DBTYPE_I8
BLOB	DBTYPE_BYTES	DBTYPE_BYTES	DBTYPE_BYTES
BOOLEAN	DBTYPE_BOOL	DBTYPE_BOOL	DBTYPE_BOOL
BYTE	DBTYPE_BYTES	DBTYPE_BYTES	DBTYPE_BYTES
CHAR	DBTYPE_STR	DBTYPE_STR	DBTYPE_STR
CLOB	DBTYPE_STR	DBTYPE_STR	DBTYPE_STR
DATE	DBTYPE_DBDATE	DBTYPE_DBDATE	DBTYPE_DBDATE
DATETIME	DBTYPE_DBTIMESTAMP	DBTYPE_DBTIMESTAMP Except: DATETIME YEAR TO DAY maps to DBTYPE_DBDATE DATETIME HOUR TO SECOND maps to DBTYPE_DBTIME	DBTYPE_DBDATE or DBTYPE_DBTIME or DBTYPE_DBTIMESTAMP
DECIMAL	DBTYPE_VARNUMERIC	DBTYPE_NUMERIC	DBTYPE_NUMERIC
DISTINCT	Same as underlying type	Same as underlying type	Same as underlying type
FLOAT	DBTYPE_R8	DBTYPE_R8	DBTYPE_R8
INT8	DBTYPE_I8	DBTYPE_I8	DBTYPE_I8
INTEGER	DBTYPE_I4	DBTYPE_I4	DBTYPE_I4
INTERVAL	DBTYPE_NUMERIC	DBTYPE_BYTES	DBTYPE_STR
LIST	DBTYPE_VARIANT	DBTYPE_STR	DBTYPE_VARIANT
LVARCHAR	DBTYPE_STR	DBTYPE_STR	DBTYPE_STR
MONEY (p<=19 s<=4)	DBTYPE_NUMERIC	DBTYPE_CY	DBTYPE_CY
MONEY (p>19 s<>4)	DBTYPE_NUMERIC	DBTYPE_NUMERIC	DBTYPE_NUMERIC
MULTISET	DBTYPE_VARIANT	DBTYPE_STR	DBTYPE_VARIANT
NCHAR	DBTYPE_STR	DBTYPE_STR	DBTYPE_STR
OPAQUE	DBTYPE_BYTES	DBTYPE_BYTES	DBTYPE_BYTES
Named ROW	DBTYPE_VARIANT	DBTYPE_STR	DBTYPE_VARIANT
Unnamed ROW	Same as underlying type	DBTYPE_STR	DBTYPE_VARIANT
SERIAL	DBTYPE_I4	DBTYPE_I4	DBTYPE_I4
SERIAL8	DBTYPE_I8	DBTYPE_UI8	DBTYPE_I8
SET	DBTYPE_VARIANT	DBTYPE_STR	DBTYPE_VARIANT
SMALLFLOAT	DBTYPE_R4	DBTYPE_R4	DBTYPE_R4
SMALLINT	DBTYPE_I2	DBTYPE_I2	DBTYPE_I2

Informix data type	OLE DB Provider type before version 2.8	MSDASQL>ODBC 3.80 type	Current OLE DB Provider type
TEXT	DBTYPE_STR	DBTYPE_STR	DBTYPE_STR
VARCHAR	DBTYPE_STR	DBTYPE_STR	DBTYPE_STR

Related reference

"The INTERVAL type mapping"

"The DATETIME type mapping"

"The decimal and money type mapping" on page 2-6

The INTERVAL type mapping

For Version 2.8, by default, the INTERVAL data type is mapped to a string with the format described in *IBM Informix Guide to SQL: Reference*. However, a conversion to a numeric type is also allowed. Conversion to a string type facilitates the easy display of user-entered data. The alternate numeric conversion facilitates mathematical manipulation of data by an application.

For day-time intervals, the recommended alternate numeric conversion is to DBTYPE_I8. The number returned has 1/100,000 seconds as the unit of measure.

For year-month intervals, the recommended alternate conversion is to DBTYPE_I8. The number returned has months as the unit of measure.

Conversions of both day-time and year-month interval types to the types DBTYPE_I4, DBTYPE_I2, and DBTYPE_I1 are also allowed; overflow errors are possible for the smaller types.

Important: For numeric conversions, the format of the returned number is different from the format returned in prior releases of the IBM Informix OLE DB Provider (in order to avoid ambiguity).

Related reference

"Data type mappings" on page 2-3

The DATETIME type mapping

Version 2.8 of IBM Informix OLE DB Provider maps DATETIME types to the smallest type that can contain the start and end fields of the DATETIME value. The following lists show how Version 2.8 IBM Informix OLE DB Provider maps to each DATETIME type.

IBM Informix OLE DB Provider DBTYPE_DBDATE is the mapped type for:

- DATETIME YEAR TO YEAR
- DATETIME YEAR TO MONTH
- DATETIME YEAR TO DAY
- DATETIME MONTH TO MONTH
- DATETIME MONTH TO DAY
- DATETIME DAY TO DAY

IBM Informix OLE DB Provider DBTYPE_DBTIME is the mapped type for:

- DATETIME HOUR TO HOUR

- DATETIME HOUR TO MINUTE
- DATETIME HOUR TO SECOND
- DATETIME MINUTE TO MINUTE
- DATETIME MINUTE TO SECOND
- DATETIME SECOND TO SECOND

IBM Informix OLE DB Provider DBTYPE_DBTIMESTAMP is the mapped type for:

- DATETIME YEAR TO HOUR
- DATETIME YEAR TO MINUTE
- DATETIME YEAR TO SECOND
- DATETIME YEAR TO FRACTION
- DATETIME MONTH TO HOUR
- DATETIME MONTH TO MINUTE
- DATETIME MONTH TO SECOND
- DATETIME MONTH TO FRACTION
- DATETIME DAY TO HOUR
- DATETIME DAY TO MINUTE
- DATETIME DAY TO SECOND
- DATETIME DAY TO FRACTION
- DATETIME HOUR TO FRACTION
- DATETIME MINUTE TO FRACTION
- DATETIME SECOND TO FRACTION
- DATETIME FRACTION TO FRACTION

Related reference

"Data type mappings" on page 2-3

The decimal and money type mapping

Microsoft Visual Basic and ADO have limitations when handling floating point numbers with a scale greater than 30 and decimals with an undefined scale. Therefore, some ADO consumers (for example, Microsoft Visual Basic 6) might encounter problems representing IBM Informix DECIMAL or MONEY values.

ADO allows you to specify DBPROP_INIT_PROVIDERSTRING parameters as part of the connection string. Some tools (for example, Microsoft Visual Basic 6) allow you to set DBPROP_INIT_PROVIDERSTRING parameters as "Extended Properties." The parameters are case-sensitive.

To allow these consumers to correctly handle decimal values, IBM Informix OLE DB Provider sets the advanced connection option Describe Decimal as Real/Double, so that decimal values with no scale are returned as the type DBTYPE_R8.

To avoid the problem of floating point numbers with a scale greater than 30, IBM Informix OLE DB Provider supplies the provider string option **decasr8=R8**, which you specify by setting the DBPROP_INIT_PROVIDERSTRING initialization property. This parameter instructs IBM Informix OLE DB Provider to map DECIMAL and MONEY values to the standard Windows DBTYPE_R8 data type. This option also resolves the decimals-with-no-scale problem, but can lead to unnecessary truncation of digits.

Starting with version 3.00, when the connection option (**decasr8=R8**) is not used for columns with DECIMAL data type and no scale is specified, the precision and scale are evaluated by the OLE DB Provider with the following formula for the non-ANSI databases:

```
DECIMAL(p) = DECIMAL(MIN(2 * p, 32),(p < 16) ? p : 12 + ((32 - p) / 4))
```

For best results, always specify a scale for DECIMAL data types.

Related reference

“Data type mappings” on page 2-3

Large object and user-defined data type mapping

IBM Informix OLE DB Provider supports large objects and user-defined data types as follows:

- The BYTE data type is reported by **IColumnsInfo::GetColumnInfo()** and appropriate schema rowsets as DBTYPE_BYTES; the TEXT data type is reported as DBTYPE_STR. Values of BYTE and TEXT types are cached in memory.
- Complex data types are reported by **IColumnsInfo::GetColumnInfo()** and appropriate schema rowsets as DBTYPE_VARIANT. The corresponding value is a safe array.

  This mapping is known to work with ADO and Visual Basic/VBScript.
- The CLOB data type is reported by **IColumnsInfo::GetColumnInfo()** and appropriate schema rowsets as DBTYPE_STR with the IS_LONG flag set; the BLOB data type is reported as DBTYPE_BYTES with the IS_LONG flag set.

  This mapping allows ADO to open storage objects on smart large object data and manipulate it with the **GetChunk()** and **AppendChunk()** methods.
- Distinct data types are generally resolved to their source type. For example, if you define an HTML type as a distinct CLOB data type, **IColumnsInfo::GetColumnInfo()** and appropriate schema rowsets report it as DBTYPE_STR with the IS_LONG flag set.
- Opaque data types are reported by **IColumnsInfo::GetColumnInfo()** and appropriate schema rowsets as DBTYPE_BYTES.

Data conversions from OLE DB types to IBM Informix types

The following tables show the supported data conversions from OLE DB types to IBM Informix types:

- OLE DB datatype compatibility with SMALLINT, INTEGER, INT8, and BIGINT.
- OLE DB datatype compatibility with SERIAL, SERIAL8, and BIGSERIAL.
- OLE DB datatype compatibility with NUMERIC, DECIMAL, FLOAT, and SMALLFLOAT.
- OLE DB datatype compatibility with CHAR, NCHAR, VARCHAR, NVARCHAR, and LVARCHAR.
- OLE DB datatype compatibility with MONEY, DATE, DATETIME, and INTERVAL.
- OLE DB datatype compatibility with CLOB, BLOB, and ROW.
- OLE DB datatype compatibility with SET, MULTISET, and LIST.

Truncation of data might occur in some cases.

The 32 K LVARCHAR feature extends LVARCHAR columns to hold up to 32 K bytes of data. This feature requires IBM Informix side support for 32 K LVARCHAR, and only works with IBM Informix Version 9.4 or later.

Important:

- All the OLE DB types that are allowed with one or more of the Informix data types for the DBTYPE_BYREF type are also allowed when combined with DBTYPE_BYREF.
- For DBTYPE_IUNKNOWN, the supported interfaces are ISequentialStream, IStream, and ILockBytes.

Table 2-1. OLE DB datatype compatibility with SMALLINT, INTEGER, INT8, and BIGINT.

	SMALLINT	INTEGER	INT8	BIGINT
DBTYPE_EMPTY	NO	NO	NO	NO
DBTYPE_NULL	NO	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO	NO
DBTYPE_I1	YES	YES	YES	YES
DBTYPE_I2	YES	YES	YES	YES
DBTYPE_I4	YES	YES	YES	YES
DBTYPE_I8	NO	NO	NO	NO
DBTYPE_UI1	YES	YES	YES	YES
DBTYPE_UI2	YES	YES	YES	YES
DBTYPE_UI4	YES	YES	YES	YES
DBTYPE_UI8	NO	NO	NO	NO
DBTYPE_R4	YES	YES	YES	YES
DBTYPE_R8	YES	YES	YES	YES
DBTYPE_CY	YES	YES	YES	YES
DBTYPE_DECIMAL	YES	YES	YES	YES
DBTYPE_NUMERIC	YES	YES	YES	YES
DBTYPE_DATE	YES	YES	YES	YES
DBTYPE_BOOL	YES	YES	YES	YES
DBTYPE_BYTES	YES	YES	YES	YES
DBTYPE_BSTR	YES	YES	YES	YES
DBTYPE_STR	YES	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES	YES
DBTYPE_VARIANT	YES	YES	YES	YES
DBTYPE_IDISPATCH	NO	NO	NO	NO
DBTYPE_IUNKNOWN	NO	NO	NO	NO
DBTYPE_GUID	NO	NO	NO	NO
DBTYPE_ERROR	NO	NO	NO	NO
DBTYPE_BYREF	*	*	*	*
DBTYPE_ARRAY	NO	NO	NO	NO
DBTYPE_VECTOR	NO	NO	NO	NO
DBTYPE_UDT	NO	NO	NO	NO

Table 2-1. OLE DB datatype compatibility with SMALLINT, INTEGER, INT8, and BIGINT. (continued)

	SMALLINT	INTEGER	INT8	BIGINT
DBTYPE_DBDATE	NO	NO	NO	NO
DBTYPE_DBTIME	NO	NO	NO	NO
DBTYPE_DBTIMESTAMP	NO	NO	NO	NO
DBTYPE_FILETIME	NO	NO	NO	NO
DBTYPE_PROP_VARIANT	YES	YES	YES	YES
DBTYPE_HCHAPTER	NO	NO	NO	NO
DBTYPE_VARNUMERIC	YES	YES	YES	YES

Table 2-2. OLE DB datatype compatibility with SERIAL, SERIAL8, and BIGSERIAL.

	SERIAL	SERIAL8	BIGSERIAL
DBTYPE_EMPTY	NO	NO	NO
DBTYPE_NULL	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO
DBTYPE_I1	YES	YES	YES
DBTYPE_I2	YES	YES	YES
DBTYPE_I4	YES	YES	YES
DBTYPE_I8	NO	NO	NO
DBTYPE_UI1	YES	YES	YES
DBTYPE_UI2	YES	YES	YES
DBTYPE_UI4	YES	YES	YES
DBTYPE_UI8	NO	NO	NO
DBTYPE_R4	YES	YES	YES
DBTYPE_R8	YES	YES	YES
DBTYPE_CY	YES	YES	YES
DBTYPE_DECIMAL	YES	YES	YES
DBTYPE_NUMERIC	YES	YES	YES
DBTYPE_DATE	YES	YES	YES
DBTYPE_BOOL	YES	YES	YES
DBTYPE_BYTES	YES	YES	YES
DBTYPE_BSTR	YES	YES	YES
DBTYPE_STR	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES
DBTYPE_VARIANT	YES	YES	YES
DBTYPE_IDISPATCH	NO	NO	NO
DBTYPE_IUNKNOWN	NO	NO	NO
DBTYPE_GUID	NO	NO	NO
DBTYPE_ERROR	NO	NO	NO
DBTYPE_BYREF	*	*	*
DBTYPE_ARRAY	NO	NO	NO
DBTYPE_VECTOR	NO	NO	NO

Table 2-2. OLE DB datatype compatibility with SERIAL, SERIAL8, and BIGSERIAL. (continued)

	SERIAL	SERIAL8	BIGSERIAL
DBTYPE_UDT	NO	NO	NO
DBTYPE_DBDATE	NO	NO	NO
DBTYPE_DBTIME	NO	NO	NO
DBTYPE_DBTIMESTAMP	NO	NO	NO
DBTYPE_FILETIME	NO	NO	NO
DBTYPE_PROP_VARIANT	YES	YES	YES
DBTYPE_HCHAPTER	NO	NO	NO
DBTYPE_VARNUMERIC	YES	YES	YES

Table 2-3. OLE DB datatype compatibility with NUMERIC, DECIMAL, FLOAT, and SMALLFLOAT.

	NUMERIC	DECIMAL	FLOAT	SMALLFLOAT
DBTYPE_EMPTY	NO	NO	NO	NO
DBTYPE_NULL	NO	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO	NO
DBTYPE_I1	YES	YES	YES	YES
DBTYPE_I2	YES	YES	YES	YES
DBTYPE_I4	YES	YES	YES	YES
DBTYPE_I8	NO	NO	NO	NO
DBTYPE_UI1	YES	YES	YES	YES
DBTYPE_UI2	YES	YES	YES	YES
DBTYPE_UI4	YES	YES	YES	YES
DBTYPE_UI8	NO	NO	NO	NO
DBTYPE_R4	YES	YES	YES	YES
DBTYPE_R8	YES	YES	YES	YES
DBTYPE_CY	YES	YES	YES	YES
DBTYPE_DECIMAL	YES	YES	YES	YES
DBTYPE_NUMERIC	YES	YES	YES	YES
DBTYPE_DATE	YES	YES	YES	YES
DBTYPE_BOOL	YES	YES	YES	YES
DBTYPE_BYTES	YES	YES	YES	YES
DBTYPE_BSTR	YES	YES	YES	YES
DBTYPE_STR	YES	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES	YES
DBTYPE_VARIANT	YES	YES	YES	YES
DBTYPE_IDISPATCH	NO	NO	NO	NO
DBTYPE_IUNKNOWN	NO	NO	NO	NO
DBTYPE_GUID	NO	NO	NO	NO
DBTYPE_ERROR	NO	NO	NO	NO
DBTYPE_BYREF	*	*	*	*
DBTYPE_ARRAY	NO	NO	NO	NO

Table 2-3. OLE DB datatype compatibility with NUMERIC, DECIMAL, FLOAT, and SMALLFLOAT. (continued)

	NUMERIC	DECIMAL	FLOAT	SMALLFLOAT
DBTYPE_VECTOR	NO	NO	NO	NO
DBTYPE_UDT	NO	NO	NO	NO
DBTYPE_DBDATE	NO	NO	NO	NO
DBTYPE_DBTIME	NO	NO	NO	NO
DBTYPE_DBTIMESTAMP	NO	NO	NO	NO
DBTYPE_FILETIME	NO	NO	NO	NO
DBTYPE_PROP_VARIANT	YES	YES	YES	YES
DBTYPE_HCHAPTER	NO	NO	NO	NO
DBTYPE_VARNUMERIC	YES	YES	YES	YES

Table 2-4. OLE DB datatype compatibility with CHAR, NCHAR, VARCHAR, NVARCHAR, and LVARCHAR.

	CHAR	NCHAR	VARCHAR	NVARCHAR	LVARCHAR
DBTYPE_EMPTY	NO	NO	NO	NO	NO
DBTYPE_NULL	NO	NO	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO	NO	NO
DBTYPE_I1	YES	YES	YES	YES	YES
DBTYPE_I2	YES	YES	YES	YES	YES
DBTYPE_I4	YES	YES	YES	YES	YES
DBTYPE_I8	YES	YES	YES	YES	NO
DBTYPE_UI1	YES	YES	YES	YES	YES
DBTYPE_UI2	YES	YES	YES	YES	YES
DBTYPE_UI4	YES	YES	YES	YES	YES
DBTYPE_UI8	YES	YES	YES	YES	NO
DBTYPE_R4	YES	YES	YES	YES	YES
DBTYPE_R8	YES	YES	YES	YES	YES
DBTYPE_CY	YES	YES	YES	YES	YES
DBTYPE_DECIMAL	YES	YES	YES	YES	YES
DBTYPE_NUMERIC	YES	YES	YES	YES	YES
DBTYPE_DATE	YES	YES	YES	YES	YES
DBTYPE_BOOL	YES	YES	YES	YES	YES
DBTYPE_BYTES	YES	YES	YES	YES	YES
DBTYPE_BSTR	YES	YES	YES	YES	YES
DBTYPE_STR	YES	YES	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES	YES	YES
DBTYPE_VARIANT	YES	YES	YES	YES	YES
DBTYPE_IDISPATCH	NO	NO	NO	NO	NO
DBTYPE_IUNKNOWN	NO	NO	NO	NO	NO
DBTYPE_GUID	YES	YES	YES	YES	NO
DBTYPE_ERROR	NO	NO	NO	NO	NO
DBTYPE_BYREF	*	*	*	*	*

Table 2-4. OLE DB datatype compatibility with CHAR, NCHAR, VARCHAR, NVARCHAR, and LVARCHAR. (continued)

	CHAR	NCHAR	VARCHAR	NVARCHAR	LVARCHAR
DBTYPE_ARRAY	NO	NO	NO	NO	NO
DBTYPE_VECTOR	NO	NO	NO	NO	NO
DBTYPE_UDT	NO	NO	NO	NO	NO
DBTYPE_DBDATE	YES	YES	YES	YES	YES
DBTYPE_DBTIME	YES	YES	YES	YES	YES
DBTYPE_DBTIMESTAMP	YES	YES	YES	YES	YES
DBTYPE_FILETIME	YES	YES	YES	YES	YES
DBTYPE_PROP_VARIANT	YES	YES	YES	YES	YES
DBTYPE_HCHAPTER	NO	NO	NO	NO	NO
DBTYPE_VARNUMERIC	YES	YES	YES	YES	YES

Table 2-5. OLE DB datatype compatibility with MONEY, DATE, DATETIME, and INTERVAL.

	MONEY	DATE	DATETIME	INTERVAL
DBTYPE_EMPTY	NO	NO	NO	NO
DBTYPE_NULL	NO	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO	NO
DBTYPE_I1	YES	NO	NO	YES
DBTYPE_I2	YES	NO	NO	YES
DBTYPE_I4	YES	NO	NO	YES
DBTYPE_I8	NO	NO	NO	YES
DBTYPE_UI1	YES	NO	NO	YES
DBTYPE_UI2	YES	NO	NO	YES
DBTYPE_UI4	YES	NO	NO	YES
DBTYPE_UI8	NO	NO	NO	YES
DBTYPE_R4	YES	NO	NO	NO
DBTYPE_R8	YES	NO	NO	NO
DBTYPE_CY	YES	NO	NO	NO
DBTYPE_DECIMAL	YES	NO	NO	NO
DBTYPE_NUMERIC	YES	NO	NO	NO
DBTYPE_DATE	YES	YES	YES	NO
DBTYPE_BOOL	YES	NO	NO	NO
DBTYPE_BYTES	YES	YES	YES	NO
DBTYPE_BSTR	YES	YES	YES	YES
DBTYPE_STR	YES	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES	YES
DBTYPE_VARIANT	YES	YES	YES	NO
DBTYPE_IDISPATCH	NO	NO	NO	NO
DBTYPE_IUNKNOWN	NO	NO	NO	NO
DBTYPE_GUID	NO	NO	NO	NO

Table 2-5. OLE DB datatype compatibility with MONEY, DATE, DATETIME, and INTERVAL. (continued)

	MONEY	DATE	DATETIME	INTERVAL
DBTYPE_ERROR	NO	NO	NO	NO
DBTYPE_BYREF	*	NO	NO	NO
DBTYPE_ARRAY	NO	NO	NO	NO
DBTYPE_VECTOR	NO	NO	NO	NO
DBTYPE_UDT	NO	NO	NO	NO
DBTYPE_DBDATE	NO	YES	YES	NO
DBTYPE_DBTIME	NO	YES	YES	NO
DBTYPE_DBTIMESTAMP	NO	YES	YES	NO
DBTYPE_FILETIME	NO	YES	YES	NO
DBTYPE_PROP_VARIANT	YES	NO	NO	NO
DBTYPE_HCHAPTER	NO	NO	NO	NO
DBTYPE_VARNUMERIC	YES	NO	NO	NO

Table 2-6. OLE DB datatype compatibility with CLOB, BLOB, and ROW.

	CLOB	BLOB	ROW
DBTYPE_EMPTY	NO	NO	NO
DBTYPE_NULL	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO
DBTYPE_I1	NO	NO	NO
DBTYPE_I2	NO	NO	NO
DBTYPE_I4	NO	NO	NO
DBTYPE_I8	NO	NO	NO
DBTYPE_UI1	NO	NO	NO
DBTYPE_UI2	NO	NO	NO
DBTYPE_UI4	NO	NO	NO
DBTYPE_UI8	NO	NO	NO
DBTYPE_R4	NO	NO	NO
DBTYPE_R8	NO	NO	NO
DBTYPE_CY	NO	NO	NO
DBTYPE_DECIMAL	NO	NO	NO
DBTYPE_NUMERIC	NO	NO	NO
DBTYPE_DATE	NO	NO	NO
DBTYPE_BOOL	NO	NO	NO
DBTYPE_BYTES	NO	YES	NO
DBTYPE_BSTR	YES	YES	YES
DBTYPE_STR	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES
DBTYPE_VARIANT	NO	NO	YES
DBTYPE_IDISPATCH	NO	NO	NO
DBTYPE_IUNKNOWN	YES	YES	NO

Table 2-6. OLE DB datatype compatibility with CLOB, BLOB, and ROW. (continued)

	CLOB	BLOB	ROW
DBTYPE_GUID	NO	NO	NO
DBTYPE_ERROR	NO	NO	NO
DBTYPE_BYREF	*	*	*
DBTYPE_ARRAY	NO	NO	NO
DBTYPE_VECTOR	NO	NO	NO
DBTYPE_UDT	NO	NO	NO
DBTYPE_DBDATE	NO	NO	NO
DBTYPE_DBTIME	NO	NO	NO
DBTYPE_DBTIMESTAMP	NO	NO	NO
DBTYPE_FILETIME	NO	NO	NO
DBTYPE_PROP_VARIANT	NO	NO	NO
DBTYPE_HCHAPTER	NO	NO	NO
DBTYPE_VARNUMERIC	NO	NO	NO

Table 2-7. OLE DB datatype compatibility with SET, MULTISET, and LIST.

	SET	MULTISET	LIST
DBTYPE_EMPTY	NO	NO	NO
DBTYPE_NULL	NO	NO	NO
DBTYPE_RESERVED	NO	NO	NO
DBTYPE_I1	NO	NO	NO
DBTYPE_I2	NO	NO	NO
DBTYPE_I4	NO	NO	NO
DBTYPE_I8	NO	NO	NO
DBTYPE_UI1	NO	NO	NO
DBTYPE_UI2	NO	NO	NO
DBTYPE_UI4	NO	NO	NO
DBTYPE_UI8	NO	NO	NO
DBTYPE_R4	NO	NO	NO
DBTYPE_R8	NO	NO	NO
DBTYPE_CY	NO	NO	NO
DBTYPE_DECIMAL	NO	NO	NO
DBTYPE_NUMERIC	NO	NO	NO
DBTYPE_DATE	NO	NO	NO
DBTYPE_BOOL	NO	NO	NO
DBTYPE_BYTES	NO	NO	NO
DBTYPE_BSTR	YES	YES	YES
DBTYPE_STR	YES	YES	YES
DBTYPE_WSTR	YES	YES	YES
DBTYPE_VARIANT	YES	YES	YES
DBTYPE_IDISPATCH	NO	NO	NO

Table 2-7. OLE DB datatype compatibility with SET, MULTISET, and LIST. (continued)

	SET	MULTISET	LIST
DBTYPE_IUNKNOWN	NO	NO	NO
DBTYPE_GUID	NO	NO	NO
DBTYPE_ERROR	NO	NO	NO
DBTYPE_BYREF	*	*	NO
DBTYPE_ARRAY	NO	NO	NO
DBTYPE_VECTOR	NO	NO	NO
DBTYPE_UDT	NO	NO	NO
DBTYPE_DBDATE	NO	NO	NO
DBTYPE_DBTIME	NO	NO	NO
DBTYPE_DBTIMESTAMP	NO	NO	NO
DBTYPE_FILETIME	NO	NO	NO
DBTYPE_PROP_VARIANT	NO	NO	NO
DBTYPE_HCHAPTER	NO	NO	NO
DBTYPE_VARNUMERIC	NO	NO	NO

Important:

- All the OLE DB types that are allowed with one or more of the Informix data types for the DBTYPE_BYREF type are also allowed when combined with DBTYPE_BYREF.
- For DBTYPE_IUNKNOWN, the supported interfaces are ISequentialStream, IStream, and ILockBytes.

The 32 K LVARCHAR feature extends LVARCHAR columns to hold up to 32 K bytes of data. This feature requires IBM Informix side support for 32 K LVARCHAR, and only works with IBM Informix Version 9.4 or later.

Data conversions from IBM Informix types to OLE DB types

The following tables show the supported data conversions from IBM Informix types to OLE DB types:

- IBM Informix datatype compatibility with DBTYPE_EMPTY, DBTYPE_NULL, DBTYPE_RESERVED, and DBTYPE_I1.
- IBM Informix datatype compatibility with DBTYPE_I2, DBTYPE_I4, DBTYPE_I8, and DBTYPE_UI1.
- IBM Informix datatype compatibility with DBTYPE_UI2, DBTYPE_UI4, DBTYPE_UI2, DBTYPE_UI8, and DBTYPE_R4.
- IBM Informix datatype compatibility with DBTYPE_R8, DBTYPE_CY, DBTYPE_DECIMAL, and DBTYPE_NUMERIC.
- IBM Informix datatype compatibility with , DBTYPE_DATE, DBTYPE_BOOL, .DBTYPE_BYTES, and DBTYPE_BSTR.
- IBM Informix datatype compatibility with DBTYPE_STR, DBTYPE_WSTR, DBTYPE_VARIANT, and DBTYPE_IDISPATCH.
- IBM Informix datatype compatibility with DBTYPE_IUNKNOWN, DBTYPE_GUID, DBTYPE_ERROR, and DBTYPE_BYREF.

- IBM Informix datatype compatibility with DBTYPE_ARRAY, DBTYPE_VECTOR, DBTYPE_UDT, and DBTYPE_DBDATE.
- IBM Informix datatype compatibility with DBTYPE_DBTIME, DBTYPE_DBTIMESTAMP, and DBTYPE_FILETIME
- IBM Informix datatype compatibility with DBTYPE_PROP_VARIANT, DBTYPE_HCHAPTER, and DBTYPE_VARNUMERIC.

Note: Truncation of data might occur in some cases.

Note: The 32 K LVARCHAR feature extends LVARCHAR columns to hold up to 32 K bytes of data. This feature requires IBM Informix side support for 32 K LVARCHAR, and only works with IBM Informix Version 9.4 or later.

Table 2-8. IBM Informix datatype compatibility with DBTYPE_EMPTY, DBTYPE_NULL, DBTYPE_RESERVED, and DBTYPE_I1.

	DBTYPE_EMPTY	DBTYPE_NULL	DBTYPE_RESERVED	DBTYPE_I1
SMALLINT	NO	NO	NO	YES
INTEGER	NO	NO	NO	YES
INT8	NO	NO	NO	YES
BIGINT	NO	NO	NO	YES
SERIAL	NO	NO	NO	YES
SERIAL8	NO	NO	NO	YES
BIGSERIAL	NO	NO	NO	YES
NUMERIC	NO	NO	NO	YES
DECIMAL	NO	NO	NO	YES
FLOAT	NO	NO	NO	YES
SMALLFLOAT	NO	NO	NO	YES
MONEY	NO	NO	NO	YES
DATE	NO	NO	NO	NO
DATETIME	NO	NO	NO	NO
INTERVAL	NO	NO	NO	YES
CHAR	NO	NO	NO	YES
NCHAR	NO	NO	NO	YES
VARCHAR	NO	NO	NO	YES
NVARCHAR	NO	NO	NO	YES
LVARCHAR	NO	NO	NO	YES
CLOB	NO	NO	NO	NO
BLOB	NO	NO	NO	NO
ROW	NO	NO	NO	NO
SET	NO	NO	NO	NO
MULTISET	NO	NO	NO	NO
LIST	NO	NO	NO	NO

Table 2-9. IBM Informix datatype compatibility with DBTYPE_I2, DBTYPE_I4, DBTYPE_I8, and DBTYPE_UI1.

	DBTYPE_I2	DBTYPE_I4	DBTYPE_I8	DBTYPE_UI1
SMALLINT	YES	YES	NO	YES
INTEGER	YES	YES	NO	YES
INT8	YES	YES	NO	YES
BIGINT	YES	YES	NO	YES
SERIAL	YES	YES	NO	YES
SERIAL8	YES	YES	NO	YES
BIGSERIAL	YES	YES	NO	YES
NUMERIC	YES	YES	NO	YES
DECIMAL	YES	YES	NO	YES
FLOAT	YES	YES	NO	YES
SMALLFLOAT	YES	YES	NO	YES
MONEY	YES	YES	NO	YES
DATE	NO	NO	NO	NO
DATETIME	NO	NO	NO	NO
INTERVAL	YES	YES	YES	YES
CHAR	YES	YES	YES	YES
NCHAR	YES	YES	YES	YES
VARCHAR	YES	YES	YES	YES
NVARCHAR	YES	YES	YES	YES
LVARCHAR	YES	YES	NO	YES
CLOB	NO	NO	NO	NO
BLOB	NO	NO	NO	NO
ROW	NO	NO	NO	NO
SET	NO	NO	NO	NO
MULTISET	NO	NO	NO	NO
LIST	NO	NO	NO	NO

Table 2-10. IBM Informix datatype compatibility with DBTYPE_UI2, DBTYPE_UI4, DBTYPE_UI2, DBTYPE_UI8, and DBTYPE_R4.

	DBTYPE_UI2	DBTYPE_UI4	DBTYPE_UI8	DBTYPE_R4
SMALLINT	YES	YES	NO	YES
INTEGER	YES	YES	NO	YES
INT8	YES	YES	NO	YES
BIGINT	YES	YES	NO	YES
SERIAL	YES	YES	NO	YES
SERIAL8	YES	YES	NO	YES
BIGSERIAL	YES	YES	NO	YES
NUMERIC	YES	YES	NO	YES
DECIMAL	YES	YES	NO	YES
FLOAT	YES	YES	NO	YES

Table 2-10. IBM Informix datatype compatibility with DBTYPE_UI2, DBTYPE_UI4, DBTYPE_UI2, DBTYPE_UI8, and DBTYPE_R4. (continued)

	DBTYPE_UI2	DBTYPE_UI4	DBTYPE_UI8	DBTYPE_R4
SMALLFLOAT	YES	YES	NO	YES
MONEY	YES	YES	NO	YES
DATE	NO	NO	NO	NO
DATETIME	NO	NO	NO	NO
INTERVAL	YES	YES	YES	NO
CHAR	YES	YES	YES	YES
NCHAR	YES	YES	YES	YES
VARCHAR	YES	YES	YES	YES
NVARCHAR	YES	YES	YES	YES
LVARCHAR	YES	YES	NO	YES
CLOB	NO	NO	NO	NO
BLOB	NO	NO	NO	NO
ROW	NO	NO	NO	NO
SET	NO	NO	NO	NO
MULTISET	NO	NO	NO	NO
LIST	NO	NO	NO	NO

Table 2-11. IBM Informix datatype compatibility with DBTYPE_R8, DBTYPE_CY, DBTYPE_DECIMAL, and DBTYPE_NUMERIC.

	DBTYPE_R8	DBTYPE_CY	DBTYPE_DECIMAL	DBTYPE_NUMERIC
SMALLINT	YES	YES	YES	YES
INTEGER	YES	YES	YES	YES
INT8	YES	YES	YES	YES
BIGINT	YES	YES	YES	YES
SERIAL	YES	YES	YES	YES
SERIAL8	YES	YES	YES	YES
BIGSERIAL	YES	YES	YES	YES
NUMERIC	YES	YES	YES	YES
DECIMAL	YES	YES	YES	YES
FLOAT	YES	YES	YES	YES
SMALLFLOAT	YES	YES	YES	YES
MONEY	YES	YES	YES	YES
DATE	NO	NO	NO	NO
DATETIME	NO	NO	NO	NO
INTERVAL	NO	NO	NO	NO
CHAR	YES	YES	YES	YES
NCHAR	YES	YES	YES	YES
VARCHAR	YES	YES	YES	YES
NVARCHAR	YES	YES	YES	YES

Table 2-11. IBM Informix datatype compatibility with DBTYPE_R8, DBTYPE_CY, DBTYPE_DECIMAL, and DBTYPE_NUMERIC. (continued)

	DBTYPE_R8	DBTYPE_CY	DBTYPE_DECIMAL	DBTYPE_NUMERIC
LVARCHAR	YES	YES	YES	YES
CLOB	NO	NO	NO	NO
BLOB	NO	NO	NO	NO
ROW	NO	NO	NO	NO
SET	NO	NO	NO	NO
MULTISET	NO	NO	NO	NO
LIST	NO	NO	NO	NO

Table 2-12. IBM Informix datatype compatibility with , DBTYPE_DATE, DBTYPE_BOOL, DBTYPE_BYTES, and DBTYPE_BSTR.

	DBTYPE_DATE	DBTYPE_BOOL	DBTYPE_BYTES	DBTYPE_BSTR
SMALLINT	YES	YES	YES	YES
INTEGER	YES	YES	YES	YES
INT8	YES	YES	YES	YES
BIGINT	YES	YES	YES	YES
SERIAL	YES	YES	YES	YES
SERIAL8	YES	YES	YES	YES
BIGSERIAL	YES	YES	YES	YES
NUMERIC	YES	YES	YES	YES
DECIMAL	YES	YES	YES	YES
FLOAT	YES	YES	YES	YES
SMALLFLOAT	YES	YES	YES	YES
MONEY	YES	YES	YES	YES
DATE	YES	NO	YES	YES
DATETIME	YES	NO	YES	YES
INTERVAL	NO	NO	NO	YES
CHAR	YES	YES	YES	YES
NCHAR	YES	YES	YES	YES
VARCHAR	YES	YES	YES	YES
NVARCHAR	YES	YES	YES	YES
LVARCHAR	YES	YES	YES	YES
CLOB	NO	NO	NO	YES
BLOB	NO	NO	YES	YES
ROW	NO	NO	NO	YES
SET	NO	NO	NO	YES
MULTISET	NO	NO	NO	YES
LIST	NO	NO	NO	YES

Table 2-13. IBM Informix datatype compatibility with DBTYPE_STR, DBTYPE_WSTR, DBTYPE_VARIANT, and DBTYPE_IDISPATCH.

	DBTYPE_STR	DBTYPE_WSTR	DBTYPE_VARIANT	DBTYPE_IDISPATCH
SMALLINT	YES	YES	YES	NO
INTEGER	YES	YES	YES	NO
INT8	YES	YES	YES	NO
BIGINT	YES	YES	YES	NO
SERIAL	YES	YES	YES	NO
SERIAL8	YES	YES	YES	NO
BIGSERIAL	YES	YES	YES	NO
NUMERIC	YES	YES	YES	NO
DECIMAL	YES	YES	YES	NO
FLOAT	YES	YES	YES	NO
SMALLFLOAT	YES	YES	YES	NO
MONEY	YES	YES	YES	NO
DATE	YES	YES	YES	NO
DATETIME	YES	YES	YES	NO
INTERVAL	YES	YES	NO	NO
CHAR	YES	YES	YES	NO
NCHAR	YES	YES	YES	NO
VARCHAR	YES	YES	YES	NO
NVARCHAR	YES	YES	YES	NO
LVARCHAR	YES	YES	YES	NO
CLOB	YES	YES	NO	NO
BLOB	YES	YES	NO	NO
ROW	YES	YES	YES	NO
SET	YES	YES	YES	NO
MULTISET	YES	YES	YES	NO
LIST	YES	YES	YES	NO

Table 2-14. IBM Informix datatype compatibility with DBTYPE_IUNKNOWN, DBTYPE_GUID, DBTYPE_ERROR, and DBTYPE_BYREF.

	DBTYPE_IUNKNOWN	DBTYPE_GUID	DBTYPE_ERROR	DBTYPE_BYREF
SMALLINT	NO	NO	NO	*
INTEGER	NO	NO	NO	*
INT8	NO	NO	NO	*
BIGINT	NO	NO	NO	*
SERIAL	NO	NO	NO	*
SERIAL8	NO	NO	NO	*
BIGSERIAL	NO	NO	NO	*
NUMERIC	NO	NO	NO	*
DECIMAL	NO	NO	NO	*

Table 2-14. IBM Informix datatype compatibility with DBTYPE_IUNKNOWN, DBTYPE_GUID, DBTYPE_ERROR, and DBTYPE_BYREF. (continued)

	DBTYPE_IUNKNOWN	DBTYPE_GUID	DBTYPE_ERROR	DBTYPE_BYREF
FLOAT	NO	NO	NO	*
SMALLFLOAT	NO	NO	NO	*
MONEY	NO	NO	NO	*
DATE	NO	NO	NO	NO
DATETIME	NO	NO	NO	NO
INTERVAL	NO	NO	NO	NO
CHAR	NO	YES	NO	*
NCHAR	NO	YES	NO	*
VARCHAR	NO	YES	NO	*
NVARCHAR	NO	YES	NO	*
LVARCHAR	NO	NO	NO	*
CLOB	YES	NO	NO	*
BLOB	YES	NO	NO	*
ROW	NO	NO	NO	*
SET	NO	NO	NO	*
MULTISET	NO	NO	NO	*
LIST	NO	NO	NO	NO

Important:

- *All the OLE DB types that are allowed with one or more of the Informix data types for the DBTYPE_BYREF type are also allowed when combined with DBTYPE_BYREF.
- For DBTYPE_IUNKNOWN, the supported interfaces are ISequentialStream, IStream, and ILockBytes.

Table 2-15. IBM Informix datatype compatibility with DBTYPE_ARRAY, DBTYPE_VECTOR, DBTYPE_UDT, and DBTYPE_DBDATE.

	DBTYPE_ARRAY	DBTYPE_VECTOR	DBTYPE_UDT	DBTYPE_DBDATE
SMALLINT	NO	NO	NO	NO
INTEGER	NO	NO	NO	NO
INT8	NO	NO	NO	NO
BIGINT	NO	NO	NO	NO
SERIAL	NO	NO	NO	NO
SERIAL8	NO	NO	NO	NO
BIGSERIAL	NO	NO	NO	NO
NUMERIC	NO	NO	NO	NO
DECIMAL	NO	NO	NO	NO
FLOAT	NO	NO	NO	NO
SMALLFLOAT	NO	NO	NO	NO
MONEY	NO	NO	NO	NO
DATE	NO	NO	NO	YES

Table 2-15. IBM Informix datatype compatibility with DBTYPE_ARRAY, DBTYPE_VECTOR, DBTYPE_UDT, and DBTYPE_DBDATE. (continued)

	DBTYPE_ARRAY	DBTYPE_VECTOR	DBTYPE_UDT	DBTYPE_DBDATE
DATETIME	NO	NO	NO	YES
INTERVAL	NO	NO	NO	NO
CHAR	NO	NO	NO	YES
NCHAR	NO	NO	NO	YES
VARCHAR	NO	NO	NO	YES
NVARCHAR	NO	NO	NO	YES
LVARCHAR	NO	NO	NO	YES
CLOB	NO	NO	NO	NO
BLOB	NO	NO	NO	NO
ROW	NO	NO	NO	NO
SET	NO	NO	NO	NO
MULTISET	NO	NO	NO	NO
LIST	NO	NO	NO	NO

Table 2-16. IBM Informix datatype compatibility with DBTYPE_DBTIME, DBTYPE_DBTIMESTAMP, and DBTYPE_FILETIME.

	DBTYPE_DBTIME	DBTYPE_DBTIMESTAMP	DBTYPE_FILETIME
SMALLINT	NO	NO	NO
INTEGER	NO	NO	NO
INT8	NO	NO	NO
BIGINT	NO	NO	NO
SERIAL	NO	NO	NO
SERIAL8	NO	NO	NO
BIGSERIAL	NO	NO	NO
NUMERIC	NO	NO	NO
DECIMAL	NO	NO	NO
FLOAT	NO	NO	NO
SMALLFLOAT	NO	NO	NO
MONEY	NO	NO	NO
DATE	YES	YES	YES
DATETIME	YES	YES	YES
INTERVAL	NO	NO	NO
CHAR	YES	YES	YES
NCHAR	YES	YES	YES
VARCHAR	YES	YES	YES
NVARCHAR	YES	YES	YES
LVARCHAR	YES	YES	YES
CLOB	NO	NO	NO
BLOB	NO	NO	NO

Table 2-16. IBM Informix datatype compatibility with DBTYPE_DBTIME, DBTYPE_DBTIMESTAMP, and DBTYPE_FILETIME. (continued)

	DBTYPE_DBTIME	DBTYPE_DBTIMESTAMP	DBTYPE_FILETIME
ROW	NO	NO	NO
SET	NO	NO	NO
MULTISET	NO	NO	NO
LIST	NO	NO	NO

Table 2-17. IBM Informix datatype compatibility with DBTYPE_PROP_VARIANT, DBTYPE_HCHAPTER, and DBTYPE_VARNUMERIC.

	DBTYPE_PROP_VARIANT	DBTYPE_HCHAPTER	DBTYPE_VARNUMERIC
SMALLINT	YES	NO	YES
INTEGER	YES	NO	YES
INT8	YES	NO	YES
BIGINT	YES	NO	YES
SERIAL	YES	NO	YES
SERIAL8	YES	NO	YES
BIGSERIAL	YES	NO	YES
NUMERIC	YES	NO	YES
DECIMAL	YES	NO	YES
FLOAT	YES	NO	YES
SMALLFLOAT	YES	NO	YES
MONEY	YES	NO	YES
DATE	NO	NO	NO
DATETIME	NO	NO	NO
INTERVAL	NO	NO	NO
CHAR	YES	NO	YES
NCHAR	YES	NO	YES
VARCHAR	YES	NO	YES
NVARCHAR	YES	NO	YES
LVARCHAR	YES	NO	YES
CLOB	NO	NO	NO
BLOB	NO	NO	NO
ROW	NO	NO	NO
SET	NO	NO	NO
MULTISET	NO	NO	NO
LIST	NO	NO	NO

Threading support

IBM Informix OLE DB Provider supports both free-threading and apartment-threading models. (For more information about these threading models search for "understanding threading models" at the Microsoft website.) The free-threading model improves scalability and allows connection caching.

Transaction support

IBM Informix OLE DB Provider supports both local and distributed transactions. A cross-server distributed transaction spans two or more database servers (also known as resource managers); these database servers can be heterogeneous server instances anywhere on the network.

Distributed transactions

A *transaction coordinator* ensures that the distributed system maintains a consistent state. The transaction coordinator that IBM Informix OLE DB Provider supports for distributed transactions is MS DTC (Microsoft Distributed Transaction Coordinator).

Local transactions

For local transactions, the default value of the Autocommit isolation level is read-committed for databases created with logging and read-uncommitted for databases created without logging.

The consumer can change the default to read-uncommitted or serializable by setting the DBPROP_SESS_AUTOCOMMITISOLEVEL property.

Important: When the isolation level is set to read-uncommitted for a database that has logging, data can be read from but not written to the database.

Distributed transaction support

IBM Informix OLE DB Provider supports distributed transactions coordinated by MS DTC.

Beginning with Windows 2003, if the XA DLL registry entry is not created during installation, you must specifically create it. Informix OLE DB Provider version 3.0 uses `IfmxConn.dll` as the XA library. Enter the name of the DLL only; do not enter the entire DLL path name in MS DTC with user-specified DLLs.

When you upgrade to Windows XP Professional SP2 or Windows Server 2003, XA transactions are disabled, which protects MS DTC from denial-of-service attacks. To enable XA transactions, see the Microsoft XA Transaction Support information located at http://support.microsoft.com.

MDAC 2.8 SP1 requires IBM Informix OLE DB Provider version 3.0 or later to function properly with MS DTC.

Identifiers

IBM Informix OLE DB Provider sets the **DELIMIDENT** environment variable in Setnet32 to Y, so that it encloses all identifiers in quotation marks in the SQL it generates (for example, when it executes an update). You can override this behavior by setting the **DELIMIDENT** environment variable to N.

For more information about the **DELIMIDENT** environment variable, see the *IBM Informix Guide to SQL: Reference*.

Tip: Identifiers are case sensitive when enclosed in quotation marks.

Support for Dynamic Query Extension

The Dynamic Query Extension feature introduces support for describing input parameters of a prepared statement. This feature is an enhancement of the Dynamic SQL functionality of the server.

This feature requires IBM Informix server-side support for Dynamic Query Extension, and works only with IBM Informix Version 9.4 or later.

To obtain the metadata for the parameters in a query in an OLE DB client, use the GetParameterInfo method of the ICommandWithParameters interface in the Command class.

Support for SQL 99 Joins

The SQL 99 Joins feature extends support for SQL joins from both within and outside of an escape sequence. This feature requires IBM Informix server-side support for SQL 99 joins, and only works with IBM Informix Version 9.4 or later.

International GLS Locales

This section offers information and tips about using IBM Informix OLE DB Provider with Global Language Support (GLS) locales other than the default locale, **en_us.1252-1**.

For complete information about GLS locales, see the *IBM Informix GLS User's Guide*.

Conversion between Unicode and MBCS character sets

IBM Informix OLE DB Provider uses Win32 functions to convert between Unicode and Multibyte Character Sequence (MBCS). Informix OLE DB Provider operates under the assumption that the client locale corresponds to one of the installed Windows code pages.

The UNICODE provider string keyword

If the UNICODE provider string keyword is set to FALSE (default), the code page corresponding to the CLIENT_LOCALE must be present as one of the operating system code pages. In this situation, the OLE DB Provider uses IBM Informix GLS functions to convert from DB_LOCALE to CLIENT_LOCALE, then uses operating system functions to convert from CLIENT_LOCALE to Unicode. This mechanism does not load theInformix GLS code page, which results in better connection performance but slower code set conversions.

If the UNICODE provider string keyword is set to TRUE, the code page corresponding to the CLIENT_LOCALE does not need to be present as one of the operating system code pages. This would be required, if you wanted to use, for example, a Hebrew code page on a US English Windows machine. In this situation, the OLE DB Provider uses Informix GLS functions to convert directly from DB_LOCALE to Unicode. This mechanism loads the Informix GLS code page, which might slow connection performance slightly but results in faster code set conversions.

Related reference

"Connect to a data source" on page 2-1

The REPORTSTRINGASWSTRING provider string keyword

The provider-string keyword RSASWS or REPORTSTRINGASWSTRING in the provider string Extended Properties enables you to control the data mapping for wide strings.

When this keyword is set to TRUE, OLE DB Provider reports DBTYPE_WSTR as a best fit for all the underlying string length data types (CHAR, VARCHAR, TEXT, and other data types) and not DBTYPE_STR, which is the normal mapping. The default setting for REPORTSTRINGASWSTRING is FALSE.

The syntax for setting this keyword is as follows (two forms of this keyword are provided; you might use either one):

- Short form:

  `RSASWS=TRUE` or `RSASWS=FALSE`
- Long form:

  `REPORTSTRINGASWSTRING=TRUE` or `REPORTSTRINGASWSTRING=FALSE`

Related reference

"Connect to a data source" on page 2-1

Troubleshooting installation, configuration, or operation problems

This section describes how to resolve problems that you might encounter when installing, configuring, or using IBM Informix OLE DB Provider.

Tip: If the problem you are experiencing does not match one listed here, or the proposed resolution does not work for you, contact Technical Support.

IBM Informix OLE DB Provider not registered

When you attempt to connect to an IBM Informix data source, a message says that IBM Informix OLE DB Provider is not registered.

Informix OLE DB Provider is not visible in the enumeration (for example, in the Initialize Data Source dialog box in the Microsoft OLE DB query demonstration).

Possible cause

Informix OLE DB Provider is not installed.

Resolution

Informix OLE DB Provider is distributed with IBM Informix Connect and the IBM Informix Client Software Development Kit, Version 2.3 and later; however, it is not installed unless you choose the **Custom** installation option and explicitly select IBM Informix OLE DB Provider.

During installation, the Informix OLE DB Provider DLL is copied to `INFORMIXDIR\bin`. If Informix OLE DB Provider is copied to your computer but still is not visible in the enumeration, make sure the DLL is registered on the local computer.

To register the DLL

1. Go to `INFORMIXDIR\bin`.
2. Run the **regsvr32** command on the DLL (`ifxoledbc.dll`).

Related reference

"Class not registered"

Class not registered

When you attempt to connect to an IBM Informix data source, the message `Class not registered` opens.

Possible cause

The Informix OLE DB Provider DLL might not be loaded.

Resolution

Check that the Informix OLE DB Provider DLL is in the location recorded in the registry entry, which points to `bin\ifxoledbc.dll` in your IBM Informix Connect or IBM Informix Client Software Development Kit installation. If that is not the case, reregister Informix OLE DB Provider. See the registry entries topics in the Microsoft OLE DB documentation for more information.

Also, make sure that `INFORMIXDIR\bin` is in the system path.

Related reference

"IBM Informix OLE DB Provider not registered" on page 2-26

"Attempt to use provider from web server or other server fails" on page 2-29

Cannot establish a connection

You cannot establish a connection.

Possible cause

Basic connectivity was not set up.

Resolution

Use **Ilogin** or **DBPing** (included with your IBM Informix Client Software Development Kit) to verify that you can connect.

Related reference

"Attempt to use provider from web server or other server fails" on page 2-29

Database not found

A connection attempt fails; a message says that the database is not found.

Possible cause

A bad database name or no database name at all was specified, and no database corresponding to your client user name exists on the server.

Resolution

Make sure that your data source name is specified correctly.

Related reference

"Connect to a data source" on page 2-1

"Attempt to use provider from web server or other server fails" on page 2-29

The oledbversion table not found

When the application attempts to fetch schema information, a message says that the table **oledbversion** was not found.

Possible cause

The setup script, `coledbp.sql`, has not been run against the **sysmaster** database of that server.

Resolution

The database administrator must run the setup script against the **sysmaster** database on the server to which you are trying to connect.

Nonalphabetic MBCS characters generate syntax errors

When you issue an SQL statement against an MBCS database (for example, SJIS-S), the IBM Informix server returns a syntax error if the statement includes table or column names containing MBCS characters not classified as alphabetic in the locale.

Possible cause

Identifiers that include nonalphabetic characters are not enclosed in quotation marks.

Resolution

Enclose identifiers in quotation marks, and make sure **`DELIMIDENT`** is set. If you have no control over the SQL produced by the application, consider using a locale that classifies the characters in question as alphabetic.

Server-side cursor fails to update records

If you open a server-side cursor (**adUseServer**) on an SQL command (**adCmdText**), attempts to perform an update fail with an `ADO provider not capable` error.

This problem applies only to 7.x, 8.x, and 9.1x servers.

Possible cause

A server-side cursor that is opened on SQL text against IBM Informix servers before version 9.2 cannot be updated.

Resolution

Use the client-side cursor (**adUseClient**) instead, or open the server-side cursor on the table (**adCmdTableDirect**).

Attempt to use provider from web server or other server fails

An attempt to use OLE DB Provider from the web server or from a process that runs as a distinct user fails. The typical error message that is displayed is `ADO cannot find the provider`.

Possible cause

OLE DB Provider is set up only for the current user.

Resolution

Perform the following steps in order:

1. Verify that you can connect to an IBM Informix data source from an application such as Microsoft Visual Basic 6 or Microsoft Query Demo.
2. Make sure that `INFORMIXDIR\bin` is in the system path (as opposed to the user path).
3. Run `INFORMIXDIR\bin\regcopy.exe` and reboot.

Also, make sure that the user has the necessary permissions to access the database, and verify that the data source has been specified correctly.

Related reference

"Connect to a data source" on page 2-1

"Class not registered" on page 2-27

"Cannot establish a connection" on page 2-27

"Database not found" on page 2-28

Cannot connect to transaction manager

A connection attempt fails, a message says that the application cannot connect to the transaction manager.

Possible cause

This message is generated when the MTS installation does not have the latest update of the component DLLs. The interface, **IDtcToXaHelperSinglePipe** that the IBM Informix OLE DB Provider uses to communicate with MS DTC might not be available in the installation (part of **MSDTCPRX.DLL**).

Resolution

This problem occurs due to incorrect setup. You must ensure that the **msdtcprx** DLL is version 1999 or higher. The MTS component files get updated by installing

Windows NT Service Pack 6. Install the Windows NT Service Pack 6 or higher after the MTS installation.

Driver not found error

When you attempt to run tracing, the server returns an error stating that the driver cannot be not found and might not be installed properly.

Possible cause

A valid path has not been set in the **IFXOLEDBTRACE** environment variable to which the system can write a trace file.

Resolution

Set a valid path such as `c:\oledb\trace.txt` in the **IFXOLEDBTRACE** environment variable.

Appendix. Accessibility

IBM strives to provide products with usable access for everyone, regardless of age or ability.

Accessibility features for IBM Informix products

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in IBM Informix products. These features support:

- Keyboard-only operation.
- Interfaces that are commonly used by screen readers.
- The attachment of alternative input and output devices.

Tip: The information center and its related publications are accessibility-enabled for the IBM Home Page Reader. You can operate all features by using the keyboard instead of the mouse.

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

Related accessibility information

IBM is committed to making our documentation accessible to persons with disabilities. Our publications are available in HTML format so that they can be accessed with assistive technology such as screen reader software.

You can view the publications in Adobe Portable Document Format (PDF) by using the Adobe Acrobat Reader.

IBM and accessibility

See the *IBM Accessibility Center* at http://www.ibm.com/able for more information about the IBM commitment to accessibility.

Dotted decimal syntax diagrams

The syntax diagrams in our publications are available in dotted decimal format, which is an accessible format that is available only if you are using a screen reader.

In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), the elements can appear on the same line, because they can be considered as a single compound syntax element.

Each line starts with a dotted decimal number; for example, `3` or `3.1` or `3.1.1`. To hear these numbers correctly, make sure that your screen reader is set to read punctuation. All syntax elements that have the same dotted decimal number (for example, all syntax elements that have the number `3.1`) are mutually exclusive alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, the word or symbol is preceded by the backslash (\) character. The * symbol can be used next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is read as 3 * FILE. Format 3* FILE indicates that syntax element FILE repeats. Format 3* * FILE indicates that syntax element * FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol that provides information about the syntax elements. For example, the lines 5.1*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, that element is defined elsewhere. The string following the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 means that you should refer to a separate syntax fragment OP1.

The following words and symbols are used next to the dotted decimal numbers:

? Specifies an optional syntax element. A dotted decimal number followed by the ? symbol indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element (for example, 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that syntax elements NOTIFY and UPDATE are optional; that is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

! Specifies a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicates that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the same dotted decimal number can specify a ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the default option for the FILE keyword. In this example, if you include the FILE keyword but do not specify an option, default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP only applies to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.

* Specifies a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be repeated. For example, if you hear the line 5.1* data-area, you know that you can include more than one data area or you can include none. If you hear the lines 3*, 3 HOST, and 3 STATE, you know that you can include HOST, STATE, both together, or nothing.

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.
3. The * symbol is equivalent to a loop-back line in a railroad syntax diagram.

+ Specifies a syntax element that must be included one or more times. A dotted decimal number followed by the + symbol indicates that this syntax element must be included one or more times. For example, if you hear the line 6.1+ data-area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. As for the * symbol, you can only repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the * symbol, is equivalent to a loop-back line in a railroad syntax diagram.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Index

Printed in USA

SC27-3549-01