

IBM[®] DB2[®] Life Sciences Data Connect



Planning, Installation, and Configuration Guide

Version 7

IBM[®] DB2[®] Life Sciences Data Connect



Planning, Installation, and Configuration Guide

Version 7

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 21.

This document contains proprietary information of IBM. It is provided under a license agreement and is protected by copyright law. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

Order publications through your IBM representative or the IBM branch office serving your locality or by calling 1-800-879-2755 in the United States or 1-800-IBM-4YOU in Canada.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 2001. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this book.	v	Adding table-structured files to a federated system	8
Who should read this book	v	Step 1: Registering the wrapper	9
Conventions	v	Step 2: (Optional) Setting the DB2_DJ_COMM environment variable	9
How to read the syntax diagrams	v	Step 3: Registering the server.	9
How to send your comments	viii	Step 4: Registering nicknames	10
Chapter 1. What is DB2 Life Sciences Data Connect?	1	Wrapper limitations and considerations	13
DB2 Life Sciences Data Connect	1	File limitations and considerations	14
IBM Life Sciences DiscoveryLink	2	File access control model.	14
Querying life sciences data	3	Optimization tips and considerations.	14
Chapter 2. Installing DB2 Life Sciences Data Connect	5	Messages	15
Chapter 3. Using table-structured files as data sources.	7	Notices	21
What are table-structured files?	7	Trademarks	24
Types of table-structured files	7	Index	27
How DB2 Life Sciences Data Connect works with table-structured files	8	Contacting IBM	29
		Product Information	29

About this book

This book contains:

- An introduction to DB2 Life Sciences Data Connect and how it fits into the IBM Life Sciences DiscoveryLink offering, a comprehensive set of software and services tailored to the life sciences
- Installation instructions for DB2 Life Sciences Data Connect on AIX
- Instructions for adding data sources to a federated system by registering wrappers — modules that enable you or an application to communicate with a data source using SQL.

Who should read this book

This book is for administrators setting up a federated database environment for life sciences research and development data and for application programmers developing applications for such an environment.

Conventions

This book uses these highlighting conventions:

Boldface type

Indicates commands and graphical user interface (GUI) controls (for example, names of fields, names of folders, menu choices).

Monospace type

Indicates examples of coding or of text that you type.

Italic type

Indicates variables that you should replace with a value. Italic type also indicates book titles and emphasizes words.

UPPERCASE TYPE

Indicates SQL keywords and names of objects (for example, tables, views, and servers).

How to read the syntax diagrams

Throughout this book, syntax is described using the structure defined as follows:

Read the syntax diagrams from left to right and top to bottom, following the path of the line.

The \blacktriangleright — symbol indicates the beginning of a statement.

The \longrightarrow symbol indicates that the statement syntax is continued on the next line.

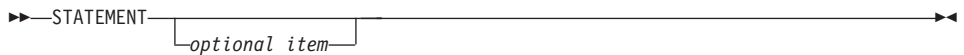
The \blacktriangleright — symbol indicates that a statement is continued from the previous line.

The $\longrightarrow\blacktriangleleft$ symbol indicates the end of a statement.

Required items appear on the horizontal line (the main path).



Optional items appear below the main path.



If an optional item appears above the main path, that item has no effect on the execution of the statement and is used only for readability.



If you can choose from two or more items, they appear in a stack.

If you *must* choose one of the items, one item of the stack appears on the main path.



If choosing none of the items is an option, the entire stack appears below the main path.



If one of the items is the default, it will appear above the main path and the remaining choices will be shown below.



An arrow returning to the left, above the main line, indicates an item that can be repeated. In this case, repeated items must be separated by one or more blanks.



If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can make more than one choice from the stacked items or repeat a single choice.

Keywords appear in uppercase (for example, FROM). They must be spelled exactly as shown. Variables appear in lowercase (for example, column-name). They represent user-supplied names or values in the syntax.

If punctuation marks, parentheses, arithmetic operators, or other such symbols are shown, you must enter them as part of the syntax.

Sometimes a single variable represents a set of several parameters. For example, in the following diagram, the variable `parameter-block` can be replaced by any of the interpretations of the diagram that is headed **parameter-block**:



parameter-block:



Adjacent segments occurring between “large bullets” (●) may be specified in any sequence.



The above diagram shows that item2 and item3 may be specified in either order. Both of the following are valid:

```
STATEMENT item1 item2 item3 item4  
STATEMENT item1 item3 item2 item4
```

How to send your comments

Your feedback helps IBM to provide quality information. Please send any comments that you have about this book or other DB2 documentation. You can use any of the following methods to provide comments:

- Send your comments from the Web. You can access the IBM Data Management online readers’ comment form at <http://www.ibm.com/software/data/rcf>
- Send your comments by e-mail to comments@vnet.ibm.com. Be sure to include the name of the product, the version number of the product, and the name and part number of the book (if applicable). If you are commenting on specific text, please include the location of the text (for example, a chapter and section title, a table number, a page number, or a help topic title).

Chapter 1. What is DB2 Life Sciences Data Connect?

This section introduces you to the DB2 Life Sciences Data Connect product, the IBM Life Sciences DiscoveryLink offering, and the general steps involved in setting up a system to query life sciences data.

DB2 Life Sciences Data Connect

IBM DB2 Life Sciences Data Connect enables a DB2 federated system to integrate genetic, chemical, biological, and other research data from distributed sources. A DB2 federated system is a distributed computing system that consists of a DB2 Universal Database (UDB) server and multiple data sources from which the DB2 UDB server retrieves data.

With a federated system, you or an application can use SQL statements to query, retrieve, and join data located in several heterogeneous data sources, such as relational databases from IBM, Oracle, Sybase, and Microsoft, and non-relational data sources, such as table-structured files. Figure 1 illustrates a federated system using DB2 Life Sciences Data Connect to access multiple sources of research data.

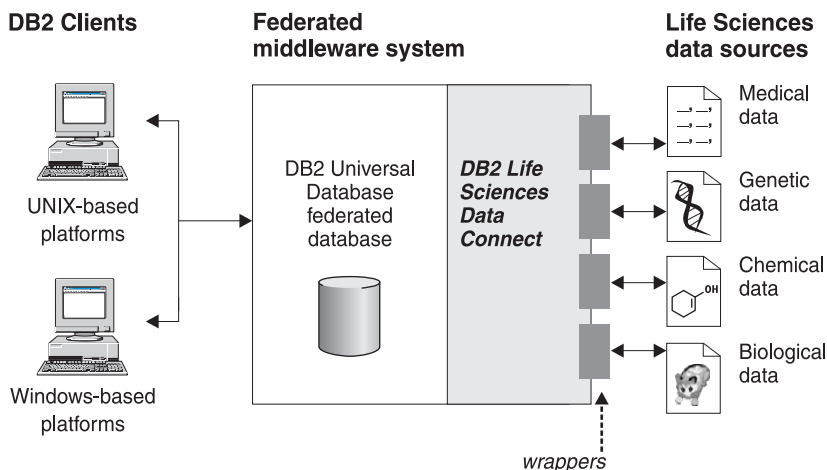


Figure 1. Accessing life sciences data with DB2 Life Sciences Data Connect

A DB2 federated system includes clients, a database to which the clients submit queries (called a federated database), an interface through which the federated database communicates with data sources, and the data sources themselves.

The mechanism by which a federated server communicates with a data source is called a *wrapper*. To implement a wrapper, the server uses routines stored in a library called a *wrapper module*. These routines allow the server to perform operations such as connecting to a data source and retrieving data from it iteratively.

After a federated system is set up, the information in data sources can be accessed as though it is in one large database. Users and applications send queries to one federated database, which retrieves data from multiple data sources. Applications work with the federated database just like with any other DB2 database.

For more information on federated systems, see the *DB2 SQL Reference*.

IBM Life Sciences DiscoveryLink

The DiscoveryLink offering is a set of middleware software and services tailored specifically to life sciences research and development requirements for integrating data from multiple heterogeneous data sources.

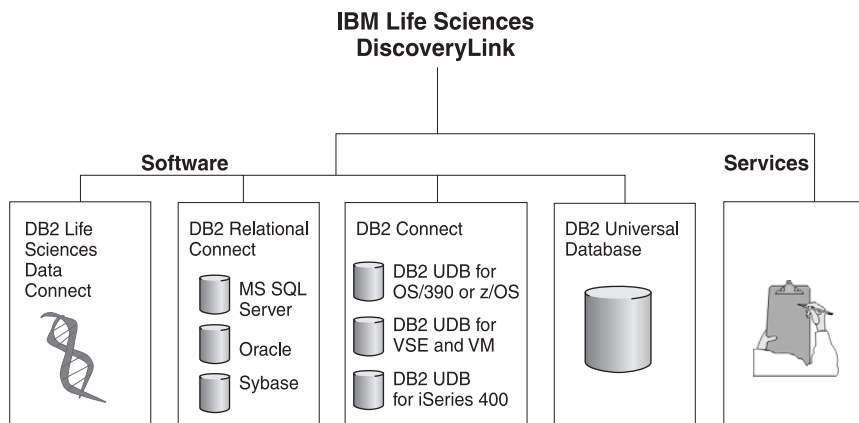


Figure 2. IBM Life Sciences DiscoveryLink

For example, with DiscoveryLink, you can use a single SQL statement to integrate protein sequence data from an Oracle database in Switzerland, chemical structure data from a Sybase database in Japan, and spectroscopic

data stored in table-structured flat files on your local area network. The data appears as if it is in one virtual database.

Software components include:

DB2 Life Sciences Data Connect

For accessing life sciences data.

DB2 Relational Connect

For accessing Oracle, Sybase, and Microsoft relational databases. For more information on DB2 Relational Connect, see the *DB2 Universal Database Release Notes Version 7.2/Version 7.1 FixPak 3*.

DB2 Connect

For accessing DB2 database servers on host systems. For more information on DB2 Connect, see the *DB2 Connect User's Guide*.

DB2 Universal Database

To optimize queries and integrate results across multiple heterogeneous data sources. For more information on DB2 Universal Database, see the *DB2 Administration Guide*.

For more information on DiscoveryLink software and services, send e-mail to ls@us.ibm.com.

Querying life sciences data

To query and retrieve data located in life sciences data sources, you must first install DB2 Life Sciences Data Connect.

After you install DB2 Life Sciences Data Connect, configure the wrapper to the data source. This process is known as registering the wrapper.

The data source supported by DB2 Life Sciences Data Connect is table-structured files on AIX.

Chapter 2. Installing DB2 Life Sciences Data Connect

This chapter describes how to install DB2 Life Sciences Data Connect to query and retrieve life sciences data on AIX.

After it is installed, you must register a wrapper for the data source. Instructions for registering life sciences wrappers are described in the “Chapter 3. Using table-structured files as data sources” on page 7.

Before you install DB2 Life Sciences Data Connect on your AIX federated server:

- Confirm that you have one of the following products installed on the federated server:
 - DB2 Universal Database Enterprise Edition
 - DB2 Universal Database Enterprise - Extended Edition
- Make sure the database has Federated Database System Support turned on. To check this setting, run the following command from the DB2 Command Line Processor:

```
GET DATABASE MANAGER CONFIGURATION
```

This command displays all of the database parameters and their current settings. Confirm that the FEDERATED parameter is set to YES.

If the FEDERATED parameter is set to NO, run the following command from the DB2 command line processor:

```
UPDATE DATABASE MANAGER CONFIGURATION USING FEDERATED YES
```

To install DB2 Life Sciences Data Connect on your AIX server:

1. Log in as a user with root authority.
2. Insert and mount your DB2 Life Sciences Data Connect CD-ROM. For information on how to mount a CD-ROM, see the *DB2 for UNIX Quick Beginnings* manual.
3. Change to the directory where the CD-ROM is mounted by entering the `cd /cdrom` command, where *cdrom* is the mount point for your product CD-ROM.
4. Type the following command:

```
./db2setup
```

The DB2 Setup Utility window opens.
5. Press the space bar to select Distributed Access for Life Sciences data sources. An asterisk appears next to the option when it is selected.
6. Select **OK** and press the Enter key. The DB2 Services window opens.

7. In the DB2 Services window, you can choose to set up an existing DB2 instance for DB2 Life Sciences Data Connect, or you can create a DB2 instance. Use the `db2setup` installation help to guide you through instance creation and setup, and through the remaining installation steps.

When the installation is complete, DB2 Life Sciences Data Connect is installed in the `/usr/lpp/db2_07_01` directory.

Chapter 3. Using table-structured files as data sources

This chapter describes:

- Table-structured files
- How to register the wrapper for table-structured files
- Wrapper and file limitations and considerations
- The file access control model used
- Optimization tips
- Messages you might encounter when working with table-structured files

What are table-structured files?

A table-structured file has a regular structure consisting of a series of records, where each record contains the same number of fields, separated by an arbitrary delimiter. Null values are represented by two delimiters next to each other.

The following example shows the contents of a file called DRUGDATA1.TXT. It contains three records, each with three fields, separated by commas:

```
234,DrugnameA,Manufacturer1
332,DrugnameB,Manufacturer2
333,DrugnameC,Manufacturer2
```

The first field is the drug's unique ID number. The second field is the name of the drug. The third field is the name of the manufacturer who produces the drug.

Types of table-structured files

Table-structured files can be sorted or unsorted.

Sorted files

DRUGDATA1.TXT contains sorted records. The file is sorted by the first field, the drug's unique ID number. This field is the primary key because it is unique for each drug. Sorted files must be sorted in ascending order.

```
234,DrugnameA,Manufacturer1
332,DrugnameB,Manufacturer2
333,DrugnameC,Manufacturer2
```

Unsorted files

DRUGDATA2.TXT contains unsorted records. There is no order to the way the records are listed in the file.

332,DrugnameB,Manufacturer2
234,DrugnameA,Manufacturer1
333,DrugnameC,Manufacturer2

The wrapper can search sorted data files much more efficiently than non-sorted files.

How DB2 Life Sciences Data Connect works with table-structured files

Using a module called a wrapper, DB2 Life Sciences Data Connect can process SQL statements that query data in a table-structured file as if it were contained in an ordinary relational table or view. This enables data in a table-structured file to be joined with relational data or data in other table-structured files.

For example, suppose that the table-structured file DRUGDATA1.TXT is located on your computer in your laboratory. Trying to query this data and match it up with other tables from other data sources that you use can be tedious.

After you register DRUGDATA1.TXT with DB2 Life Sciences Data Connect the file behaves as if it is a relational data source. You can now query the file together with other relational and non-relational data sources and analyze the data together.

For example, you could run the following query:

```
SELECT * FROM DRUGDATA1 ORDER BY DCODE
```

This query produces the following results.

Dcode	Drug	Manufacturer
234	DrugnameA	Manufacturer1
332	DrugnameB	Manufacturer2
333	DrugnameC	Manufacturer2

Adding table-structured files to a federated system

To add a data source for a table-structured file to a federated server, you need to:

1. Register the wrapper using the CREATE WRAPPER command.
2. Optional: Set the DB2_DJ_COMM environment variable to improve query performance.
3. Register the server using the CREATE SERVER command.
4. Register nicknames using the CREATE NICKNAME command for all table-structured files.

These steps are explained in detail in this section. The commands can be run from the DB2 Command Line Processor.

Step 1: Registering the wrapper

Use the CREATE WRAPPER statement to specify the wrapper that will be used to access table-structured files. Wrappers are mechanisms that federated servers use to communicate with and retrieve data from data sources. For example, run the following statement to register a wrapper called `laboratory_flat_files` which is associated with the library `liblsfile.a`.

```
CREATE WRAPPER laboratory_flat_files LIBRARY 'liblsfile.a'
```

In this example, `laboratory_flat_files` is the name chosen for the wrapper. This name must be unique within the database in which it is being registered.

In this example, the required library name for the table-structured file wrapper is `liblsfile.a`. The library name is installed as `liblsfile.a` by default, but it might have been customized during installation. Check with your system administrator for the correct name.

For more information on the CREATE WRAPPER statement, see the *DB2 SQL Reference*.

Step 2: (Optional) Setting the DB2_DJ_COMM environment variable

To improve performance when table-structured files are accessed, set the `DB2_DJ_COMM` environment variable. This variable determines whether the federated server loads the wrapper upon initialization. Set the `DB2_DJ_COMM` environment variable to include the wrapper library that corresponds to the wrapper that you specified in “Step 1: Registering the wrapper”. For example:

```
export DB2_DJ_COMM='liblsfile.a'
```

Ensure that there are no spaces on either side of the equal sign (=).

For more information about the `DB2_DJ_COMM` environment variable see the *DB2 Administration Guide*.

Step 3: Registering the server

Use the CREATE SERVER statement to define servers that will access table-structured files. A server can be configured to access either sorted or unsorted table-structured files, but a single server cannot be used for both. However, you can define one server to manage sorted table-structured files, and another server to manage unsorted table-structured files. For example:

```
CREATE SERVER biochem_lab TYPE SORTED VERSION 1.0 WRAPPER laboratory_flat_files  
OPTIONS (NODE 'biochem_node1')
```

In this example, `biochem_lab` is the name assigned to the table-structured file server. The name must be unique to the database in which it is being registered.

The `TYPE` keyword is required. It is used to determine which search algorithm is used by the server. Specify `SORTED` or `UNSORTED` depending on the data source. For more information on sorted and unsorted file types, see “Types of table-structured files” on page 7.

`VERSION` is required. It must be set to 1.0.

The wrapper name, in this example, is `laboratory_flat_files`. This is the name previously specified on the `CREATE WRAPPER` statement.

The `NODE` option is required. It is the name given to the local node. It can contain any text string you like.

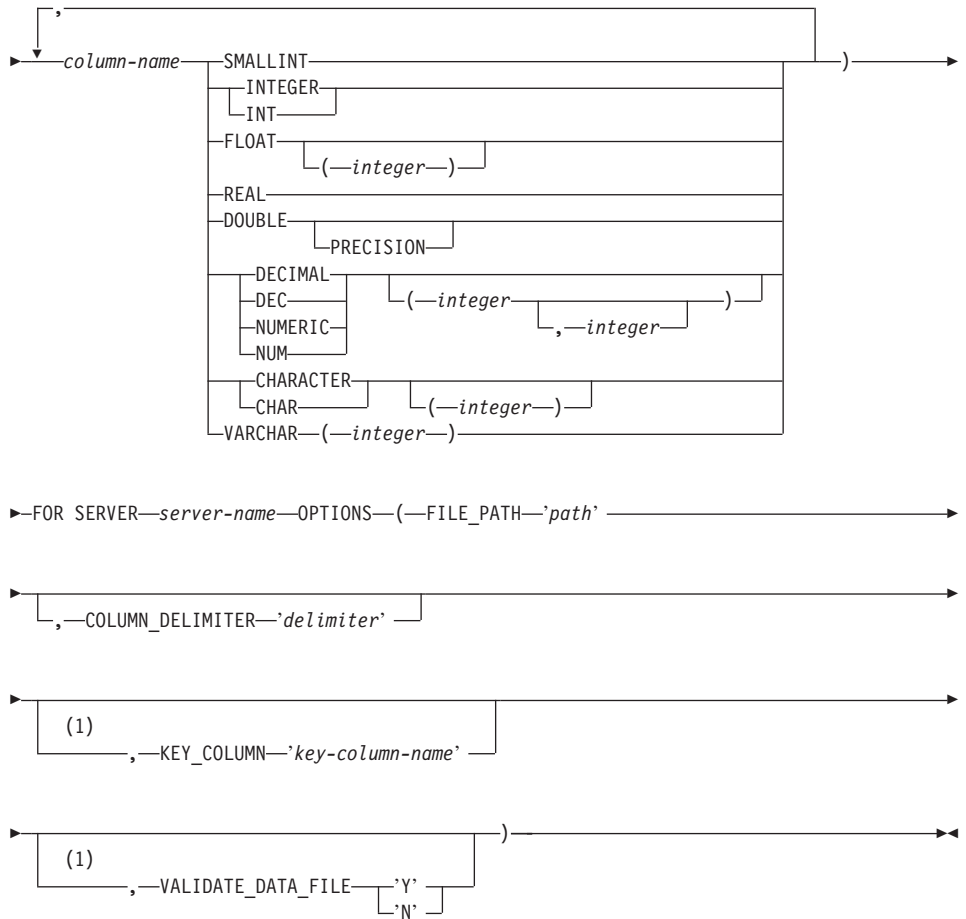
For more information about the `CREATE SERVER` statement, see the *DB2 SQL Reference*.

Step 4: Registering nicknames

Use the `CREATE NICKNAME` statement to register a nickname for each table-structured file that you want to access using any of the servers that you registered in “Step 3: Registering the server” on page 9. Nicknames are used when you refer to a table-structured file in a query.

The syntax for the `CREATE NICKNAME` statement is:

►► `CREATE NICKNAME—nickname—(` 



Notes:

1 Optional for sorted files only.

nickname

A unique nickname for the table-structured file to be accessed. It must be distinct from all other nicknames, tables, and views in the schema in which it is being registered.

column-name

A unique name given to each field in the table-structured file. Follow each column name with its data type. Only columns of type CHAR, VARCHAR, SMALLINT, INTEGER, FLOAT, DOUBLE, REAL, and DECIMAL are supported.

SMALLINT

For a small integer.

INTEGER or INT

For a large integer.

FLOAT(*integer*)

For a single or double precision floating-point number, depending on the value of *integer*. The value of *integer* must be in the range 1 through 53. The values 1 through 24 indicate single precision and the values 25 through 53 indicate double precision.

REAL For single precision floating-point.

DOUBLE or DOUBLE PRECISION

For double precision floating-point.

FLOAT

For double precision floating-point.

DECIMAL(*precision-integer, scale-integer*) or DEC(*precision-integer, scale-integer*)

For a decimal number.

The first integer is the precision of the number; that is, the total number of digits. This value can range from 1 to 31.

The second integer is the scale of the number; that is, the number of digits to the right of the decimal point. This value can range from 0 to the precision of the number.

If precision and scale are not specified, the default values of 5,0 are used.

The words **NUMERIC** and **NUM** can be used as synonyms for **DECIMAL** and **DEC**.

CHARACTER(*integer*) or CHAR(*integer*) or CHARACTER or CHAR

For a fixed-length character string of length *integer*, which can range from 1 to 254. If the length specification is omitted, a length of 1 character is assumed.

VARCHAR(*integer*)

For a varying-length character string of maximum length *integer*, which can range from 1 to 32672.

server-name

Identifies the server you registered in “Step 3: Registering the server” on page 9 that will be used to access the table-structured file. If the file is sorted, the server specified should be of type **SORTED**; otherwise specify a server of type **UNSORTED**.

'path'

The fully qualified path to the table-structured file to be accessed. The data file must be a standard file or a symbolic link, not a pipe or

another non-standard file type. Data files must be readable by the DB2 instance owner. For more information on instance owners, see the *DB2 Administration Guide*.

'delimiter'

The delimiter used to separate columns of the table-structured file. If no column delimiter is defined, the column delimiter defaults to the comma. The column delimiter cannot exist as valid data for a column. For example, a column delimiter of a comma cannot be used if one of the columns contains data with embedded commas.

'key-column-name'

The name of the column in the file that forms the key on which the file is sorted. Use this option for sorted files only. It is case insensitive.

Only single-column keys are supported. The value must be the name of a column defined in the CREATE NICKNAME statement. The column must be sorted in ascending order. If the value is not specified for a sorted server, it defaults to the first column in the nicknamed file.

VALIDATE_DATA_FILE

For sorted files, this option specifies whether the wrapper verifies that the key column is sorted in ascending order. The only valid values for this option are 'Y' or 'N'. The check is done once at registration time. If this option is not specified, then no validation takes place.

The following example shows a CREATE NICKNAME statement for the table-structured file DRUGDATA1.TXT described in “What are table-structured files?” on page 7:

```
CREATE NICKNAME DRUGDATA1(Dcode Integer, Drug CHAR(20), Manufacturer CHAR(20))
FOR SERVER biochem_lab OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',
COLUMN_DELIMITER ',', KEY_COLUMN 'Dcode', VALIDATE_DATA_FILE 'Y')
```

See the *DB2 SQL Reference* for more information about the CREATE NICKNAME statement. For more information about nicknames, see the *DB2 Administration Guide*.

Wrapper limitations and considerations

- The wrapper for table-structured files is only available on AIX.
- Passthru sessions are not allowed with the wrapper.
- Multi-column keys are not allowed.
- Sorted files must be in ascending order only. Descending order is not supported.
- The wrapper does not enforce the NOT NULL constraint, but DB2 does. If you create a nickname and attach a NOT NULL constraint on a column and

then select a row containing a null value for the column, DB2 will issue a SQL0407N error stating that you can't assign a NULL value to a NOT NULL column. The exception to this rule is for sorted servers. The key column for nicknames defined for sorted servers cannot be NULL. If a NULL key column is found for a nickname using a sorted server, the SQL1822N error is issued, stating that the key column is missing.

File limitations and considerations

- Files are limited to one record per line.
- Each record must have an equal number of delimited columns.
- Each record must be terminated by a linefeed character.
- The column delimiter must be consistent throughout the file.
- A null value is represented by two delimiters next to each other.
- The radix character is determined by the RADIXCHAR item of the LC_NUMERIC National Language Support category.
- Sorted data sources must be sorted in ascending order according to the collation sequence for the current locale as defined by the settings in the LC_COLLATE National Language Support category.
- The database codepage must match the file's character set; otherwise, you could get unexpected results.
- Files containing multibyte characters are not supported.
- If a non-numeric field is too long for its column type, the excess data is truncated.
- If a decimal field in the file has more digits after the radix char than are allowed by the scale parameter of its column type, the excess data is truncated.

File access control model

The database management system will access table-structured files with the authority of the DB2 instance owner. The wrapper can only access files that can be read by this user ID (or group ID). The authorization ID of the application (the ID that establishes the connection to the federated database) is not relevant.

Optimization tips and considerations

- The system can search sorted data files much more efficiently than non-sorted files.
- For sorted files, you can improve performance by specifying a value or range for the key column.

- Statistics for nicknames of table-structured files must be updated manually by updating the SYSTAT views. For more information on manually updating SYSTAT views, see the *DB2 Administration Guide*.

Messages

This section lists and describes messages you might encounter while working with the wrapper for table-structured files. For more information on messages, see the *DB2 Message Reference*.

Table 1. Messages issued by the wrapper for table-structured files

Error Code	Message	Explanation
SQL0405N	The numeric literal "<column_name or predicate_value>" is not valid because its value is out of range.	A column in the data file, or a predicate value in an SQL statement, contains a value that is out of the possible range for that data type. Correct the data file or redefine the column to a more appropriate type.
SQL0408N	A value is not compatible with the data type of it's assignment target. Target name is "<column_name>".	A column in the data file contains characters that are invalid for that data type. Correct the data file or redefine the column to a more appropriate type.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "Data source path is NULL".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "Key Column retrieval failure".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "STAT failed on data source. ERRNO = <error_number>".)	Contact IBM Software Support.

Table 1. Messages issued by the wrapper for table-structured files (continued)

Error Code	Message	Explanation
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "No column info found".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "Server parser failed, RC = <parser_return_code>".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "Unsupported operator".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "Cannot identify bind variable".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "Unable to identify query components".)	Contact IBM Software Support.
SQL0901N	The SQL statement failed because of a non-severe system error. Subsequent statements can be processed. (Reason "Could not access data when converting values".)	Contact IBM Software Support.

Table 1. Messages issued by the wrapper for table-structured files (continued)

Error Code	Message	Explanation
SQL1816N	Wrapper "<wrapper_name>" cannot be used to access the "type" of data source ("<type>" """) that you are trying to define to the federated database.	The server type was invalid. The only server types allowed are SORTED or UNSORTED. Change the SQL statement and rerun it.
SQL1822N	Unexpected error code "ERRNO = <error_number>" received from data source "<server_name>". Associated text and tokens are "Unable to read file".	Check the value of the error number. Make sure that the file can be read by the DB2 instance owner. Then rerun the SQL command.
SQL1822N	Unexpected error code "Data Error" received from data source "<server_name>". Associated text and tokens are "Data source is a non-standard file".	The data source file is a directory, socket, or FIFO. Only standard files can be accessed as data source. Change the FILE_PATH option to point to a valid file and reissue the SQL command.
SQL1822N	Unexpected error code "ERRNO = <error_number>" received from data source "<server_name>". Associated text and tokens are "File open error".	The wrapper was unable to open the file. Check the error number to determine why the error occurred. Correct the problem with the data source and reissue the SQL command.
SQL1822N	Unexpected error code "Data Error" received from data source "<server_name>". Associated text and tokens are "Key column missing".	A record retrieved from the data source was missing the key field. The key column must not be null. Correct the data, or register the file with an unsorted server.
SQL1822N	Unexpected error code "Data Error" received from data source "<server_name>". Associated text and tokens are "File not sorted".	The file was not sorted on the key column. Do one of the following: change the KEY_COLUMN option to point to the correct column; resort the data file; or register the nickname with an unsorted server.

Table 1. Messages issued by the wrapper for table-structured files (continued)

Error Code	Message	Explanation
SQL1822N	Unexpected error code "Data Error" received from data source "<server_name>". Associated text and tokens are "Key exceeds definition size".	The key column field read from the data source was larger than the DB2 column definition which could cause the wrapper search routines to function incorrectly. Correct the data or correct the nickname definition, and reregister the nickname.
SQL1822N	Unexpected error code "Data Error" received from data source "<server_name>". Associated text and tokens are "Line in data file exceeds 32k".	A line in the data file exceeded the maximum line length allowed by the wrapper. The line length cannot be greater than 32768. Shorten the length of the line in the data file.
SQL1823N	No data type mapping exists for data type "<data_type>" from server "<server_name>".	Then nickname was defined with an unsupported data type. Redefine the nickname with using only supported data types.
SQL1881N	"<option_name>" is not a valid "<component>" option for "<object_name>".	The listed value is not a valid option for the listed object. Remove or change the invalid option then resubmit the SQL statement.
SQL1882N	The "Nickname" option "COLUMN_DELIMITER" cannot be set to "<delimiter>" for "<nickname_name>".	The column delimiter was more than one character long. Redefine the option with a single character. Then rerun the SQL statement command.
SQL1882N	The "Nickname" option "KEY_COLUMN" cannot be set to "<column_name>" for "<nickname_name>".	The column selected as the key column is not defined for this nickname. Correct the KEY_COLUMN option to be one of the sorted columns for this nickname, then reissue the SQL command.
SQL1882N	The "Nickname" option "VALIDATE_DATA_FILE" cannot be set to "<option_value>" for "<nickname_name>".	The option value was invalid. Valid values are "Y" or "N". Correct the option and register the nickname again.
SQL1883N	"<option_name>" is a required "<component>" option for "<object_name>".	A required option for the wrapper was missing from the SQL statement. Add the required option and resubmit the SQL statement.

Table 1. Messages issued by the wrapper for table-structured files (continued)

Error Code	Message	Explanation
SQL30090N	Operation invalid for application execution environment. Reason code = "21".	You attempted a passthru session. The table-structured file wrapper does not support passthru sessions.

Notices

IBM may not offer the products, services, or features discussed in this document in all 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 give 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:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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 Canada Limited
Office of the Lab Director
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADA

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 information 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.

This information may contain 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 may contain sample application programs in source language, which illustrates 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.

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.

Trademarks

The following terms, which may be denoted by an asterisk(*), are trademarks of International Business Machines Corporation in the United States, other countries, or both.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

The following terms are trademarks or registered trademarks of other companies:

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

Java or all Java-based trademarks and logos, and Solaris are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Tivoli and NetView are trademarks of Tivoli Systems Inc. in the United States, other countries, or both.

UNIX is a registered trademark in the United States, other countries or both and is licensed exclusively through X/Open Company Limited.

Other company, product, or service names, which may be denoted by a double asterisk(**) may be trademarks or service marks of others.

Index

C

CREATE NICKNAME statement,
table-structured files 10
CREATE SERVER statement,
table-structured files 9
CREATE WRAPPER statement,
table-structured files 9

D

DB2_DJ_COMM environment
variable 9
DB2 Life Sciences Data Connect
definition 1
installing on AIX 5
DiscoveryLink 2

F

federated database system 1

L

life sciences data sources 1

M

messages, table-structured files 15

T

table-structured files
accessing with DB2 Life Sciences
Data Connect 8
adding to a federated server
CREATE NICKNAME
statement 10
CREATE SERVER
statement 9
CREATE WRAPPER
statement 9
registering nicknames 10
registering the server 9
registering the wrapper 9
example 7
file access control model 14
limitations and considerations
file 14
wrapper 13
messages 15
optimization 14
overview 7
types
sorted 7
unsorted 7

W

wrapper
definition 2
table-structured files 7

Contacting IBM

If you have a technical problem, please review and carry out the actions suggested by the *Troubleshooting Guide* before contacting DB2 Customer Support. This guide suggests information that you can gather to help DB2 Customer Support to serve you better.

For information or to order any of the DB2 Universal Database products contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-237-5511 for customer support
- 1-888-426-4343 to learn about available service options

Product Information

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

<http://www.ibm.com/software/data/>

The DB2 World Wide Web pages provide current DB2 information about news, product descriptions, education schedules, and more.

<http://www.ibm.com/software/data/db2/library/>

The DB2 Product and Service Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information.

Note: This information may be in English only.

<http://www.elink.ibm.com/pbl/pbl/>

The International Publications ordering Web site provides information on how to order books.

<http://www.ibm.com/education/certify/>

The Professional Certification Program from the IBM Web site provides certification test information for a variety of IBM products, including DB2.

ftp.software.ibm.com

Log on as anonymous. In the directory /ps/products/db2, you can find demos, fixes, information, and tools relating to DB2 and many other products.

comp.databases.ibm-db2, bit.listserv.db2-l

These Internet newsgroups are available for users to discuss their experiences with DB2 products.

On CompuServe: GO IBMDB2

Enter this command to access the IBM DB2 Family forums. All DB2 products are supported through these forums.

For information on how to contact IBM outside of the United States, refer to Appendix A of the *IBM Software Support Handbook*. To access this document, go to the following Web page: <http://www.ibm.com/support/>, and then select the IBM Software Support Handbook link near the bottom of the page.

Note: In some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.



Part Number: CT64WNA



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

(1P) P/N: CT64WNA

