

Database Toolbox™ Release Notes

How to Contact MathWorks



www.mathworks.com Web
comp.soft-sys.matlab Newsgroup
www.mathworks.com/contact_TS.html Technical Support



suggest@mathworks.com Product enhancement suggestions
bugs@mathworks.com Bug reports
doc@mathworks.com Documentation error reports
service@mathworks.com Order status, license renewals, passcodes
info@mathworks.com Sales, pricing, and general information



508-647-7000 (Phone)



508-647-7001 (Fax)



The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098

For contact information about worldwide offices, see the MathWorks Web site.

Database Toolbox™ Release Notes

© COPYRIGHT 2004–2013 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

R2013b

Fast access to ODBC connections via a native ODBC driver	2
table data type support	2

R2013a

fetch function accepts user-defined batch sizes	4
---	---

R2012b

Database Explorer app for interactively exchanging data with databases	6
Functionality Being Removed or Changed	6

R2012a

Execute .SQL Files	8
Improvements to the Database Constructor	8

R2011b

Preferences Now Persistent Across MATLAB Sessions ...	10
Change in Behavior for the update Function	10
Warning and Error ID Changes	10

R2011a

New datainsert Function Exports MATLAB Cell Array
Data into a Database Table 12

R2010b

Now Possible to Import Data into MATLAB Dataset
Object 14

R2010a

New Connection Object Methods 16
Enhanced Error Messages 16
Improved Write Performance 16

R2009b

Bug Fixes

R2009a

Bug Fixes

R2008b

Bug Fixes

R2008a

Bug Fixes

R2007b

Bug Fixes

R2007a

setdbprefs Accepts Structure Input	28
Visual Query Builder Generated M-File Includes Placeholder for Password and Assigns Preferences to Structure	28
Preference Added for Temporary Registry Output; Ensures Full Output for getdatasources	28

R2006b

Enhanced fetch Combines exec with Existing fetch	32
Import Data from Multiple Resultsets	32
Run Stored Procedures to Return Output Parameters	32
Specify Catalog and Schema Using Visual Query Builder	32
Preferences Option to Find Additional Data Sources	33
MATLAB Change to Assignment of Nonscalar Structure Array Fields Might Impact Database Toolbox Users ...	33

R2006a

Bug Fixes

R14SP3

fastinsert Function Added	38
JDBC Drivers Now Supported for Visual Query Builder on Microsoft Windows Systems	38
Define Data Sources from Within the Visual Query Builder	38
setdbprefs Function Enhanced	39
Dynamically Add JDBC Drivers File	39
64-Bit FLOAT for Microsoft SQL Server Software Is Fully Supported	39
Generate M-File from VQB	39
update Function Enhanced to Export Multiple Records ...	40
logintimeout Function Now Supported on Linux Platforms	40

R14SP2

Bug Fixes

R2013b

Version: 5.0

New Features: Yes

Bug Fixes: Yes

Fast access to ODBC connections via a native ODBC driver

Support for native ODBC database connection for Windows® platforms. The native ODBC interface is available only for the command line. To use this interface, see “Using the Native ODBC Database Connection”. The native ODBC interface supports the following functions:

- database
- fetch
- exec
- insert
- fastinsert
- close

table data type support

You can return a `table` data type rather than a cell array. Use the `setdbprefs` command to set the database preference for the `DataReturnFormat` property to `'table'`.

R2013a

Version: 4.1

New Features: Yes

Bug Fixes: Yes

fetch function accepts user-defined batch sizes

setdbprefs is updated with new properties (FetchInBatches and FetchBatchSize) that support fetch when requesting large data.

R2012b

Version: 4.0

New Features: Yes

Bug Fixes: Yes

Database Explorer app for interactively exchanging data with databases

Compatibility Considerations: Yes

`dexplore` starts Database Explorer, which is the Database Toolbox™ GUI for connecting to a database and importing data to the MATLAB® workspace. Alternatively, you can start Database Explorer by selecting **Database Explorer** from the **Database Connectivity and Reporting** section of the **Apps** tab in the MATLAB Toolstrip.

Functionality Being Removed or Changed

Compatibility Considerations: Yes

Functionality	What Happens When You Use It?	Use This Instead	Compatibility Considerations
querybuilder	Warns	<code>dexplore</code>	Continue to use querybuilder for exporting data.

R2012a

Version: 3.11

New Features: Yes

Bug Fixes: Yes

Execute .SQL Files

The new `runsqlscript` function lets you execute SQL commands from a .SQL file on a connected database, and store the results in a cursor array. You can input the results from executing `runsqlscript` to functions that accept cursor array inputs.

Improvements to the Database Constructor

When using a JDBC driver, you can input individual connection properties to the database constructor, `database`.

R2011b

Version: 3.10

New Features: Yes

Bug Fixes: Yes

Preferences Now Persistent Across MATLAB Sessions

Compatibility Considerations: Yes

The preferences you set using the Preference dialog box or the `setdbprefs` function now persist across MATLAB sessions.

Compatibility Considerations

In releases before R2011b, if you changed your preferences during a MATLAB session, these preferences would not remain in the next MATLAB session.

Change in Behavior for the update Function

`update` lets you update images, Booleans, doubles, and strings in a manner consistent with `fastinsert`.

Warning and Error ID Changes

Compatibility Considerations: Yes

Many warning and error IDs have changed from their previous versions. These warnings or errors typically appear during a function call.

Compatibility Considerations

If using warning or error IDs, you might need to change the strings you use. For example, if you turned off a warning for a certain ID, the warning might now appear under a different ID. If you use a `try/catch` statement in your code, replace the old identifier with the new identifier. There is no definitive list of the differences, or of the IDs that changed.

R2011a

Version: 3.9

New Features: Yes

Bug Fixes: Yes

New `datainsert` Function Exports MATLAB Cell Array Data into a Database Table

The new `datainsert` function inserts data from the MATLAB workspace into a database table, much like the `fastinsert` function. The new `datainsert` function is faster.

R2010b

Version: 3.8

New Features: Yes

Bug Fixes: Yes

Now Possible to Import Data into MATLAB Dataset Object

If you have Statistics Toolbox™ installed, you can now return a dataset object rather than a cell array. Use the `setdbprefs` command to set the database preference for the `DataReturnFormat` property to `'dataset'`.

R2010a

Version: 3.7

New Features: Yes

Bug Fixes: Yes

New Connection Object Methods

Several new connection object methods provide database-specific information. The new methods are:

- `database.catalogs`
- `database.columns`
- `database.schemas`
- `database.tables`

See the individual reference pages for more information on how to use these methods.

Enhanced Error Messages

New enhanced error messages provide more information about the error. For example, the 2009b error message `Drivers not Found/Loaded` is now `Drivers not Found/Loaded`. Please verify that login information and database url are valid in 2010b. This error will appear when the driver input is valid but the database URL is invalid.

Improved Write Performance

New bulk insert code templates provide significant performance upgrades.

R2009b

Version: 3.6

New Features: No

Bug Fixes: Yes

R2009a

Version: 3.5.1

New Features: No

Bug Fixes: Yes

R2008b

Version: 3.5

New Features: No

Bug Fixes: Yes

R2008a

Version: 3.4.1

New Features: No

Bug Fixes: Yes

R2007b

Version: 3.4

New Features: No

Bug Fixes: Yes

R2007a

Version: 3.3

New Features: Yes

Bug Fixes: Yes

setdbprefs Accepts Structure Input

The `setdbprefs` function now accepts a structure as input. For example, you can run the following commands to assign values to `s`:

```
s.DataReturnFormat = 'numeric';  
s.ErrorHandling = 'report';
```

You can also do this for other `setdbprefs` properties whose values you want to change. Then set the preferences using the values in `s` by running the command:

```
setdbprefs(s)
```

For more information, see the `setdbprefs` reference page.

Visual Query Builder Generated M-File Includes Placeholder for Password and Assigns Preferences to Structure

When you run a query in the Visual Query Builder and select **File > Generate M-File**, the resulting M-file now includes a placeholder string `password` in the database statement. If a password is required for the connection, such as for connections established via JDBC drivers, substitute the password for the `password` string. If no password is required, the M-file will run as is. For more information, see *About Generated Files*.

The generated M-file assigns values for the preferences to the structure `s`. For more information, see the `setdbprefs` reference page.

Preference Added for Temporary Registry Output; Ensures Full Output for getdatasources

When you use `getdatasources` to view the data sources for your system, ensure that you view all data sources by specifying a temporary, writable, output directory using the new preference, `TempDirForRegistryOutput`. This is useful when you add data sources and do not have write access for the MATLAB current directory, where the toolbox temporarily writes ODBC

registry settings. Without write access, `getdatasources` does not always return data sources you added. In that event, run `setdbprefs` to specify a value for the `TempDirForRegistryOutput` preference, where the value is the full path name to a directory for which you have write access.

R2006b

Version: 3.2

New Features: Yes

Bug Fixes: Yes

Enhanced `fetch` Combines `exec` with Existing `fetch`

The new function, `database.fetch`, executes the specified SQL query and imports results into the MATLAB workspace, given the connection handle `conn`. It is provided for convenience, to combine capabilities of the existing `exec` and `cursor.fetch` functions. In statements and code, do not specify `database.fetch` or `cursor.fetch` but rather, just specify `fetch` with the appropriate objects provided as arguments; the toolbox runs `database.fetch` or `cursor.fetch` as appropriate based on the arguments.

Unlike `cursor.fetch`, `database.fetch` does not return a cursor object on which you can run subsequent Database Toolbox functions, but rather returns all data to a MATLAB variable. For more information about `database.fetch` and how it differs from `cursor.fetch`, see the `fetch` reference page, as well as the `database.fetch` and `cursor.fetch` reference pages.

Import Data from Multiple Resultsets

The new function, `fetchmulti`, imports data into the MATLAB workspace from multiple resultsets, which you retrieve via an `exec` call to a stored procedure that contains two or more `select` statements.

Run Stored Procedures to Return Output Parameters

The new function, `runstoredprocedure`, executes a stored procedure using input parameters specified in a cell array to return output parameters. This allows you to retrieve the value of a variable into a MATLAB variable. `runstoredprocedure` overcomes a limitation of `exec`; when you run a stored procedure via `exec`, you can only retrieve resultsets.

Specify Catalog and Schema Using Visual Query Builder

You can now specify the catalog and schema for a data source using the Visual Query Builder. The `default` is `none`, meaning you do not need to select values for them.

Preferences Option to Find Additional Data Sources

The new `setdbprefs` option, `UseRegistryForSources`, instructs the Visual Query Builder to search the Microsoft® Windows registry to find any ODBC data sources not uncovered using the system `ODBC.INI`.

MATLAB Change to Assignment of Nonscalar Structure Array Fields Might Impact Database Toolbox Users

Compatibility Considerations: Yes

In Version 7.3 (R2006b) of the MATLAB software, a change was made to how a nonscalar structure array field is assigned to a single MATLAB variable. For more information, see [Assigning Nonscalar Structure Array Fields to a Single Variable](#) in the MATLAB Release Notes.

Compatibility Considerations

As a result of this change in the MATLAB software, you may need to modify your Database Toolbox M-files.

R2006a

Version: 3.1.1

New Features: No

Bug Fixes: Yes

R14SP3

Version: 3.1

New Features: Yes

Bug Fixes: Yes

fastinsert Function Added

There is a new function, `fastinsert`, that you can use instead of the `insert` function to export data about three times more quickly than `insert`. It also allows exporting for all object types, so that any data you can retrieve from a database you now can export to the database, including binary objects.

While there are no known problems with `fastinsert`, if you receive unexpected results, return to using `insert` and report the problem with `fastinsert` via Technical Support.

Note that the Visual Query Builder `insert` feature uses the `insert` function instead of `fastinsert`.

JDBC Drivers Now Supported for Visual Query Builder on Microsoft Windows Systems

You now can use the Visual Query Builder (VQB) with JDBC drivers on Windows platforms. Previously, only ODBC drivers were supported.

The `confds` function now displays an enhanced dialog box you use to define JDBC data sources. With it, you save and load data source information via MATLAB MAT-files.

For details, see [Setting Up JDBC Data Sources](#) in the Database Toolbox documentation.

Define Data Sources from Within the Visual Query Builder

The Visual Query Builder now includes two new items under the **Query** menu:

- **Define ODBC Data Source**—Directly access your Windows ODBC Data Source Administrator dialog box where you define ODBC data sources.
- **Define JDBC Data Source**—Access the Define JDBC Data Source dialog box for defining JDBC data sources to use with the VQB. The

function equivalent is `confds`. When you define a JDBC data source, the information is saved in a MAT-file so you can use it again in a later session. Later, open the MAT-file using the Define JDBC Data Source dialog box, or using `setdbprefs('JDBCDataSourceFile','fullpathtomatfile')`.

For details, see *Configuring Your Environment in the Database Toolbox* documentation.

setdbprefs Function Enhanced

New arguments are supported for defining the JDBC data source MAT-file. For details, see the `setdbprefs` reference page.

Dynamically Add JDBC Drivers File

You can dynamically add a JDBC drivers file to the MATLAB Java[®] classpath using the MATLAB `javaaddpath` function. You can use this method instead of adding a pointer to the JDBC drivers file in your `classpath.txt` file. The advantage of using `javaaddpath` is that you do not have to restart the MATLAB software session after running the `javaaddpath` statement. The disadvantage is that this only applies to the current session and so you need to run the `javaaddpath` statement in each new session. For details, see *Setting Up JDBC Data Sources in the Database Toolbox* documentation.

64-Bit FLOAT for Microsoft SQL Server Software Is Fully Supported

You now can retrieve 64-bit FLOAT data using Microsoft SQL Server[®] software.

Generate M-File from VQB

After running a query using the Visual Query Builder, you can generate an M-file consisting of Database Toolbox functions that perform the query. This is useful if you know how to run queries with the VQB and want to determine the equivalent functions, particularly the SQL statements in `exec` and `insert`.

update Function Enhanced to Export Multiple Records

The update function has been enhanced so that you can export multiple records based on different where clauses. The number of where clauses must equal the number of records in the array of data you are exporting. For details, see the reference page for update.

logintimeout Function Now Supported on Linux Platforms

The logintimeout function is now supported on Linux® platforms.

R14SP2

Version: 3.0.2

New Features: No

Bug Fixes: Yes