

Database Toolbox Release Notes

Summary by Version	1
Version 3.3 (R2007a) Database Toolbox	4
Version 3.2 (R2006b) Database Toolbox	6
Version 3.1.1 (R2006a) Database Toolbox	9
Version 3.1 (R14SP3) Database Toolbox	10
Version 3.0.2 (R14SP2) Database Toolbox	13
Version 3.0.1 (R14SP1) Database Toolbox	14
Version 3.0 (R14) Database Toolbox	15
Compatibility Summary for Database Toolbox	18

Summary by Version

This table provides quick access to what's new in each version. For clarification, see "About Release Notes" on page 1.

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Latest Version V3.3 (R2007a)	Yes Details	No	Bug Reports Includes fixes	Printable Release Notes: PDF Current product documentation
V3.2 (R2006b)	Yes Details	Yes Summary	Bug Reports Includes fixes	No
V3.1.1 (R2006a)	No	No	Bug Reports Includes fixes	No
V3.1 (R14SP3)	Yes Details	No	Bug Reports Includes fixes	No
V3.0.2 (R14SP2)	No	No	Bug Reports Includes fixes	No
V3.0.1 (R14SP1)	No	No	Fixed bugs	No
V3.0 (R14)	Yes Details	Yes Summary	Fixed bugs	No

About Release Notes

Use release notes when upgrading to a newer version to learn about new features and changes, and the potential impact on your existing files and practices. Release notes are also beneficial if you use or support multiple versions.

If you are not upgrading from the most recent previous version, review release notes for all interim versions, not just for the version you are installing. For example, when upgrading from V1.0 to V1.2, review the New Features and

Changes, Version Compatibility Considerations, and Bug Reports for V1.1 and V1.2.

New Features and Changes

These include

- New functionality
- Changes to existing functionality
- Changes to system requirements (complete system requirements for the current version are at the MathWorks Web site)
- Any version compatibility considerations associated with each new feature or change

Version Compatibility Considerations

When a new feature or change introduces a known incompatibility between versions, its description includes a **Compatibility Considerations** subsection that details the impact. For a list of all new features and changes that have compatibility impact, see the “Compatibility Summary for Database Toolbox” on page 18.

Compatibility issues that become known after the product has been released are added to Bug Reports at the MathWorks Web site. Because bug fixes can sometimes result in incompatibilities, also review fixed bugs in Bug Reports for any compatibility impact.

Fixed Bugs and Known Problems

MathWorks Bug Reports is a user-searchable database of known problems, workarounds, and fixes. The MathWorks updates the Bug Reports database as new problems and resolutions become known, so check it as needed for the latest information.

Access Bug Reports at the MathWorks Web site using your MathWorks Account. If you are not logged in to your MathWorks Account when you link to Bug Reports, you are prompted to log in or create an account. You then can view bug fixes and known problems for R14SP2 and more recent releases.

The Bug Reports database was introduced for R14SP2 and does not include information for prior releases. You can access a list of bug fixes made in prior versions via the links in the summary table.

Related Documentation at Web Site

Printable Release Notes (PDF). You can print release notes from the PDF version, located at the MathWorks Web site. The PDF version does not support links to other documents or to the Web site, such as to Bug Reports. Use the browser-based version of release notes for access to all information.

Product Documentation. At the MathWorks Web site, you can access complete product documentation for the current version and some previous versions, as noted in the summary table.

Version 3.3 (R2007a) Database Toolbox

This table summarizes what's new in Version 3.3 (R2007a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports Includes fixes	Printable Release Notes: PDF Current product documentation

New features and changes introduced in this version are

- “setdbprefs Accepts Structure Input” on page 4
- “Visual Query Builder Generated M-File Includes Placeholder for Password and Assigns Preferences to Structure” on page 5
- “Preference Added for Temporary Registry Output; Ensures Full Output for getdatasources” on page 5

setdbprefs Accepts Structure Input

The setdbprefs function now accepts a structure as input. For example, assign values to s:

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

and so on for setdbprefs properties whose values you want to change. Then set the preferences using the values in s by running

```
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 actual password. If no password is required, the M-file will run as is. For more information, see “Placeholder for Password Included in Generated M-File”.

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 you will view all data sources by specifying a temporary, writable, output directory via a new preference, `TempDirForRegistryOutput`. This is useful when you add data sources and do not have write access for the MATLAB current directory, where the Database 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 pathname to a directory for which you have write access.

Version 3.2 (R2006b) Database Toolbox

This table summarizes what's new in Version 3.2 (R2006b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes — Details labeled as Compatibility Considerations in descriptions of new features and changes. See also Summary.	Bug Reports Includes fixes	No

New features and changes introduced in this version are

- “Enhanced fetch Combines exec with existing fetch” on page 6
- “Import Data from Multiple Resultsets” on page 7
- “Run Stored Procedures to Return Output Parameters” on page 7
- “Specify Catalog and Schema Using Visual Query Builder” on page 7
- “Preferences Option to Find Additional Data Sources” on page 7
- “MATLAB Change to Assignment of Nonscalar Structure Array Fields Might Impact Database Toolbox Users” on page 7

Enhanced fetch Combines exec with existing fetch

The new function, `database.fetch`, executes the specified SQL query and imports results into MATLAB®, 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; Database 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 variable in MATLAB. 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 MATLAB 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 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

In MATLAB Version 7.3 (R2006b), a change was made to how MATLAB assigns a nonscalar structure array field to a single variable. For more information, see “Assigning Nonscalar Structure Array Fields to a Single Variable” in the MATLAB Release Notes.

Compatibility Considerations

Some users reported that as a result of this change in MATLAB, they needed to modify some of the M-files they use with the Database Toolbox.

Version 3.1.1 (R2006a) Database Toolbox

This table summarizes what's new in Version 3.1.1 (R2006a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
No	No	Bug Reports Includes fixes	No

Version 3.1 (R14SP3) Database Toolbox

This table summarizes what's new in Version 3.1 (R14SP3):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports Includes fixes	No

New features and changes introduced in this version are

- “fastinsert Function Added” on page 10
- “JDBC Drivers Now Supported for Visual Query Builder on Windows” on page 11
- “Define Data Sources from Within the Visual Query Builder” on page 11
- “setdbprefs Function Enhanced” on page 11
- “Dynamically Add JDBC Drivers File” on page 12
- “64-Bit FLOAT for MS SQL Server Fully Supported” on page 12
- “Generate M-File from VQB” on page 12
- “update Function Enhanced to Export Multiple Records” on page 12
- “logintimeout Function Now Supported on Linux” on page 12

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 Windows

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

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 a Data Source for JDBC Drivers” 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 “Setting Up a Data Source” 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 MATLAB 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 “Dynamically Update the MATLAB Java Classpath” in the Database Toolbox documentation.

64-Bit FLOAT for MS SQL Server Fully Supported

You now can retrieve 64-bit FLOAT data from MS SQL Server without any problems.

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

The `logintimeout` function is now supported on Linux platforms, but is not supported on the Macintosh.

Version 3.0.2 (R14SP2) Database Toolbox

This table summarizes what's new in Version 3.0.2 (R14SP2):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
No	No	Bug Reports Includes fixes	No

Version 3.0.1 (R14SP1) Database Toolbox

This table summarizes what's new in Version 3.0.1 (R14SP1):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
No	No	Fixed bugs	No

Version 3.0 (R14) Database Toolbox

This table summarizes what's new in Version 3.0 (R14):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Fixed bugs	No

New features and changes introduced in this version are

- “New Data Types Supported” on page 15
- “Export Using Visual Query Builder (VQB)” on page 16
- “Display Menu in VQB Now Works for Structure and Numeric Formats” on page 16
- “Spaces in VQB Tables and Fields Now Allowed” on page 16
- “Open Data from VQB into Array Editor” on page 16
- “VQB Query Execute Updates Statement” on page 16
- “Database Toolbox Functions Compilable” on page 17
- “logintimeout not Supported on Linux” on page 17
- “BOOLEAN Data Types Now Represented as 1’s and 0’s” on page 17

New Data Types Supported

Database Toolbox now supports Java BINARY (BLOB or Binary Large Objects) and OTHER data types, such as bitmap images and MAT-files. For more information about using these data types in the Visual Query Builder, see “Retrieving BINARY and OTHER Java Data” in the Database Toolbox documentation. For more information about using these data types with functions, see “Retrieving BINARY or OTHER Java SQL Data Types” in the Database Toolbox documentation.

Export Using Visual Query Builder (VQB)

You now can use the VQB to export data from MATLAB, and insert it into new rows in a database. For more information, see “Exporting Data Using the VQB” in the Database Toolbox documentation.

Display Menu in VQB Now Works for Structure and Numeric Formats

You now can use **Display** menu items in the VQB with structure and numeric formats for retrieved data as specified in **Preferences** or with `setdbprefs`. In previous releases, you could use **Display** menu items only for the cell array format.

Spaces in VQB Tables and Fields Now Allowed

You now can use the VQB for tables and fields that contain spaces in their names. They appear inside quotation marks in the VQB lists.

Open Data from VQB into Array Editor

When you double-click a variable in the **Data** field of the VQB, it now opens in the Array Editor. In previous versions, the contents displayed in the Command Window. You still can display the contents in the Command Window by typing the variable name in the Command Window.

VQB Query Execute Updates Statement

In the VQB, when you edit the **SQL statement** field, the query represented by the other fields does not match what is in the **SQL statement** field. When you then select **Query > Save**, it saves exactly that mismatch. When you then load that saved query, you again see the mismatch. When you execute it, it updates, the **SQL statement** field to match the values in the other fields for that query.

Compatibility Considerations

This represents a change from previous versions in which the **SQL statement** field updated to reflect the values in the other fields when you loaded the query, rather than when you executed it.

Database Toolbox Functions Compilable

Because of changes made to the MATLAB Compiler, you can compile Database Toolbox functions to create stand-alone applications that do not require MATLAB to run.

logintimeout not Supported on Linux

On Linux platforms, `logintimeout` is not supported.

BOOLEAN Data Types Now Represented as 1's and 0's

Previously, if you retrieved a `BOOLEAN` field from a database using `fetch`, the values were returned as `true` or `false` strings. Now they are returned as 1's and 0's in MATLAB and have a class of `logical`. For more information, see “`BOOLEAN (MATLAB logical) Data`” in the Database Toolbox documentation, and the `fetch` and `insert` function reference pages.

Compatibility Considerations

If you have M-files that rely upon the string values for data being `true` or `false`, you need to modify your code to use 1 and 0 instead.

Compatibility Summary for Database Toolbox

This table summarizes new features and changes that might cause incompatibilities when you upgrade from an earlier version, or when you use files on multiple versions. Details are provided in the description of the new feature or change.

Version (Release)	New Features and Changes with Version Compatibility Impact
Latest Version V3.3 (R2007a)	None
V3.2 (R2006b)	See the Compatibility Considerations subheading for this change: <ul style="list-style-type: none"> • “MATLAB Change to Assignment of Nonscalar Structure Array Fields Might Impact Database Toolbox Users” on page 7
V3.1.1 (R2006a)	None
V3.1 (R14SP3)	None
V3.0.2 (R14SP2)	None
V3.0.1 (R14SP1)	None
V3.0 (R14)	See the Compatibility Considerations subheading for each of these new features and changes: <ul style="list-style-type: none"> • “VQB Query Execute Updates Statement” on page 16 • “BOOLEAN Data Types Now Represented as 1’s and 0’s” on page 17