
Name

rddbmgr — Tool for managing Rivendell databases

Synopsis

rddbmgr [*OPTIONS*]

Description

rddbmgr(8) is a command-line tool used for managing (creating, modifying and checking) Rivendell databases. When invoked without any options, **rddbmgr**(8) will attempt to connect to the database specified in **rd.conf**(5), print a single line summary of its status to standard output and then exit.

rddbmgr(8) can be invoked with one of three primary modes, as follows:

- `--check` Check an existing Rivendell database for errors and inconsistencies.
- `--create` Initialize a new Rivendell database in an empty database container.
- `--modify` Modify the schema of an existing Rivendell database.

General Options

The following options can be used with any of the commands:

<code>--mysql-hostname=<i>hostname</i></code>	Connect to the database on <i>hostname</i> . Default is to use the corresponding value found in the [mysql] section of rd.conf (5).
<code>--mysql-loginname=<i>username</i></code>	Connect to the database as user <i>username</i> . Default is to use the corresponding value found in the [mysql] section of rd.conf (5).
<code>--mysql-password=<i>password</i></code>	Connect to the database using password <i>password</i> . Default is to use the corresponding value found in the [mysql] section of rd.conf (5).
<code>--mysql-database=<i>dbname</i></code>	Connect to the database named <i>dbname</i> . Default is to use the corresponding value found in the [mysql] section of rd.conf (5).
<code>--mysql-driver=<i>driver</i></code>	Connect to the database using the <i>driver</i> Qt driver. Default is to use the corresponding value found in the [mysql] section of rd.conf (5).
<code>--mysql-engine=<i>engine</i></code>	Use table engine <i>engine</i> when creating new DB tables. Default is to use the corresponding value found in the [mysql] section of rd.conf (5).
<code>--print-progress</code>	Print a message to standard error giving current and elapsed time whenever the database schema version is altered. Useful only for debugging.

Consistency Checking and Audio Recovery

The `--check` switch causes **rddbmgr**(8) to run a series of consistency checks of an existing database and audio store. The database to be checked must be at the native schema of the installed Rivendell version.

The following options can be used in conjunction with `--check`:

<code>--check-log-line-ids</code>	Disable all checks. Add check for log line IDs. Specifically, verify that all IDs are valid (>0) and unique within the context of the given log.						
<code>--check-strings</code>	Check all database fields of type VARCHAR for null characters and then exit.						
<code>--dump-cuts-dir=<i>dir-name</i></code>	Move any "orphaned" audio files found to the <i>dir-name</i> directory.						
<code>--no</code>	Automatically answer all interactive prompts with No .						
<code>--orphan-group=<i>grp-name</i></code>	Move carts with missing/invalid GROUP information to the <i>grp-name</i> group.						
<code>--orphaned-audio</code>	Disable all checks. Add check for orphaned audio files.						
<code>--orphaned-carts</code>	Disable all checks. Add check for orphaned carts.						
<code>--orphaned-cuts</code>	Disable all checks. Add check for orphaned cuts.						
<code>--orphaned-tracks</code>	Disable all checks. Add check for orphaned voice tracks.						
<code>--rehash=<i>target</i></code>	Generate SHA-1 hashes from the specified files in the audio store and validate them against the values stored in the database. This is an inherently expensive operation that may require a significant amount of time to complete as each audio file to be checked must be opened and read in its entirety.						
	The <i>target</i> value can be one of the following:						
	<table> <tr> <td>Cart number --e.g. 123456</td> <td>Check hashes for all cuts within the specified cart.</td> </tr> <tr> <td>Cart/cut specification --e.g. 123456_001</td> <td>Check the hash for the specified cut.</td> </tr> <tr> <td>ALL</td> <td>Check hashes for all cuts in the audio store.</td> </tr> </table>	Cart number --e.g. 123456	Check hashes for all cuts within the specified cart.	Cart/cut specification --e.g. 123456_001	Check the hash for the specified cut.	ALL	Check hashes for all cuts in the audio store.
Cart number --e.g. 123456	Check hashes for all cuts within the specified cart.						
Cart/cut specification --e.g. 123456_001	Check the hash for the specified cut.						
ALL	Check hashes for all cuts in the audio store.						
<code>--relink-audio=<i>dir-name</i></code>	Scan the files in the directory <i>dir-name</i> and copy/rename them into the Rivendell audio store if their SHA-1 hash matches that for one or more Rivendell cuts.						
<code>--relink-audio-move</code>	When given along with <code>--relink-audio</code> , move the audio data to the Rivendell audio store rather than copying it --i.e. copy the audio data, then delete the original.						
<code>--yes</code>	Automatically answer all interactive prompts with Yes .						

Database Creation

The `--create` switch causes **rddbmgr(8)** to initialize a new Rivendell database at the location specified by **rd.conf(5)** and/or the relevant general options (see above). The location so specified must contain an empty database, otherwise **rddbmgr(8)** will exit with an error.

When creating an empty database for initialization by **rddbmgr(8)**, the following permissions should be enabled:

- Select
- Insert
- Update
- Delete
- Create
- Drop
- References
- Index
- Alter
- Create Temporary Tables
- Lock Tables

The following options can be used in conjunction with `--create`:

<code>--generate-audio</code>	In addition to initializing a database, also generate the audio for the test-tone cart in the audio store.
<code>--set-schema=<i>schema</i></code>	Create a database with a schema of <i>schema</i> [integer], which must be between 242 and the native schema (inclusive).
<code>--set-version=<i>version</i></code>	Create a database with a schema appropriate for Rivendell version <i>version</i> [X.Y.Z], which must be between 2.10.0 and the native Rivendell version (inclusive).

If neither `--set-schema` nor `--set-version` options are given, the database created will have the native schema of the installed version of Rivendell.

Database Modification

The `--modify` switch causes **rddbmgr(8)** to modify the schema of the indicated database, in accordance with the following options:

<code>--set-schema=<i>schema</i></code>	Change the database schema to <i>schema</i> [integer], which must be between 242 and the native schema (inclusive).
<code>--set-version=<i>version</i></code>	Change the database schema to that appropriate for Rivendell version <i>version</i> [X.Y.Z], which must be between 2.10.0 and the native Rivendell version (inclusive).

If neither `--set-schema` nor `--set-version` options are given, the database will be changed to native schema of the installed version of Rivendell.

Warning

When reverting the schema of a database, the contents of fields that do not exist in the reverted schema will be irrecoverably lost!

Warning

Schema versions earlier than 297 (Rivendell version 3.0.0) utilize the Latin1 character set; reverting a database containing non-Latin1 characters to these schema versions will cause irreversible corruption of those characters!

See Also

[rdmarkerset\(8\)](#)