XBMC databases

From XBMC

Development ◂ XBMC databases

See also: ***XBMC databases/Versions***

This page or section **may require cleanup, updating, spellchecking, reformatting** and/or **updated images**. Please improve this page if you can. The discussion page may contain suggestions.

INCOMPLETE:
This page or section is **incomplete**. Please add information or correct uncertain data which is marked with a **?**

XBMC uses SQLite (http://www.sqlite.org/) , an open source light-weight SQL database-engine, to store all its library related data (Music, Video, and Program databases). By default, the database files (*.db) are stored in The UserData Folder, specifically in userdata/Database.

In addition to indexing media files when activated by user-selected Content settings, XBMC also puts a video in its database if you change any OSD setting while watching it. Resume points are stored in this database as well. These entries are added to the database whether the affected video is part of the Video Library or not.

Contents

1 Using the Databases

The XBMC databases are automatically maintained whenever you use XBMC. You can activate the most powerful database functionality by setting the Content property on all of your media sources, and using XBMC Library Mode. This view mode allows you to browse your media based on the details available in the databases, rather than using simple folder and filenames for details. You can read more about Library Mode for Music and Video files on their respective pages.

Since XBMC maintains the databases on its own, the only time a developer really needs to access the databases is for display information. The following sections discuss how you can access the information contained in the XBMC databases, and give some brief examples of how to use it.

1.1 Building SQL Queries

SQLite queries can be incredibly powerful (and extraordinarily complicated). If you are not already familiar with SQL syntax, it would probably be a good idea to check out a general tutorial, such as this one (http://www.1keydata.com/sql/sql.html) .

For most XBMC development projects, you're going to be doing select statements. "Select" is a SQL command used to gather data (in the form of "rows") out of a SQL database. Your select statement will include:

- A list of all the data fields (columns in the database table) you want for each row.
- A list of all the tables you need to get information from
- A list of comparisons used to narrow down your results. This last component is optional, but it makes sure your results are relevant and is often used to link database entries across multiple tables.

Below are a few sample select statements, so you can see how it works.

This query grabs all of the information for every movie in the Video Library.

```
select * from movie
```

Note that "*" is used to indicate all fields. Also,there is no "where" clause in this statement, so it returns every row in the table.

This query narrows down the results to just those movies released in 2007.

```
select * from movie where c07 = 2007
```

Note that the column containing the movie's release year is labelled simply "c07." The tables further down this page help you find out which columns contain the information you're looking for.

This query example is more semantic. Lists your movies including, in this order, internal ID, internal file ID, rating, movie year, IMDB ID, movie name and movie plot.

```
select idMovie,idFile,c05,c07,c09,c00,c03,c02 from movie;
```

Now the following query is a bit more complex because it joins 3 movie-related tables (movie, files and path) to list a single useful view of your movies. In human language it lists movie ID, movie year, IMDB rating, IMDB ID, movie name and full path of the movie file, ordered by IMDB rating with highest rating appearing first:

```
select idMovie,c07,c05,c09,c00,path.strPath||files.strFilename from movie, files,path where mov
```

You could also use less than or greater than symbols to get newer or older movies.

This query gets just the path and filename of all of the video files from season two of Chuck.

```

```

If you're not familiar with SQL queries, this query probably looks pretty complicated. It serves as a good demonstration of why there are so many tables in the list below, and how to use them. Many of the elements of a TV show's path and filename and used repeatedly, so SQL allows us to save space and speed up our searches by storing each of those elements just once, in one place, and referencing them repeatedly by the same ID.

In this case, the root path that contains your video files is a long string that repeats at the beginning of many files. The name of a TV series, too, is repeated in every single episode of that series, so it makes the most sense to save the series name once (along with all information relevant to the series). We do that in the table tvshow, and every episode of the TV show can access all of that information using just the TV show's ID.

1.2 Accessing the Databases with XBMC Python

Many Python plugins (and some scripts) can use the information in the XBMC database to offer users additional convenience and functionality. The easiest way to access the XBMC database via XBMC Python is using JSON RPC

2 The Music Library

This database contains all information concerning music files that you've added to the Music Library. It is used in the Music portion of XBMC. The following information is for version 32.

The music database is stored in userdata/Database/MyMusicXX.db, where XX is the version number.

2.1 Views

Views are standard queries, often long or complicated queries saved in the database for convenience. The views below allow you to easily access all the information about songs and albums in the Music Library, across all the linking tables.

2.1.1 albumview

A view that joins album to artist, genre, thumb, and albuminfo.

Column Name	Data Type	Description
idAlbum	integer	ID from Album Table
strAlbum	varchar(256)	Album Name
strArtists	text	Album Artists
strGenres	text	Album Genres
iYear	integer	Album Release Year
idAlbumInfo	integer	ID from AlbumInfo Table
strMoods	integer	Album Moods
strStyles	text	Album Styles
strReview	text	Album Review
strLabel	text	Album Label
strType	text	Album Type
strImage	text	URL for Album Image
iRating	integer	Album Rating
bCompilation	integer	[unknown]
iTimesPlayed	integer	# of Times Played

2.1.2 artistview

A view that joins artist to artistinfo.

Column Name	Data Type	Description
idArtist	integer	ID from Artist Table
strArtist	varchar(256)	Artist Name
strBorn	text	Artist Birthday
strFormed	text	Band Formation Date
strGenres	text	Artist Genres
strMoods	text	Artist Moods
strStyles	text	Artist Styles
strInstruments	text	Artist Instruments
strBiography	text	Artist Biography
strDied	text	Artist Date Died
strDisbanded	text	Band Disbanded Date
strYearsActive	text	Years Artist is Active
strImage	text	URL of Artist Image
strFanart	text	URL of Fanart

2.1.3 songview

A view that joins song to album, path, artist, genre, thumb, and karaokedata.

Column Name	Data Type	Description
idSong	integer	ID from Song Table
strArtists	text	Song Artists
strGenres	text	Song Genres
strTitle	varchar(512)	Song Title
iTrack	integer	Song Track Number
iDuration	integer	Song Duration in seconds
iYear	integer	Song's Release Year
dwFileNameCRC	text	[unknown]
strMusicBrainzTrackID	text	Song's MusicBrainz Track ID
strMusicBrainzArtistID	text	Song's MusicBrainz Artist ID
strMusicBrainzAlbumID	text	Song's MusicBrainz Album ID
strMusicBrainzAlbumArtistID	text	Song's MusicBrainz Album Artist ID
strMusicBrainzTRMID	text	[unknown]
iTimesPlayed	integer	# of Times Played
iStartOffset	integer	[unknown]
iEndOffset	integer	[unknown]
idThumb	integer	[unknown]
lastplayed	varchar(20)	Date & Time Last Played
rating	char	Song Rating
comment	text	Song Comment
idAlbum	integer	ID from Album Table
strAlbum	varchar(256)	Album Name
strPath	varchar(512)	Media Path
iKaraNumber	integer	[unknown]
iKaraDelay	integer	[unknown]
strKaraEncoding	text	[unknown]
bCompilation	integer	[unknown]
strAlbumArtists	text	Album Artists

2.2 Tables

2.2.1 album

This table contains basic album information.

Column Name	Data Type	Description
idAlbum	integer	Primary Key
strAlbum	varchar(256)	Album Name
strArtists	text	Artist Name
strGenres	text	Album Genre
iYear	integer	Album Year
idThumb	integer	Foreign key to thumb table
bCompilation	integer	[unknown]

2.2.2 album_artist

Column Name	Data Type	Description
idArtist	integer	Primary Key
idAlbum	integre	Foreign key to album table
boolFeatured	integer	[unknown]
iOrder	integer	[unknown]

2.2.3 album_genre

This links albums with their genres.

Column Name	Data Type	Description
idGenre	integer	Foreign key to genre(music) table
idAlbum	integer	Foreign key to album table
iOrder	integer	[unknown]

2.2.4 albuminfo

This table contains additional information about an album, such as Rating, Moods, Styles, Reviews, Image URL, and type.

Column Name	Data Type	Description
idAlbumInfo	integer	Primary Key
idAlbum	integer	Foreign key to album table
iYear	integer	Album Year
strMoods	text	Album Moods
strStyles	text	Album Styles
strThemes	text	Album Themes
strReview	text	Album Review
strImage	text	URL of Album Image
strLabel	text	Album Label
strType	text	Album Type
iRating	integer	Album Rating

2.2.5 albuminfosong

This table links songs to albums and stores the duration of each track.

Column Name	Data Type	Description
idAlbumInfoSong	integer	Primary Key
idAlbumInfo	integer	Foreign key to albuminfo table
iTrack	integer	Track #
strTitle	text	Song Title
iDuration	integer	Song Duration in seconds

2.2.6 art

Column Name	Data Type	Description
art_id	integer	Primary Key
media_id	integer	[unknown]
media_type	text	[unknown]
type	text	[unknown]
url	text	[unknown]

2.2.7 artist

This table stores the name of each artist.

Column Name	Data Type	Description
idArtist	integer	Primary Key
strArtist	varchar(256)	Artist Name

2.2.8 artistinfo

This table stores relevant information about each artist, such as when they were born, Fan Art URL's, biographical information, etc.

Column Name	Data Type	Description
idArtistInfo	integer	Primary Key
idArtist	integer	Foreign key to artist table
strBorn	text	Artist Birthday
strFormed	text	Band Formation Date
strGenres	text	Artist Genres
strMoods	text	Artist Moods
strStyles	text	Artist Styles
strInstruments	text	Artist's Instruments
strBiography	text	Artist's Biography
strDied	text	Artist Date Died
strDisbanded	text	Band Disbanded Date
strYearsActive	text	Years Artist is Active
strImage	text	URL of Artist Image
strFanart	text	URL of Fanart

2.2.9 content

This table is related to the scraper.

Column Name	Data Type	Description
strPath	text	[unknown]
strScraperPath	text	[unknown]
strContent	text	[unknown]
strSettings	text	[unknown]

2.2.10 discography

Links albums to artists with the year produced.

Column Name	Data Type	Description
idArtist	integer	Foreign key to artist table
strAlbum	text	Album Name
strYear	text	Year Album Produced.

2.2.11 exartistablum

Links artists to albums

2.2.12 exartistsong

Links artists to songs

2.2.13 exgenrealbum

Links genres to albums

2.2.14 exgenresong

Links genres to songs

2.2.15 genre(music)

This table contains genre titles.

Column Name	Data Type	Description
idGenre	integer	Primary Key
strGenre	varchar(256)	Genre Name

2.2.16 karaokedata

This table contains karaoke specific information for certain songs

Column Name	Data Type	Description
iKaraNumber	integer	[unknown]
idSong	integer	Foreign key to song table
iKaraDelay	integer	[unknown]
strKaraEncoding	text	[unknown]
strKaralyrics	text	[unknown]
strKaraLyFileCRC	text	[unknown]

2.2.17 path(music)

This table contains paths and hashes of files in the Music Database.

Column Name	Data Type	Description
idPath	integer	Primary Key
strPath	varchar(512)	Media Paths
strHash	text	[unknown]

2.2.18 song

This table contains song information such as Name, Track Title, MusicBrainz information, times played, last played, rating, etc.

Column Name	Data Type	Description
idSong	integer	Primary Key
idAlbum	integer	Foreign key to album table
idPath	integer	Foreign key to path table
strArtists	text	Song Artists
strGenres	text	Song Genres
strTitle	varchar(512)	Song Title
iTrack	integer	Song Track Number
iDuration	integer	Song Duration in seconds
iYear	integer	Song's Release Year
dwFileNameCRC	text	[unknown]
strMusicBrainzTrackID	text	Song's MusicBrainz Track ID
strMusicBrainzArtistID	text	Song's MusicBrainz Artist ID
strMusicBrainzAlbumID	text	Song's MusicBrainz Album ID
strMusicBrainzAlbumArtistID	text	Song's MusicBrainz Album Artist ID
strMusicBrainzTRMID	text	[unknown]
iTimesPlayed	integer	# of Times Played
iStartOffset	integer	[unknown]
iEndOffset	integer	[unknown]
idThumb	integer	Foreign key to thumb table
lastplayed	varchar(20)	Date & Time Last Played
rating	char	Song Rating
comment	text	Song Comment

2.2.19 song_artist

This table links songs to artists.

Column Name	Data Type	Description
idArtist	integer	Foreign key to artist table
idSong	integer	Foreign key to song table
boolFeatured	integer	Is Artist Featured on song?
iOrder	integer	[unknown]

2.2.20 song_genre

This table links the songs with their genre.

Column Name	Data Type	Description
idGenre	integer	Foreign key to genre(music) table
idSong	integer	Foreign key to song table
iOrder	integer	[unknown]

2.2.21 thumb

Column Name	Data Type	Description
idThumb	integer	Primary Key
strThumb	text	[unknown]

2.2.22 version(music)

Column Name	Data Type	Description
idVersion	integer	Version of the music database
idCompressCount	integer	Number of times database has been compressed

3 The Video Library

This database contains all information concerning TV shows, movies, and music videos. It is used in the Videos portion of XBMC. The following information is for version 75.

The video database is stored in userdata/Database/MyVideosXX.db, where XX is the version number.

3.1 Views

Views are standard queries, often long or complicated queries saved in the database for convenience. The views below allow you to easily access all the information about each of the main media types in the Video Library, across all the linking tables.

3.1.1 episodeview

A view that joins episode to file and tvshow (through tvshowlinkepisode) and path.

Column Name	Data Type	Description
idEpisode	integer	Primary Key
idFile	integer	Foreign key to the files table
c00	text	Episode Title
c01	text	Plot Summary
c02	text	[unknown - listed as Votes]
c03	text	Rating
c04	text	Writer
c05	text	First Aired
c06	text	Thumbnail URL
c07	text	[unknown - listed as Thumbnail URL Spoof, unused?]
c08	text	Has the episode been watched? (unused?)
c09	text	Episode length in minutes
c10	text	Director
c11	text	[unknown - listed as Indentifier]
c12	text	Season Number
c13	text	Episode Number
c14	text	[unknown - listed as Original Title, unused?]
c15	text	Season formatted for sorting
c16	text	Episode formatted for sorting
c17	text	Bookmark
c18	text	Not used
c19	text	Not used
c20	text	Not used
idShow	integer	Foreign key to the tvshow table
strFileName	text	Full name of file including extension
strPath	text	Path URL
playCount	integer	# of Times Played
lastPlayed	text	Date & Time Last Played
dateAdded	text	Date & Time Added to Library
strTitle	text	Episode Title
strStudio	text	Network
premiered	text	First Aired
mpaa	text	Content Rating
strShowPath	text	Show Path URL
resumeTimeInSeconds	double	Time in seconds of bookmark location

totalTimeInSeconds	double	Time in seconds of the video
idSeason	integer	Foreign key to the seasons table

3.1.2 movieview

A view that joins movie to file and path.

Column Name	Data Type	Description
idMovie	integer	Primary Key
idFile	integer	Foreign Key to files table
c00	text	Local Movie Title
c01	text	Movie Plot
c02	text	Movie Plot Outline
c03	text	Movie Tagline
c04	text	Rating Votes
c05	text	Rating
c06	text	Writers
c07	text	Year Released
c08	text	Thumbnails
c09	text	IMDB ID
c10	text	Title formatted for sorting
c11	text	Runtime (UPnP devices see this as seconds)
c12	text	MPAA Rating
c13	text	IMDB Top 250 (http://www.imdb.com/chart/top) Ranking
c14	text	Genre
c15	text	Director
c16	text	Original Movie Title
c17	text	[unknown - listed as Thumbnail URL Spoof]
c18	text	Studio
c19	text	Trailer URL
c20	text	Fanart URLs
c21	text	Country (Added in r29886[1] (http://trac.xbmc.org/changeset/29886/trunk))
c23	text	idPath
idSet	integer	Foreign Key to sets table
strSet	text	The name of the set
strFileName	text	Full name of file including extension

strPath	text	Path URL
playCount	integer	# of Times Played
lastPlayed	text	Date & Time Last Played
dateAdded	text	Date & Time Added to Library
resumeTimeInSeconds	double	Time in seconds of bookmark location
totalTimeInSeconds	double	Time in seconds of the video

3.1.3 musicvideoview

A view that joins musicvideo to file and path.

Column Name	Data Type	Description
idMVideo	integer	Primary Key
idFile	integer	Foreign Key to files table
c00	text	Title
c01	text	Thumbnail URL
c02	text	[unknown - listed as Thumbnail URL spoof]
c03	text	Play count (unused?)
c04	text	Run time
c05	text	Director
c06	text	Studios
c07	text	Year
c08	text	Plot
c09	text	Album
c10	text	Artist
c11	text	Genre
c12	text	Track
c13	text	
c14	text	
c15	text	
c16	text	
c17	text	
c18	text	
c19	text	
c20	text	
c21	text	
c22	text	
c23	text	
strFileName	text	Full name of file including extension
strPath	text	Path URL
playCount	integer	# of Times Played
lastPlayed	text	Date & Time Last Played
dateAdded	text	Date & Time Added to Library
resumeTimeInSeconds	double	Time in seconds of bookmark location
totalTimeInSeconds	double	Time in seconds of the video

3.1.4 tvshowview

View that joins tvshow to path. Also produces information about total number of episodes as well as number of watched and unwatched episodes.

Column Name	Data Type	Description
idShow	integer	Primary Key
c00	text	Show Title
c01	text	Show Plot Summary
c02	text	Status
c03	text	Votes
c04	text	Rating
c05	text	First Aired
c06	text	Thumbnail URL
c07	text	[unknown - Spoof Thumbnail URL?]
c08	text	Genre
c09	text	Original Title
c10	text	Episode Guide URL
c11	text	Fan Art URL
c12	text	SeriesId (when using thetvdb.com scraper)
c13	text	Content Rating
c14	text	Network
c15	text	Title formatted for sorting
c16	text	Not Used
c17	text	Not Used
c18	text	Not Used
c19	text	Not Used
c20	text	[unknown]
c21	text	[unknown]
c22	text	[unknown]
c23	text	[unknown]
strPath	text	Path URL
dateAdded	text	Date & Time Added to Library
lastPlayed	text	Date & Time Last Played
totalCount	integer	# of Episodes
watchedcount	integer	# of Times Played
totalSeasons	integer	# of Seasons

3.2 Tables

The information in the Video Library is organized into the following tables. Several large tables (such as episode, movie, settings, and tvshow) contain the bulk of the information, while most of the others are used to link a long string to a common ID key.

3.2.1 actorlinkepisode

This table links actors to episodes and stores role information.

Column Name	Data Type	Description
idActor	integer	Foreign key to actors table
idEpisode	integer	Foreign key to episode table
strRole	text	Role the actor played in this episode

3.2.2 actorlinkmovie

This table links actors to movies and stores role information.

Column Name	Data Type	Description
idActor	integer	Foreign key to actors table
idMovie	integer	Foreign key to movie table
strRole	text	Role the actor played in this movie

3.2.3 actorlinktvshow

This table links actors to TV shows and stores role information.

Column Name	Data Type	Description
idActor	integer	Foreign key to actors table
idShow	integer	Foreign key to tvshow table
strRole	text	Role the actor played in this TV show

3.2.4 actors

This table stores actor, artist, director, and writer information.

Column Name	Data Type	Description
idActor	integer	Primary Key
strActor	integer	Name of the actor, artist, director, or writer
strThumb	text	Thumbnail URL

3.2.5 art

This table stores URLs for video art metadata.

Column Name	Data Type	Description
art_id	integer	Primary Key
media_id	integer	The id of the media this piece of art is for
media_type	text	The type of media this art applies to - movie, set, tvshow, season, episode, musicvideo or actor
type	text	The image type - poster, fanart, thumb, banner, landscape, clearlogo, clearart, characterart or discart
url	text	Thumbnail URL

3.2.6 artistlinkmusicvideo

This table links artists to music videos.

Column Name	Data Type	Description
idArtist	integer	Foreign key to actors table
idMVideo	integer	Foreign key to musicvideo table

3.2.7 bookmark

This table stores bookmarks, which are timestamps representing the point in a video where a user stopped playback, an explicit bookmark requested by the user, or an automatically generated episode bookmark.

Column Name	Data Type	Description
idBookmark	integer	Primary Key
idFile	integer	Foreign key to files table
timeInSeconds	double	Time in seconds of bookmark location
totalTimeInSeconds	integer	Time in seconds of the video
thumbNailImage	text	Thumbnail for bookmark
player	text	Player used to store bookmark
playerState	text	Player's internal state in XML
type	integer	Type of bookmark (0=standard, 1=resume, 2=episode

3.2.8 country

This table lists countries.

Column Name	Data Type	Description
idCountry	integer	Primary Key
strCountry	text	Country Name

3.2.9 countrylinkmovie

This table links countries to movies.

Column Name	Data Type	Description
idCountry	integer	Foreign key to country table
idMovie	integer	Foreign key to movie table

3.2.10 directorlinkepisode

This table links directors to TV show episodes.

Column Name	Data Type	Description
idDirector	integer	Foreign key to actors table
idEpisode	integer	Foreign key to episode table
strRole	text	[Appears to be unused]

3.2.11 directorlinkmovie

This table links directors to movies.

Column Name	Data Type	Description
idDirector	integer	Foreign key to actors table
idMovie	integer	Foreign key to movie table

3.2.12 directorlinkmusicvideo

This table links directors to music videos.

Column Name	Data Type	Description
idDirector	integer	Foreign key to actors table
idMVideo	integer	Foreign key to musicvideo table

3.2.13 directorlinktvshow

This table links directors to TV shows.

Column Name	Data Type	Description
idDirector	integer	Foreign key to actors table
idShow	integer	Foreign key to tvshow table

3.2.14 episode

This table stores television episode information. Information concerning the series is stored in tvshow. To link an episode to its parent series, use tvshowlinkepisode.

Column Name	Data Type	Description
idEpisode	integer	Primary Key
c00	text	Episode Title
c01	text	Plot Summary
c02	text	[unknown - listed as Votes]
c03	text	Rating
c04	text	Writer
c05	text	First Aired
c06	text	Thumbnail URL
c07	text	[unknown - listed as Thumbnail URL Spoof, unused?]
c08	text	Has the episode been watched? (unused?)
c09	text	Episode length in minutes
c10	text	Director
c11	text	[unknown - listed as Indentifier]
c12	text	Season Number
c13	text	Episode Number
c14	text	[unknown - listed as Original Title, unused?]
c15	text	Season formatted for sorting
c16	text	Episode formatted for sorting
c17	text	Bookmark
c18	text	Not used
c19	text	Not used
c20	text	Not used
idFile	integer	Foreign key to the files table

3.2.15 files

This table stores filenames and links the path.

Column Name	Data Type	Description
idFile	integer	Primary Key
idPath	integer	Foreign key to path table
strFilename	text	Full name of file including extension
playCount	integer	# of Times Played
lastPlayed	text	Date & Time Last Played
dateAdded	test	Date & Time Added to Library

3.2.16 genre

This table stores genre information. For convenience the contents are duplicated in movie and tvshow, so a join isn't necessary.

Column Name	Data Type	Description
idGenre	integer	Primary Key
strGenre	text	Genre label

3.2.17 genrelinkmovie

This table links genres to movies. (The contents are also stored in movies.c14, though.)

Column Name	Data Type	Description
idGenre	integer	Foreign key to genre table
idMovie	integer	Foreign key to movie table

3.2.18 genrelinkmusicvideo

This table links genres to music videos.

Column Name	Data Type	Description
idGenre	integer	Foreign key to genre table
idMVideo	integer	Foreign key to musicvideo table

3.2.19 genrelinktvshow

This table links genres to TV show. (The contents are also stored in tvshow.c08, though.)

Column Name	Data Type	Description
idGenre	integer	Foreign key to genre table
idShow	integer	Foreign key to tvshow table

3.2.20 movie

This table stores movie information.

Column Name	Data Type	Description
idMovie	integer	Primary Key
c00	text	Local Movie Title
c01	text	Movie Plot
c02	text	Movie Plot Outline
c03	text	Movie Tagline
c04	text	Rating Votes
c05	text	Rating
c06	text	Writers
c07	text	Year Released
c08	text	Thumbnails
c09	text	IMDB ID
c10	text	Title formatted for sorting
c11	text	Runtime (UPnP devices see this as seconds)
c12	text	MPAA Rating
c13	text	IMDB Top 250 (http://www.imdb.com/chart/top) Ranking
c14	text	Genre
c15	text	Director
c16	text	Original Movie Title
c17	text	[unknown - listed as Thumbnail URL Spoof]
c18	text	Studio
c19	text	Trailer URL
c20	text	Fanart URLs
c21	text	Country (Added in r29886[2] (http://trac.xbmc.org/changeset/29886/trunk))
c23	text	idPath
idFile	integer	Foreign Key to files table
idSet	integer	Foreign Key to sets table

3.2.21 movielinktvshow

This table links movies to TV shows.

Column Name	Data Type	Description
idMovie	integer	Foreign key to movie table
idShow	integer	Foreign key to tvshow table

3.2.22 musicvideo

Column Name	Data Type	Description
idMVideo	integer	Primary Key
c00	text	Title
c01	text	Thumbnail URL
c02	text	[unknown - listed as Thumbnail URL spoof]
c03	text	Play count (unused?)
c04	text	Run time
c05	text	Director
c06	text	Studios
c07	text	Year
c08	text	Plot
c09	text	Album
c10	text	Artist
c11	text	Genre
c12	text	Track
c13	text	
c14	text	
c15	text	
c16	text	
c17	text	
c18	text	
c19	text	
c20	text	
c21	text	
c22	text	
c23	text	
idFile	integer	Foreign Key to files table

3.2.23 path

This table stores path information.

Column Name	Data Type	Description
idPath	integer	Primary Key
strPath	text	Path URL
strContent	text	Type of content (tvshows, movies, etc...)
strScraper	text	XML file of scraper used for this path
strHash	text	Hash
scanRecursive	integer	Recursive scan setting
useFolderNames	bool	User folder names setting
strSettings	text	Custom settings used by scraper

3.2.24 seasons

This table stores the links between tv show and seasons.

Column Name	Data Type	Description
idSeason	integer	Primary Key
idShow	integer	Foreign key to tvshow table
season	integer	Season number

3.2.25 sets

This table stores the id and name for movie sets. Sets are linked to movies in the movie table (idSet column).

Column Name	Data Type	Description
idSet	integer	Primary Key
strSet	text	The name of the set

3.2.26 settings

This table stores settings for individual files.

Column Name	Data Type	Description
idFile	integer	Foreign Key to files table
Interleaved	bool	Interleaved
Nocache	bool	NoCache
Deinterlace	bool	Deinterlace
FilmGrain	integer	FilmGrain
ViewMode	integer	ViewMode
ZoomAmount	float	ZoomAmount
PixelRatio	float	PixelRatio
AudioStream	integer	Selected audio stream
SubtitleStream	integer	Selected subtitle stream
SubtitleDelay	float	Amount of delay for subtitles
SubtitleOn	bool	Enable subtitles
Brightness	integer	Brightness
Contrast	integer	Contrast
Gamma	integer	Gamma
VolumeAmplification	float	VolumeAmplification
AudioDelay	float	AudioDelay
OutputToAllSpeakers	bool	OutputToAllSpeakers
ResumeTime	integer	ResumeTime
Crop	bool	Crop
CropLeft	integer	CropLeft
CropRight	integer	CropRight
CropTop	integer	CropTop
CropBottom	integer	CropBottom

3.2.27 stacktimes

This table stores playing times for files (used for playing multi-file videos).

Column Name	Data Type	Description
idFile	integer	Foreign key to files table
times	text	Times

3.2.28 streamdetails

This table contains information regarding codecs used, aspect ratios etc

Column Name	Data Type	Description
idFile	integer	Foreign Key to files table
iStreamType	integer	0 = video, 1 = audio, 2 = subtitles
strVideoCodec	text	Video codex (xvid etc)
fVideoAspect	real	Aspect ratio
iVideoWidth	integer	Width of the video
iVideoHeight	integer	Height of the video
strAudioCodec	text	Audio codec (aac, mp3 etc)
iAudioChannels	integer	Number of audio channels (2 for stereo, 6 for 5.1 etc)
strAudioLanguage	text	Language of the audio track
strSubtitleLanguage	text	Language of the subtitles

3.2.29 studio

This table stores studio information.

Column Name	Data Type	Description
idStudio	integer	Primary Key
strStudio	text	Studio Label

3.2.30 studiolinkmovie

This table links studios to movies.

Column Name	Data Type	Description
idStudio	integer	Foreign key to studio table
idMovie	integer	Foreign key to movie table

3.2.31 studiolinkmusicvideo

This table links studios to music videos.

Column Name	Data Type	Description
idStudio	integer	Foreign key to studio table
idMVideo	integer	Foreign key to movievideo table

3.2.32 studiolinktvshow

This table links studios to tv shows.

Column Name	Data Type	Description
idStudio	integer	Foreign key to studio table
idShow	integer	Foreign key to tvshow table

3.2.33 tag

This stores tags.

Column Name	Data Type	Description
idTag	integer	Primary Key
strTag	integer	Tag

3.2.34 taglinks

This table links tags to various media.

Column Name	Data Type	Description
idTag	integer	Foreign key to tag table
strTag	integer	Foreign key to a media table
media_type	text	Media type for link

3.2.35 tvshow

This table stores information about a television series. Information concerning the shows episodes is stored in episode. To link a TV show to its episodes, use tvshowlinkepisode.

Column Name	Data Type	Description
idShow	integer	Primary Key
c00	text	Show Title
c01	text	Show Plot Summary
c02	text	Status
c03	text	Votes
c04	text	Rating
c05	text	First Aired
c06	text	Thumbnail URL
c07	text	[unknown - Spoof Thumbnail URL?]
c08	text	Genre
c09	text	Original Title
c10	text	Episode Guide URL
c11	text	Fan Art URL
c12	text	SeriesId (when using thetvdb.com scraper)
c13	text	Content Rating
c14	text	Network
c15	text	Title formatted for sorting
c16	text	Not Used
c17	text	Not Used
c18	text	Not Used
c19	text	Not Used
c20	text	[unknown]
c21	text	[unknown]
c22	text	[unknown]
c23	text	[unknown]

3.2.36 tvshowlinkepisode

This table links TV shows (series) to episodes.

Column Name	Data Type	Description
idShow	integer	Foreign Key to tvshow table
idEpisode	integer	Foreign Key to episode table

3.2.37 tvshowlinkpath

This table links a TV show to its path.

Column Name	Data Type	Description
idShow	integer	Foreign key to tvshow table
idPath	integer	Foreign key to path table

3.2.38 version

This table stores database information.

Column Name	Data Type	Description
idVersion	integer	Version of database
idCompressCount	integer	Number of times database has been compressed

3.2.39 writerlinkepisode

This table links writers to TV show episodes.

Column Name	Data Type	Description
idWriter	integer	Foreign key to actors table
idEpisode	integer	Foreign key to episode table

3.2.40 writerlinkmovie

This table links writers to movies.

Column Name	Data Type	Description
idWriter	integer	Foreign key to actors table
idMovie	integer	Foreign key to movie table

4 The View Modes Database

XBMC can track a user's View Mode for every path, so you could browse movies using DVD Thumbs and browse TV shows using Fanart. This database contains the last view and sorting method a user chose for each path while navigating XBMC.

The View Modes database is stored in userdata/Database/ViewModes.db.

4.1 Tables

The View Modes database uses only two tables. Most of the useful information is in view.

4.1.1 version

This table stores database information.

Column Name	Data Type	Description
idVersion	integer	Version of database
idCompressCount	integer	Number of times database has been compressed

4.1.2 view

This table stores details on the saved view mode for all known paths.

Column Name	Data Type	Description
idView	integer	Primary Key
window	integer	Window GUI ID
path	text	Path to trigger the view on
viewMode	integer	View Mode
sortMethod	integer	ID of sort method
sortOrder	integer	Sort order (ascending or descending)

5 See also

- Version table
- HOW-TO:Share libraries using MySQL

Retrieved from "http://wiki.xbmc.org/index.php?title=XBMC_databases&oldid=67950"
Categories: Cleanup | Incomplete | Development
