



Last indexed: 27 August 2025 (c12f7a)

PadelFlow Overview

Core Application Architecture

Server Setup and Configuration

User Roles and Authentication

Tournament Management
Features

Real-time Features

Database Layer

SQLite Database Management

Database Extensions

Qt Framework Components

Image Format Support

Development Environment

IDE Configuration

Debugging Setup

Project Configuration

Image Format Support

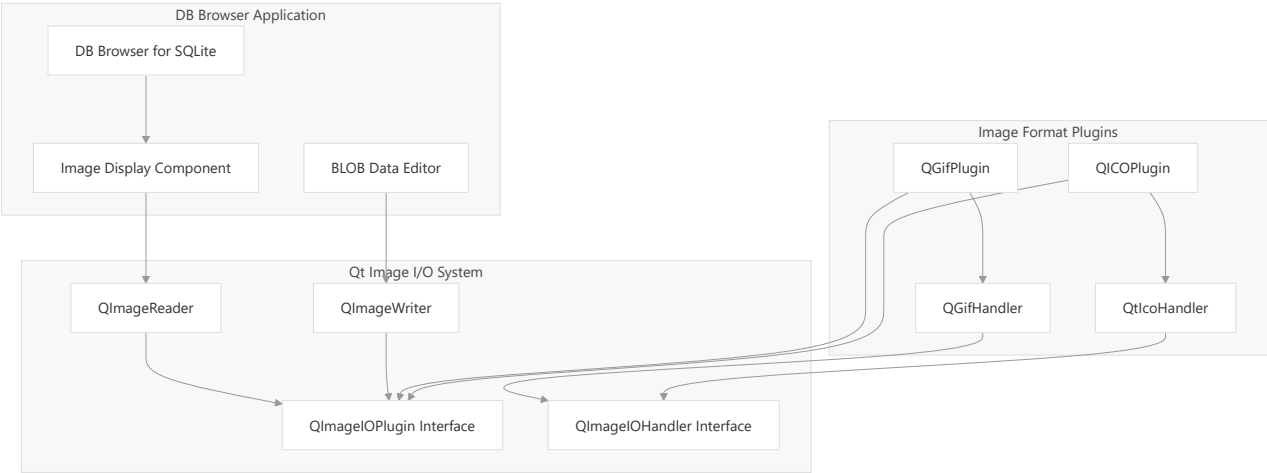
Relevant source files

This document describes the image format plugins that extend DB Browser for SQLite's capability to handle various image file formats. These plugins enable the application to display and manipulate image data stored as BLOBs in SQLite databases.

For information about the overall Qt framework components, see [Qt Framework Components](#). For details on the core SQLite database functionality, see [SQLite Database Management](#).

Plugin Architecture Overview

The image format support is implemented through Qt's image I/O plugin system, which provides a modular architecture for extending image format capabilities. Each format is handled by a dedicated plugin that implements the Qt image handler factory interface.



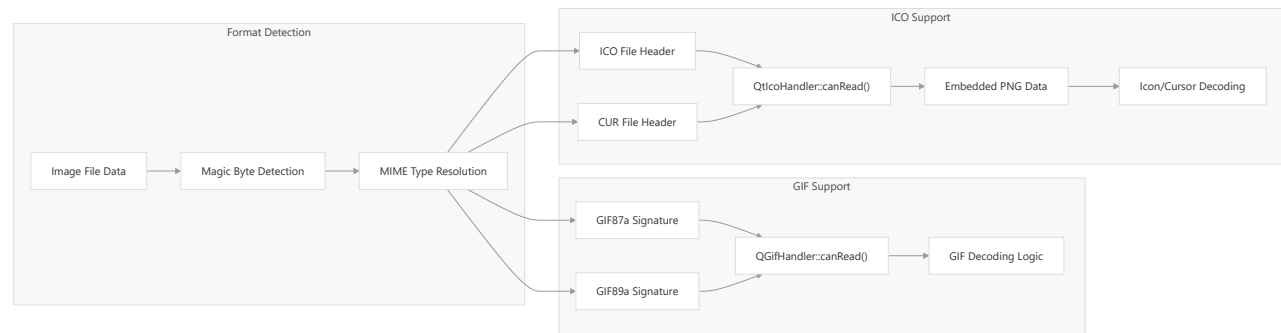
Sources: db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qgif.dll

db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qico.dll

Supported Image Formats

The image format support includes plugins for commonly used formats, each with specific capabilities and MIME type associations.

Plugin	Format	File Extensions	MIME Types	Handler Class
qgif.dll	GIF	.gif	image/gif	QGifHandler
qico.dll	Windows Icon	.ico , .cur	image/vnd.microsoft.icon	QtIcoHandler



Sources: db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qgif.dll

db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qico.dll

Plugin Implementation Details

GIF Plugin (qgif.dll)

The GIF plugin implements support for both GIF87a and GIF89a format specifications through the `QGifPlugin` and `QGifHandler` classes.

Key Features:

- Supports both static and animated GIF images
- Handles GIF87a and GIF89a format variants
- Implements transparency and color palette management
- Provides frame-by-frame animation control

Format Detection:

- Magic bytes: `GIF87a` and `GIF89a`

- MIME type: `image/gif`
- Error handling: `QGifHandler::canRead()` called with no device

ICO Plugin (`qico.dll`)

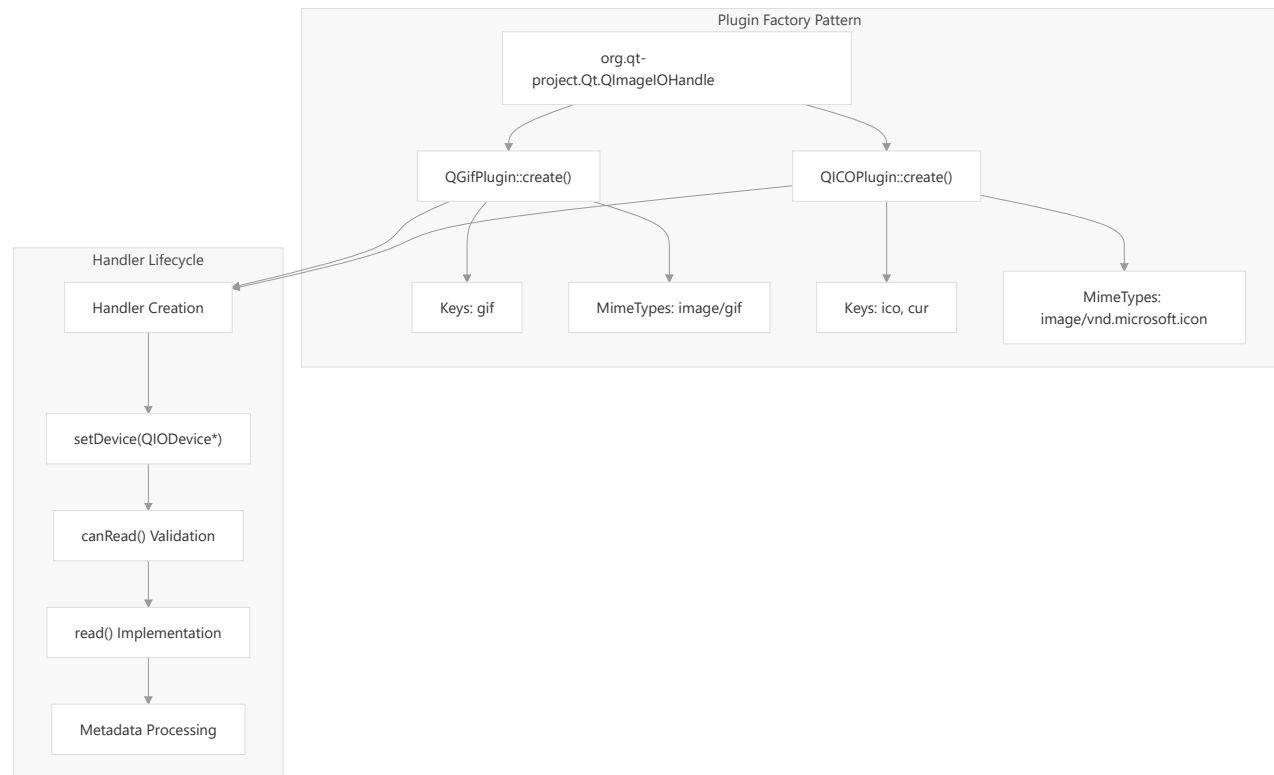
The ICO plugin provides support for Windows icon and cursor formats through the `QICOPlugin` and `QtIcoHandler` classes.

Key Features:

- Supports Windows ICO icon format
- Handles Windows CUR cursor format
- Manages multiple icon sizes within single files
- Supports embedded PNG data for modern icons
- Maintains original icon depth information via `_q_icoOrigDepth`

Format Detection:

- MIME type: `image/vnd.microsoft.icon`
- Supports both traditional bitmap and PNG-compressed icons
- Error handling: `QtIcoHandler::canRead()` called with no device



Sources: `db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qgif.dll`

`db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qico.dll`

Integration with DB Browser

The image format plugins are dynamically loaded by the Qt framework and integrated with DB Browser's image viewing capabilities. They enable the application to display image data stored in SQLite BLOB columns directly within the interface.


Plugin Location: `db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/`

Runtime Dependencies:

- Qt5Gui.dll - Core Qt GUI functionality
- Qt5Core.dll - Qt core system components
- KERNEL32.dll - Windows system API
- VCRUNTIME140.dll - Visual C++ runtime

The plugins are compiled with Qt 5.15.2.0 and signed with DigiCert certificates, ensuring authenticity and compatibility with the DB Browser application.

Sources:  `db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qgif.dll`

 `db/DB.Browser.for.SQLite-v3.13.1-win32/imageformats/qico.dll`