



Code less.
Create more.
Deploy everywhere.

Qt

NOKIA

**A technical look at the
framework that can take you
anywhere.**



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Cross-platform application framework

- Qt is a cross platform application framework.
- Part of the Linux Foundation's “Linux Standards Base”
- LGPL
- Free Software team player
- Long history, Development started 1991, first release 1995
- It includes:
 - Intuitive C++ class library
 - Integrated development tools
 - Support for desktop and embedded development



Binary Compatible

- Qt 4 is backwards compatible with older versions of Qt 4
- Your application will keep working even if the system Qt is updated
- How does it work? Strict coding practice & Magic “d” pointer



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Overview

- The Qt class library encompasses all the functions needed to build robust, high-end applications
- Separated into several modules, Qt's API includes
 - Core Classes
 - GUI Classes
 - SQL Database Classes
 - XML Classes
 - Networking Classes
 - OpenGL® Classes
 - Qt WebKit Integration
 - Multimedia classes
 - Testing Classes
 - And more



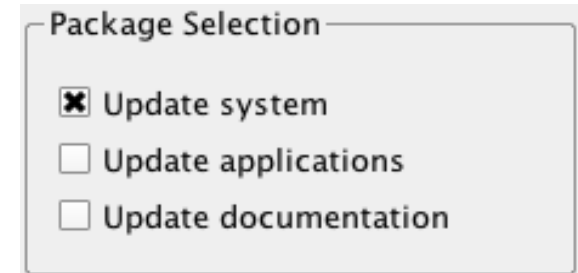
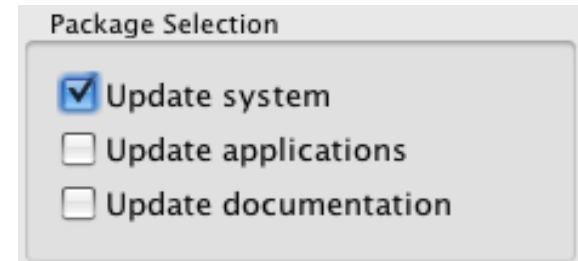
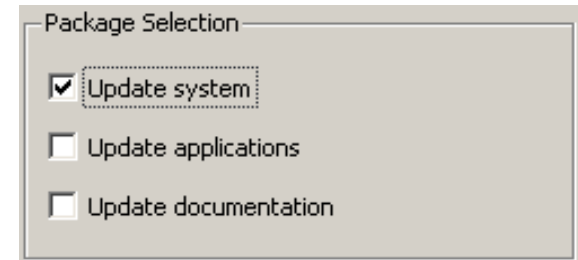
Qt Core Classes

- Qt core modules and utility classes
 - Qt kernel: Object model, event handling, application infrastructure
 - Basic input / output: File, IO, Streams
 - Threading
 - Signals and slots support, atomic reference counting, per-thread event loops, concurrency support
 - Container classes
 - For storing, sorting and retrieving groups of data
 - Robust, easy to use, low memory overhead, highly optimized
 - Alternative to STL (but Qt and STL work well together)
 - Fast XML stream reader and writer
 - IPC classes
 - shared memory, local domain sockets, memory mapped files



Qt GUI Classes

- Provides rich set of GUI components and supporting functionality
 - User interface components
 - Simple to complex widgets and controls, dialogs
 - Model-View-Controller (MVC) Itemviews
 - Styles – ensuring native or custom look and feel on target platform(s)
 - Font-aware layout system
 - Most major writing systems supported, even RTL.
 - Accessibility classes
 - Canvas component with Widget support





Qt GUI Classes

```
#include <QApplication>
```

```
#include <QPushButton>
```

```
int main(int argc, char* argv[])  
{  
    QApplication app(argc, argv);  
    QPushButton button("Foo");  
    button.show();  
    return app.exec();  
}
```



Concurrency Framework in Core Module

- Provides an abstraction on top of multithreading
- Write multithreaded programs without using basic synchronization primitives such as mutexes and wait conditions
- Reduces the programmer's need to deal with thread management
- Provides approximation of required threads for the available hardware
- Software scales without recompilation



Concurrency Framework in Core Module

```
QImage scale(const QImage &image)
{
    qDebug() << "Scaling image in thread" << QThread::currentThread();
    return image.scaled(QSize(100, 100), Qt::IgnoreAspectRatio, Qt::SmoothTransformation);
}
```

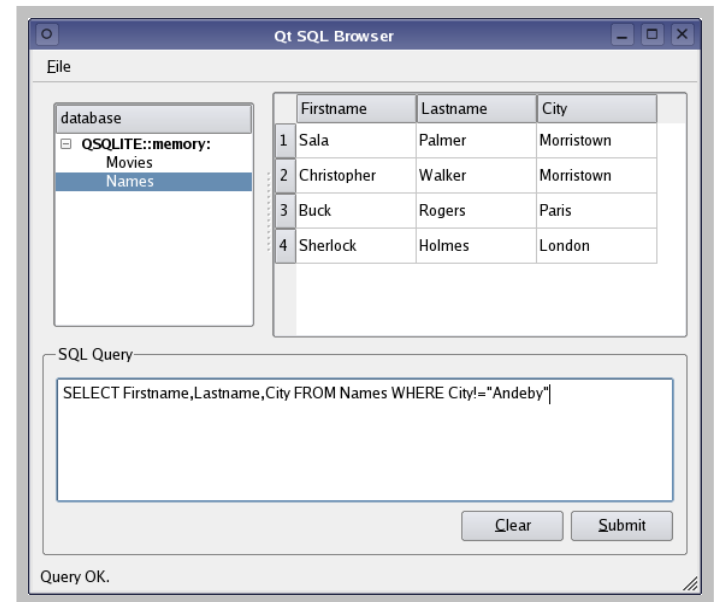
```
Void someFunction()
{
    // Create a list containing imageCount images.
    QList<QImage> images;
    for (int i = 0; i < imageCount; ++i)
        images.append(QImage(1600, 1200, QImage::Format_ARGB32_Premultiplied));

    // Use QtConcurrentBlocking::mapped to apply the scale function to all the
    // images in the list.
    QList<QImage> thumbnails = QtConcurrent::blockingMapped(images, scale);
}
```



Qt Database Classes

- Provide platform and database-independent access functionality
- Driver Layer
 - Low-level bridge between specific databases and the SQL API layer
- SQL API Layer
 - Provide access to databases
- User Interface Layer
 - Link data from a database to data-aware widgets
- Supports most major database drivers
 - DB2, IBASE, MySQL, OCI, ODBC, PSQL, SQLITE, TDS





Qt Networking Classes

- Provides TCP/IP networking functionality
- TCP sockets for clients and for servers
- Encapsulates TCP and UDP APIs
- SSL support
- HTTP 1.1 compliant asynchronous API
- FTP, DNS implementation
- Works with network proxies



Qt OpenGL Classes

- Allows you to build your user interface in Qt, display and manipulate 3D model in OpenGL®
- Integrates OpenGL canvas with Qt
- Provides frame buffer and pixel buffer abstraction
- Supports accelerating 2D painting with OpenGL
- Mix 2D painting and 3D scenes
- OpenGL graphics system draws everything using OpenGL



Qt XML Classes

- Core Module
 - Simple XML stream reader and writer
- XML Module
 - A well-formed XML parser using the SAX2 (Simple API for XML) interface
 - Implementation of the DOM Level 2 (Document Object Model)
- XmlPatterns module
 - An implementation of the XQuery standard
 - Enable users to query XML files similar to SQL
 - Semantics for value assignment, filtering, and simple operations
 - XSL-T support for advanced output formatting



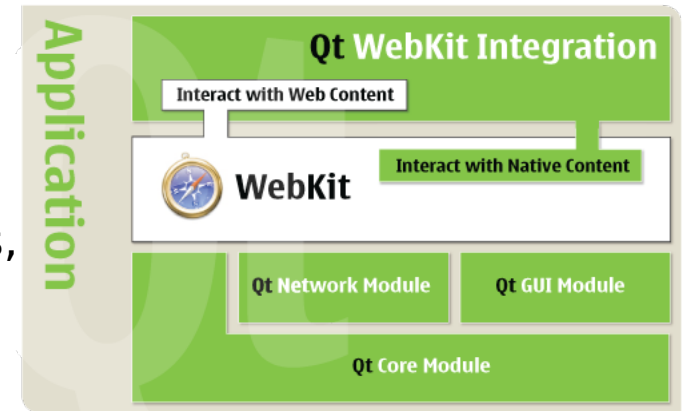
Qt XML Classes

```
QXmlStreamReader xml;  
xml.addData(<xml here>);  
  
while (!xml.atEnd()) {  
    xml.readNext();  
    if (xml.name() == "foo")  
        Qdebug() << "Foo's value is:" << xml.text();  
}  
if (xml.hasError()) {  
    // do error handling  
}
```




Qt WebKit Integration

- An open source HTML rendering component integrated with Qt
- Web standards compliant
 - support for HTML, XHTML, XML, stylesheets, JavaScript, HTML editing, HTML canvas, AJAX, XSLT, XPath, some SVG.
 - Acid3 Test compliance
- Deployable wherever Qt is: cross-platform/device
- Interact with Web environment, expose native objects
- NPAPI (Flash) support
- HTML 5 media element
- Various technologies from webkit.org
 - Full page zoom, css effects and animations, SquirrelFish JavaScript engine, client side storage





Qt WebKit Integration

```
QWebView *view = new QWebView(parent);  
view->load(QUrl("http://2009.encuentrolinux.com"));  
view->show();
```



Phonon Multimedia Framework

- Single, easy to use API for playback
- Provides possibility to play/synchronize multiple sound/video streams
- Will use native back-end format support
 - DirectShow on Windows
 - Gstreamer or Xine on X11
 - QuickTime on Mac
- Plan to add more video and authoring support in the future
- Also used as the backend for the HTML5 media element



QTestLib

- Allows for easy straight forward unit testing for your Qt applications.
 - Self contained
 - Data-driven
 - Simulate mouse/keyboard input
 - Thread safe
 - Extendable
- Benchmark measurements



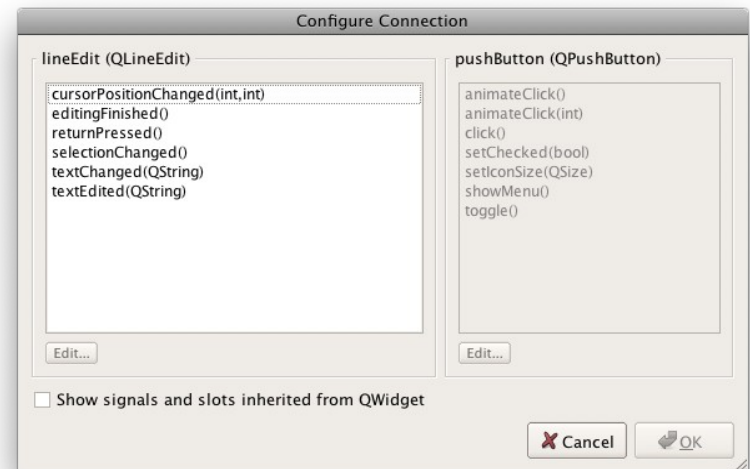
QObject / Metaobject

- Introspection
 - Enumerate classtype, properties and methods
- Q_OBJECT in the class definition along with QObject inheritance handles all the magic
- Signals and Slots



Signals and Slots

- Unique inter-object communication mechanism provides
 - Type-safe callback between objects
 - Facilitates loose coupling / encapsulation
 - Sender and receiver does not “know about” each other
 - 1-to-many, many-to-1 communication between objects
 - Works across threads
- Fully object-oriented





Signals and Slots

```
class AddressBook : public QWidget
{
    Q_OBJECT
public:
    AddressBook(QWidget *parent = 0);

private slots:
    void addName(QString);

signals:
    void needsSyncing();
};
```



Qt core tools

- qmake
 - Cross platform build system
- moc
 - Metaobject compiler
- uic
 - Ui Compiler
- rcc
 - Resource Compiler
- Irelease /lupdate
 - Translation tools
- More



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Development Tools: Overview

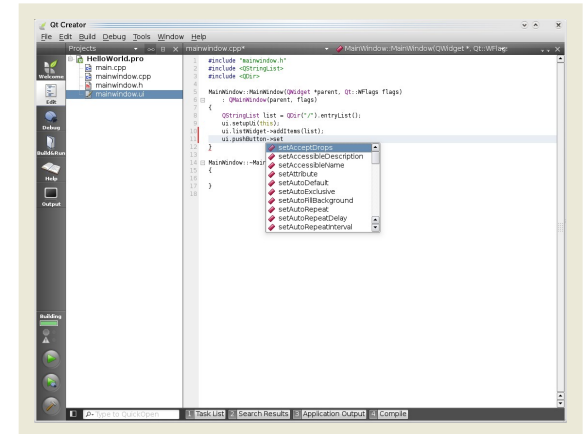
- Qt Creator: Cross-platform IDE
- Qt Designer: GUI layout and forms builder
- Qt Linguist: Internationalization toolset
- Qt Assistant: Customizable documentation reader
- qmake: Cross-platform build tool
- Integration with Visual Studio® and Eclipse® IDEs



Qt Creator: Cross Platform



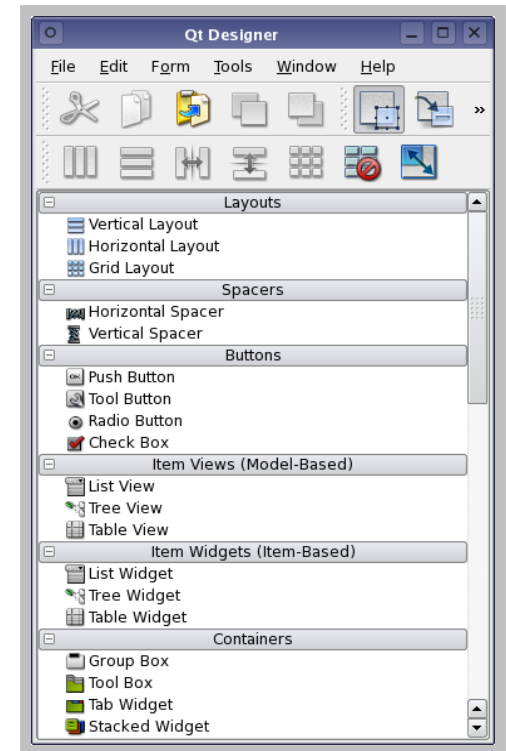
- Lightweight cross-platform IDE for Qt
 - An advanced C++ code editor
 - Integrated, context sensitive help system
 - Integrated visual debugger (based on gdb)
 - Integrated source code management
 - Project and build management tools
 - Qt developer tools integrated
- Benefits
 - New developers get up and running faster
 - Boosts developer productivity
 - Tailored to the needs of Qt developers





Qt Designer

- Qt Designer is a powerful, drag-and-drop GUI layout and forms builder
- Features
 - Supports forms and dialog creation with instant preview
 - Integrates with Qt layout system
 - Extensive collection of standard widgets
 - Support for custom widgets and dialogs
 - Seamless integration with Microsoft® Visual Studio .NET and Eclipse
 - Allows preview with embedded profiles
- Benefits
 - Greatly speeds the interface design process
 - Enables native look and feel across all supported platforms
 - Developers work within the environment of their choice, leveraging existing skills

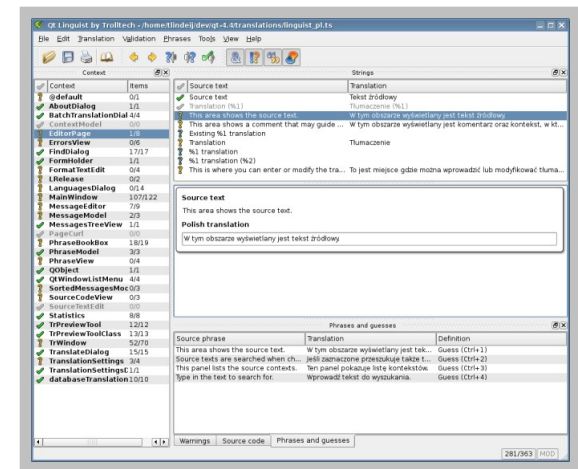




Qt Linguist



- Qt Linguist is a set of tools that smooth the internationalization workflow
- Features
 - Collects all UI text and presents it to a human translator in a simple window
 - Support for all languages, including CJK
 - Simultaneous support for multiple languages and writing systems from within a single application binary
 - Load and edit multiple translations at once
 - Support for .po and .xliff
- Benefits
 - Greatly speeds the translation/localization process
 - Works with Qt's language-aware layout engine for clean, consistent interface no matter the language
 - Easily target international markets





Qt Assistant



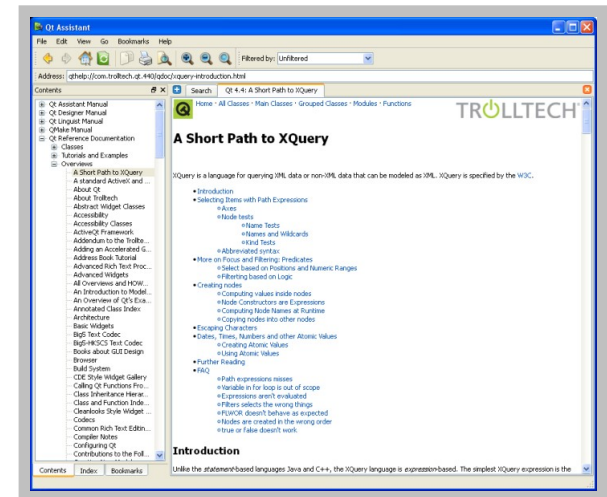
- Qt Assistant is a fully customizable, redistributable help file/documentation browser

- Features

- Simple, web-browser-like navigation, bookmarking and linking of documentation files
- Support for rich text and HTML
- Full text and keyword lookup
- Can be customized and shipped with Qt applications

- Benefits

- No longer have to build a help system from scratch
- Leverage existing HTML skills
- Deliver documentation in an easily searchable and navigable format to your end users





IDE Integrations

- (Integration for Microsoft Visual Studio 2003, 2005 and 2008)
- Integration for Eclipse
- Provides Qt API integration, Qt Designer and Qt Documentation
- Class Wizard
- Project Wizard
- Import and Export to Qt .pro files
- Qt version management



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Platform Specific Additions: X11

- Session Management Handling
- QMotif extension (Solutions)
 - Porting aid to facilitate gradual migration from Motif to Qt
- XEmbed component (Solutions)
 - Allows developers to easily run Unix/X11 programs inside their Qt applications



Platform Specific Additions: Windows

- Active Qt
 - Provides Qt applications as ActiveX objects
 - Integrating ActiveX objects into Qt applications
 - .NET interoperability
- MFC migration Framework (Qt Solutions)
 - Enables a gradual migration from MFC to Qt
 - Allows hosting Qt widgets in MFC, or MFC controls in Qt widgets
 - Functionality to merge MFC and Qt event loops
- WinForms interop Framework
 - Enables gradual migration from WinForms to Qt
 - Similar to MFC migration, allows hosting Qt in .NET WinForms and vice versa
 - Includes utils for value conversion



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Examples

Demo time!



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Upcoming features - 4.6

- Gestures & Multitouch
- Kinetic
- OpenVG
- Symbian
- Windows 7



Upcoming features - 4.7+

- Declarative UI
- Bauhaus
- Mobility
 - Contacts
 - Messaging
 - Location
 - Bearer



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Platform Specific Additions
5. Examples & Demos
6. Future Plans
7. Free Software



Free Software

- <http://qt.gitorious.org>
- Public roadmap
- Public bugtracker
- Anyone can clone and contribute to Qt
- Make a merge request and see your own feature in Qt-4.7!



Code less.
Create more.
Deploy everywhere.

Thank you! Questions?

<http://qt.nokia.com>

NOKIA