

RUSSOFT Software Development Forum

St. Petersburg

June 10th

2009

NOKIA
Connecting People

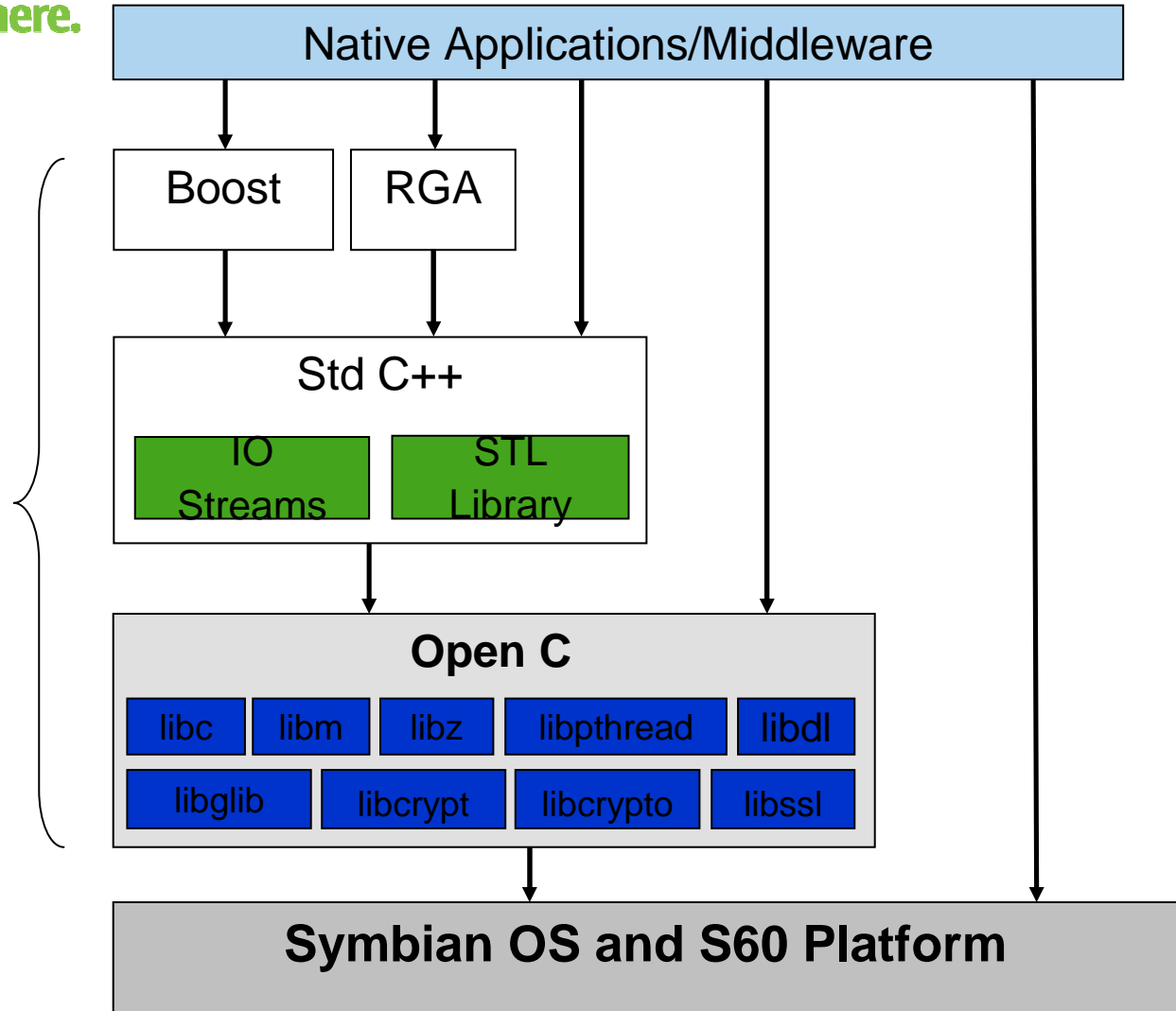
Qt

Forum **NOKIA**



Code less.
Create more.
Deploy everywhere.

Open
C++





**Code less.
Create more.
Deploy everywhere.**

Introducing Qt

**A technical look at the Qt cross-
platform application framework**



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Qt 4.5 highlights



Cross-platform application framework

- Qt is a cross platform application framework.
- It includes:
 - Intuitive C++ class library
 - Integrated development tools
 - Support for desktop and embedded development
- Current version is Qt 4.5 (published in March 2009)



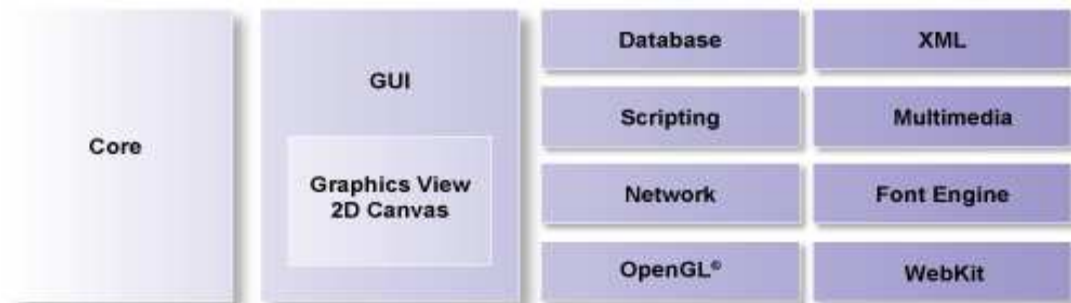
Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Qt 4.5 highlights



Overview

- The Qt class library encompasses all the functions needed to build robust, high-end applications
- Separated into several modules, Qt's intuitive API includes
 - Core Classes
 - GUI Classes
 - SQL Database Classes
 - XML Classes
 - Networking Classes
 - OpenGL® Classes
 - Qt WebKit Integration
 - Multimedia 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
 - Supporting tools: qmake, rcc, configure, moc



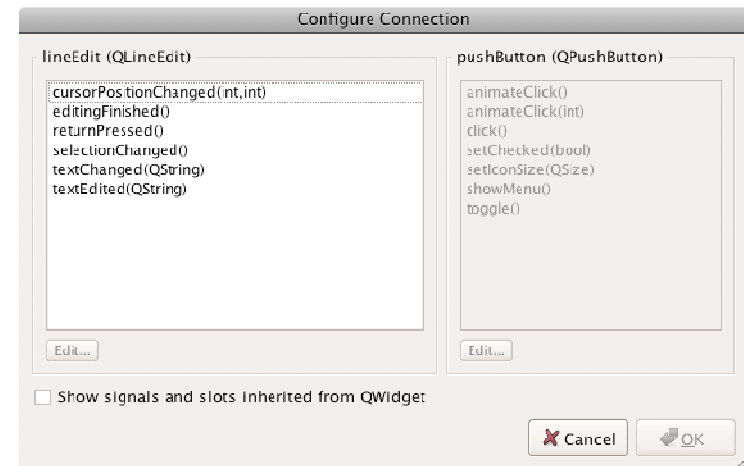
Concurrency Framework in Core Module

- 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



Signals and Slots in Core Module

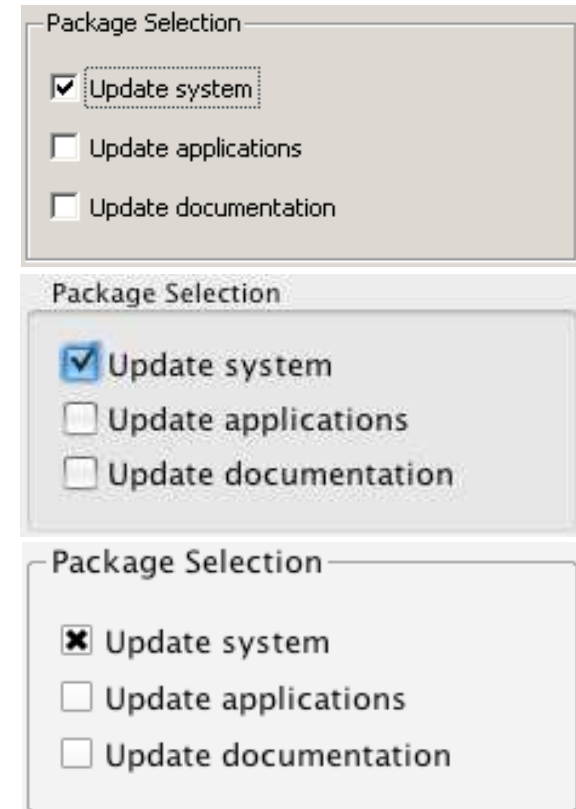
- Unique inter-object communication mechanism provides
 - Type-safe callback between objects
 - Facilitates loose coupling / encapsulation
 - Sender and receiver does not “know about” each other
 - 1-to-many, many-to-1 communication between objects
 - Works across threads
- Fully object-oriented





Qt GUI Classes

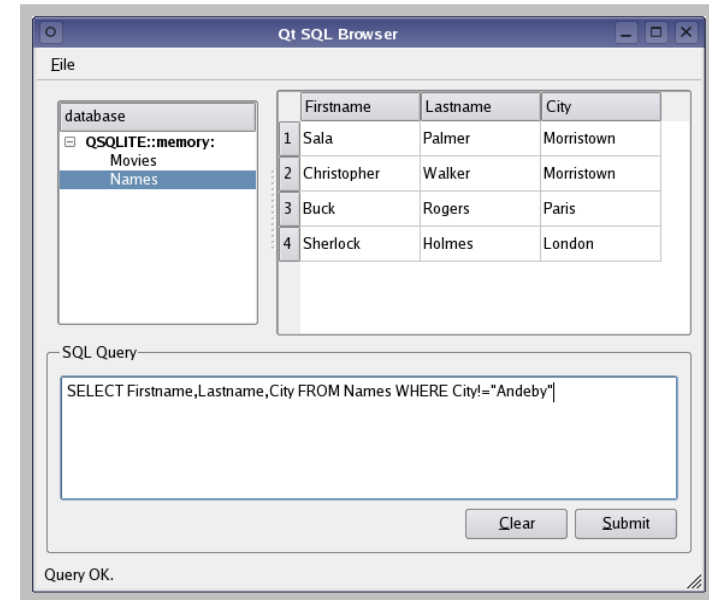
- Provides rich set of GUI components and supporting functionality
 - User interface components
 - Simple to complex widgets and controls, dialogs
 - Model-View-Controller (MVC) Itemviews
 - Styles – ensuring native or custom look and feel on target platform(s)
 - Font-aware layout system
 - Accessibility classes
 - Canvas component with Widget support
- [UI candy demonstration](#)





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



```
QSqlDatabase db =  
    QSqlDatabase::database("sales");  
QSqlQuery query  
    ("SELECT Lastname, City FROM Names", db);
```



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

```
QHttp http;  
connect(http, SIGNAL(done(bool)), this, SLOT(done(bool)));  
http->get("http://www.softafoorumi.com/amazingApi.jsp?action=doGreatThings");
```

```
void MyClass::done(bool error)  
{  
    QString data = http->readAll();  
    //handle data  
}
```



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

```
QXmlStreamReader xml;  
xml.addData( data );  
while (!xml.atEnd())  
{  
    xml.readNext(); ...  
    // do processing  
}
```

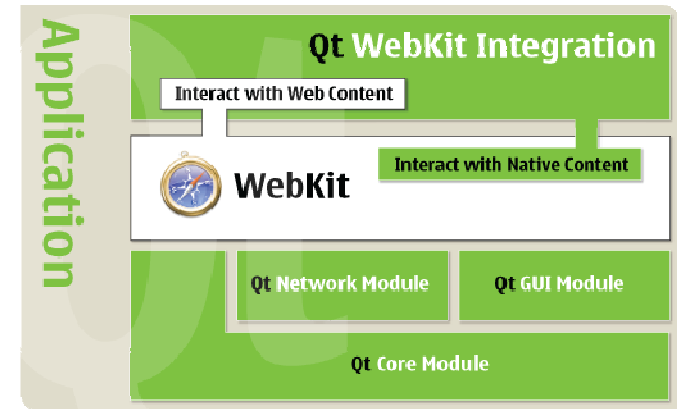


Qt WebKit Integration

- 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/cross-version/cross-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

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

<http://webkit.org>





Phonon Multimedia Framework

- Single, easy to use API ("Phonon") for playback
- Provides possibility to play/synchronize multiple sound/video streams
- Will use native back-end format support
 - DirectShow on Windows
 - GStreamer 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

```
Phonon::VideoPlayer *player =  
    new Phonon::VideoPlayer(Phonon::VideoCategory, parentWidget);  
player->play(url);
```



Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Qt 4.5 highlights



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

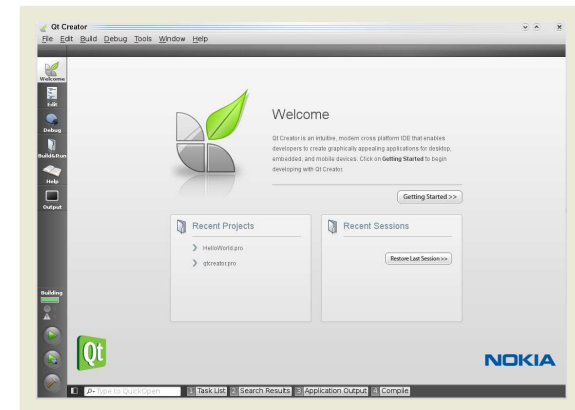
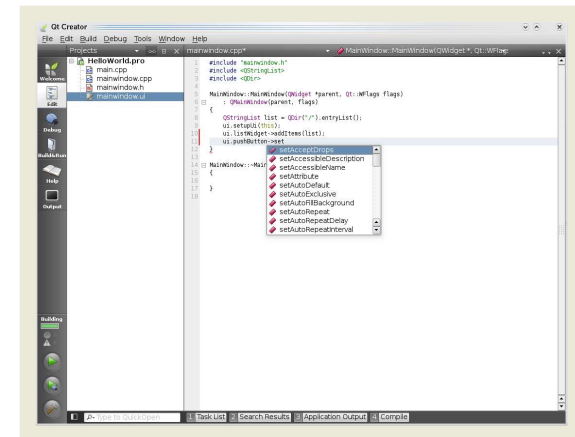
- For S60 development Carbide.c++ is recommended



Qt Creator: Cross Platform IDE



- 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





Contents

1. Introducing Qt
2. The Qt Library
3. Qt Development Tools
4. Qt 4.5 highlights



Qt 4.5: Design Goals

1. Improve the runtime performance of Qt-based applications
2. Allow Qt users to realize the full potential of WebKit: The best browser engine on the market
3. Futureproof users' investments on the Mac platform with 64-bit support



Realizing Full Potential of WebKit (1/2)

- Upgrade to latest upstream (trunk) version of WebKit
 - Near 100% compliance with Acid3 testsuite
 - Full page zooming: images and fonts
 - CSS-based transformation and animation (WebKit extension)
 - Standards compatibility improvements
- Netscape plugin API support
 - Load any Netscape plugin (i.e. Flash, Flash Lite) in-process
- HTML 5 support
 - Audio and video elements using Phonon
 - Client-side storage of web content
- SquirrelFish JavaScript engine
 - New engine for execution of JavaScript code in WebKit





Realizing Full Potential of WebKit (2/2)

- Benefits to developers:
 - Use existing installed plug-ins to show rich content (i.e. Flash)
 - Better performance for exposed code and scripts
 - Provide off-line access to web data
 - Integrate media into web-aware applications with less code



More Information on Qt WebKit Integration

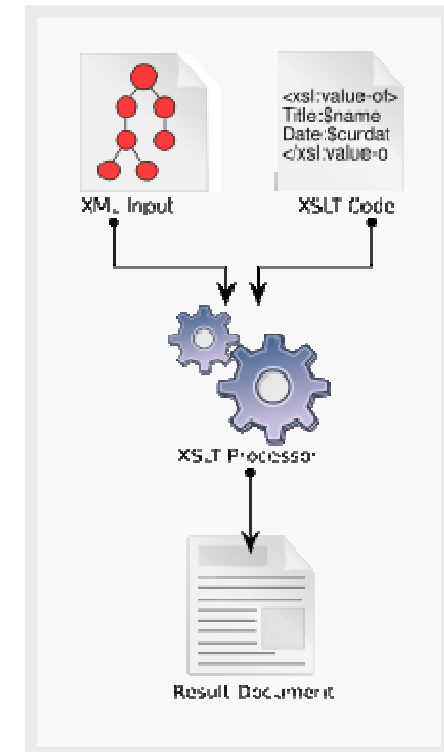
Whitepaper: *Qt Features for Hybrid Web/Native Application Development*

Download from qtsoftware.com/whatsnew



XSL-T Support

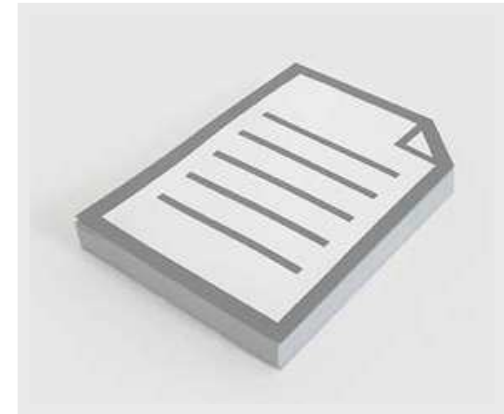
- Support for transformation of XML content into XML, HTML or other text
- Benefits to developers:
 - Simple to reformat XML content without changing data structure
 - Removes need for intermediary DOM presentation layer
 - Faster time to market with XML data
 - e.g. reports as HTML or PDF





Additional Features

- OpenDocument Format support
 - Interoperate with OpenOffice.org and KOffice (MS Office support in future releases)
 - Currently only writing support
- Improved proxy support
 - Integration with system proxy settings and ability to handle non-trivial proxy cases
- Qt Designer improvements
 - Support for changing item types
 - Support for different screen resolutions
 - Usability improvements
- ARGB top level windows
- Improved GTK integration





Code less.
Create more.
Deploy everywhere.

Questions?