Qt Contributions
in many shapes and forms

FRUCT 10
by Knut Yrvin – 09.11.2011
Knut Yrvin
Open Source Community Manager

• Telenor in 1986 – technical assistant
• Engineer in electronics – 1992
• Telenor – project planning – 1993
• Master in system development – 2000
• Consultant for three years
  • Norwegian Central Securities Depository, different banks and other industries
• Co-founded Skolelinux in 2001
  • Now a part of Debian Edu
• Trolltech ASA – 2006-2008
• Qt at Nokia – 2008 - present
1  About Qt
2  Why Qt contributions?
3  Experiences with Qt contributions
Qt history, size and scale

- Trolltech founded in 1994
- Trolltech acquired by Nokia in 2008
- Used by more than 500k commercial and open source developers
- Flexible licensing gives developers options – Commercial, LGPL v2.1, GPL v3
Qt overview

Qt: a tool for software developers to make the user experience better on different types of platforms

Qt is a powerful and rich application framework for mobile, desktop and embedded platforms.

Using Qt, developers can reuse significant amounts of code when deploying their applications across desktop, mobile and embedded operating systems. For some use cases, Qt allows full cross platform deployment with without any changes to source code.

“Code less. Create more. Deploy everywhere”
Qt and commercial customers

From embedded devices to mobile and desktop applications

Used by leaders from many industries

© Nokia 2011
Qt and open source projects

VLC media player

© Nokia 2011
Accelerated Qt growth

The Qt SDK - downloads
- 2008: < 400,000
- 2010: > 1,500,000

Qt Developer Days
- 2008: 500 in Munich, 300 in San Fran
- 2010: 1000+ in Munich, 650 in San Fran

Estimate < 2,000,000
Responding to market changes

Accelerated hardware everywhere

From mouse/keypad to direct touch manipulation

More iterative co-operation between designers and developers

Web applications: solving deployment problems and increasing OS fragmentation
Qt roadmap in brief

Qt 4.7
- Introducing QML and Qt Declarative
- Established a new benchmark-driven development culture
- Increased focus on quality
- More complete tools for mobile development

Qt 4.8
- Lighthouse platform abstraction - port Qt to new platforms in a simpler and quicker way
- Qt Quick 1.1 gives an evolution to the Qt Quick revolution
- Qt WebKit updates

Qt 5 - the future
- Smooth graphics performance on limited hardware
- Simple, powerful UI creation with QML and JavaScript
- Make application web integration as powerful as possible
- Greater modularity in Qt framework
- Continue to reduce complexity to implement and maintain a port
- Open Governance

© Nokia 2011
Using Qt makes developers happy :)

Qt simply rocks. #qtdd10
4:12 PM Oct 12th via web

QML is soo sexy and lovely #qt #qtdd10
10:21 AM Oct 14th via Twitter for iPhone

Compiling the @qtbynokia port of @WebKit. Having fun!
18 Jun via Ubuntu Favorite Retweet Reply

wow #qt SDK Maintenance Tool is awesome!
21 hours via Choqok Favorite Retweet Reply

RT @jmons: So it looks like we'll be programming in QT after this week, Really impressed by the power - #NokiaWorld
14 Sep via web Favorite Retweet Reply

@qtbynokia Qt Creator just went from being awesome, to absolutely amazing. I feel stupid for not upgrading earlier."Qt user" #Qt
1 Jun via web Favorite Retweet Reply

Will be going back to doing Qt development soon. @qtbynokia produce awesome APIs.
3 hours ago via web
Graphical User Interfaces with widgets

- User interfaces are constructed using Qt widgets
- Look and feel adapt to the current platform
- Unicode and internationalization
- Arranging in Qt’s layout classes gives sane sizes and placements across all platforms
The **QWizard** and **QWizardPage** classes lets you create flexible wizards

- Can be linear or non-linear
- Adapts to the look of the used platform
Qt provides classes for animations

QPropertyAnimation lets you animate any property of any object

QEasingCurve lets you decide the shape of the movement

There are also a set of classes for graphics effects

QGraphicsBlurEffect
QGraphicsDropShadowEffect
QGraphicsColorizeEffect
QGraphicsOpacityEffect
Graphical User Interfaces with widgets

QWidget

Examples of international text
e.g. Arabic and Chinese on buttons
Desktop Applications

QMdiArea

QMainWindow

QDialog

QMainWin
```cpp
QString fileName = QFileDialog::getOpenFileName();

QColor color = QColorDialog::getColor(
    QColor(64,171,89), 0, "Select color");
```
Desktop Applications

Interactive Help

Tips on:
- Hover (tool tips)
- Status bar
- What’s this
Web Integration

The **QtWebKit** module makes it easy to include web content in your Qt application:

- HTML rendering
- JavaScript
- Supports NPAPI
Qt provides two sets of classes for handling video and sound

**Phonon** provides a source, sink and path based API
- Developed in cooperation with the KDE project
- GStreamer on Linux
- DirectX on Windows
- QuickTime on Mac

**QtMultimedia** provides low level access to video and sound devices
QtCreator

A cross platform integrated dev environment

Optimized for Qt
Integrated documentation

There are alternatives
Visual Studio
X Code
KDevelop
QDevelop
Edyuk
emacs
etc
Using the `QObject::tr` function and the Linguist tool, translation is easy.

Encapsulate all user visible strings in calls to `tr()`, e.g. `tr("Hello World!")`

Extract strings using `lupdate`

Translate in Linguist

Create translation file using `lrelease` and load as `QTranslator`
Styling

Qt uses styles (**QStyle**) to mimic the look of the current platform.
The Windows styles use the native infrastructure.
The OS X style uses the native infrastructure.
CDE, Cleanlooks, GTK, Motif, Plastique are implemented by Qt – i.e. movable across platform boundaries.
Qt Quick: Qt User Interface Creation Kit

QML

Qt Declarative
Challenge between design and implementation

The UI design gets lost between idea and realization
Qt Quick

Fixes the UI design workflow
Developers and designers work together!

QML

Qt C++

Qt Quick Designer
What's so Quick about it?

Quick to Learn
Quick to Iterate
Quick to Render
Quick to Extend

© Nokia 2011
Qt Project - Open Governance
Open Governance

To empower developers using Qt, moving towards a system of Open Governance

Open Governance = Nokia **Qt developers** and **external contributors** being equal

**Fair**: Treat everyone contributing to Qt – the Qt dev team and external contributors - under the same rules

**Transparent**: Decisions about Qt made in the open

**Meritocratic**: Rights and duties assigned according to merit
Why Open Governance?

• Empowering the Qt community of close to 500,000 developers
• Qt has received tens of thousands of bug fixes and thousands of code contributions through the last years.
• We expect the governance model to simplify and speed up contributions. More contributions mean more value to the project and the contributors themselves.
• Increase adoption of Qt and increase consumption of Qt apps with support for more platforms.

It will be easier to participate in the evolution of Qt.
Qt Open Governance 'The virtuous cycle'

- Increased use of Qt
- Faster pace of innovation = a better Qt
- Larger Qt ecosystem
- More feedback & contributions
What

More open Governance

- Next logical step in Qt’s progressive opening up
- Model chosen: an Open Source project

Applies to all Qt Projects

- Qt, Qt Tools, Qt Creator, Qt Mobility, Qt WebKit, etc.

Key Concepts

- Fair,
- Transparent
- Inclusive
- Meritocratic
Making all tools available for all

T = Qt Nokia internal developer contributor
C = external contributor
CI = Continuous Integration System
Governance Modell

General:
• Inspired on WebKit and kernel models
• Anyone can be contributor
• Easy to become approver
• Maintainer only if needed

Open Governance Bodies
• Open Governance summit
• Mailing lists
• Blogs
• Direct mails
• Wikis
• Qt Project Internet pages
Roles

Role mapping:
Chief Maintainer = Lars Knoll
Maintainers: ~50 appointed 11 will be appointed.
Latest update: http://wiki.qt-project.org/Maintainers
## Who has contributed

<table>
<thead>
<tr>
<th>Org.</th>
<th>Open Requests</th>
<th>Merged Requests</th>
<th>Rejected Requests</th>
<th>Closed Requests</th>
<th>Total Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nokia</td>
<td>19</td>
<td>41</td>
<td>14</td>
<td>0</td>
<td>74</td>
</tr>
<tr>
<td>KDE</td>
<td>6</td>
<td>50</td>
<td>11</td>
<td>0</td>
<td>67</td>
</tr>
<tr>
<td>Digia</td>
<td>15</td>
<td>14</td>
<td>5</td>
<td>0</td>
<td>34</td>
</tr>
<tr>
<td>KDAB</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>OpenBossa</td>
<td>1</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Basyskom</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Tieto</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Intel</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Tandberg</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Accenture</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Fedora Project</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>RedHat</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>(gmail)</td>
<td>48</td>
<td>170</td>
<td>63</td>
<td>1</td>
<td>282</td>
</tr>
<tr>
<td>(Hotmail)</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>(Sourceforge)</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>(Yahoo)</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>7</td>
</tr>
</tbody>
</table>
Examples of community contributions from KDE

- Xquery/XPath support currently in Qt started as a KDE hosted project
- Phonon started as a KDE project, and continues to be maintained and developed primarily by KDE community members
- KDE are early adopters of new Qt technologies, providing actionable feedback
- Key feedback and early adoption on QGraphicsView and QML
- 500+ reports currently in JIRA
- Ensures products release in much better form
Community ports take Qt even further*

* Qt does not discourage community ports, however these ports, including the community Qt for Android port are not officially supported, or official Qt ports.

© Nokia 2011
You might contribute

• **Direct** – spending time and/or money to make direct code contributions, unit tests, bug-reports, documentation and translations

• **Indirect** – by making applications (code, translate ++) which occasionally reveals Qt fixes and/or need for improvements, new features etc.
"We fully support the work being done with the Qt Project. An openly governed Qt is in the best interests of all Qt developers. The Open Governance structure of the Qt Project empowers developers to influence the direction and the pace of Qt development. Stakeholders in the future of Qt, such as KDE, can now contribute according to their own priorities and take ownership over areas of Qt that are of particular importance to them."

Olaf Schmidt-Wischhöfer, KDE Free Qt Foundation
Martin Konold, Co-Founder K Desktop Environment
"The Qt Project is a welcome development in the evolution of Qt. This move should allow for participation in the project from a broad set of developers based on the quality of their code. This is certainly a welcome change."

Jim Zemlin, Executive Director, The Linux Foundation
Qt Everywhere
Desktop example - KDE (Linux)
15 years of building better software with Qt

> 5000 applications
Qt is default on Symbian

100 Million + 150 Million = 250 Million

Installed  To be sold  Symbian
MeeGo is more than just mobile

• In-Vehicle Infotainment
• Set-top boxes
• Tablets and much more

Qt is the way to get your app on this free, Linux-based OS (and on the beautiful Nokia N9!)
Community ports take Qt even further*

* Qt does not discourage community ports, however these ports, including the community Qt for Android port are not officially supported, or official Qt ports.

© Nokia 2011
Summary

- Qt allows code reuse across platforms. Code once, deploy everywhere.
- Create advanced applications. Concentrate on innovation, not basic coding!
- Broad platform support across desktop mobile and embedded opens up hundreds of millions of devices that can be targeted.
- Community support is a strong part of Qt, and is getting stronger. Everyone contributes by either funding the ongoing development, or contributing code via GPL and LGPL.
- Choice of license – Commercial (via Digia), LGPL or GPL.
- Qt Quick makes rich, quick mobile UIs and applications quicker and easier.
Please visit:

http://qt-project.org/

Thank you!