Leveraging Qt in the MeeGo Ecosystem

Zhang Chi
Nokia, Qt Development Frameworks, Greater China
What is the MeeGo project?

Building a modern OS with no strings attached for multiple device segments, merging the best of Moblin and Maemo.

Consolidating the power, resources, and innovation of the leaders in communication and computing.

Facilitating operating system convergence.
One MeeGo stack for Multiple Segments, Multiple Application Stores, & Multiple HW Architectures

Easy integration to multiple Application Stores

MeeGo OS

Multiple hardware architectures: Intel Atom, ARM, others
MeeGo Linux Stack is Built on Many Open Source Projects
MeeGo consumer promise: Personalized Internet on the Go

Delivering the best web experience
- Delivering the leading browser experience
- True Adobe Flash integration

Always online on the Go
- Always-connected software design delivering a prompt-free experience
- Power consumption and network optimization across the whole software stack

Personalization with Native Applications
- 1,000s of Qt apps leveraging the installed base of MeeGo devices in multiple categories
- Delivered with a renewed experience in Intel AppUp Centers and Ovi Store
Our developer promise: One API – multiple device categories

- One MeeGo API based on Qt
- Fastest growing installed base
- Lighting fast Qt Creator IDE tool
- Global reach of distribution channels with Dedicated go-to-market programs
- Proven Qt track record in CE devices
MeeGo built with Qt expands Developer opportunities

One code base can address multiple consumer screens
MeeGo application ecosystem

- 1000s of Qt cross-platform apps
- 1000s of web runtime apps
- 100s of native Linux apps
- Operator service apps

Millions of MeeGo devices from dozens of companies
Qt Application Framework

Cross platform application development

Qt modular class library
- Core
- GUI
- WebKit
- Graphics View
- Scripting
- OpenGL
- XML
- Multimedia
- Database
- Network
- Unit Tests
- Benchmarking

Qt development tools
- Qt Creator
- Qt Designer
- Qt Linguist
- qmake
- Cross-Platform IDE
- Build Tool
- Help reader

Cross-platform support
- Windows
- Mac
- Linux/X11
- Embedded Linux
- Maemo/MeeGo
- Symbian
- Windows Mobile
- WinCE

Desktop, web, mobile and embedded software development

Fast, efficient programs

Allows developers to focus on **value added functionality** instead of infrastructure code

Used by more than 350,000 developers around the globe

*Write code once to target multiple platforms & OSes*
Market Needs / Roadmap Drivers

**Up to Qt 4.5**
- Cross-platform requirement
- Improve developer efficiency

**Qt 4.5**
- Better out-of-the-box experience for developers
- Best possible runtime performance

**Qt 4.6**
- Expanded support for embedded, smartphone development
- Enablers for cutting edge UI development

**Future direction**
- Developer & designer collaboration, declarative UI
- Deeper support for web/native hybrid development

2008 2009 2010
Qt is hot in multiple screens today

**Slate/Pad/E-Reader**
- Windows XP/7 and/or Linux-based OS
- Create Qt-based SDKs and Applications/Services
- Multi-touch and Gesture

Key Success: Asus EEEpc, Skype Phone, E-Reader, EEE tablet

**Home & Entertainment**
- Chipset pre-integration
- Online Content and services delivery
- Performance and WebKit Integration

Key Success: TCL MiTV, Netflix, CNTV

**Mobile Applications**
- Major improvements in overall performance
- Next generation mobile UI
- Nokia mobile platforms

Key Success: AccuWeather, Shazam, many on

**Automotive Infotainment**
- Connected & innovative user experience
- Terminal Model
- Qt for Windows CE

Key Success: Genivi, Visteon, MagnetiMarelli
Qt Openness

First Step
Enter into new ERA - Qt is Now Open

Rationale for Nokia to add LGPL
• Make it easier for commercial users to choose Qt
• Increase Qt Ecosystem
  – Increase available Qt developer competence
  – Provide more business opportunities based on Qt
• Improve Qt through valuable feedback and community contributions

Increased use of Qt
Faster pace of innovation = a better Qt
Larger Qt ecosystem
More feedback & contributions
Open Source
Commercial

© 2010 Nokia
Open Qt Development Model

True Open Source project working model

- Development - Source Code management system
- Bug Track system
- Roadmap
Source Code Management System

• Git
  – A distributed revision control system
  – Used by Qt, Linux, Android, Gnome, Perl...
• Gitorious
  – A web based project host for collaborative open source projects using the Git distributed revision control system
  – Used by Qt, OpenSuse, MeeGo...
• Bring many interesting projects
Bug Track System

You can
– Find a bug
– Report a bug
– Keep track of reported bug
– Vote a bug
• Both external people and Qt developers are working at the same system!
Roadmap

We open our product direction for discussion:

- Public Qt Roadmap
- Qt Labs
- Blogs
- Validation Meetings
Qt Contributions & Workflow

Developers can submit patches and add-ons, or help jointly develop new features in Qt.
What can I contribute?

• You can contribute to Qt and Qt tools
• Suggestion list for discussion:
  – Performance improvements
  – Benchmark improvements
  – Optimizations for platforms / devices
  – Power-consumption optimizations
  – Localization!
Qt Governance

Evolving towards Transparency
What is Open Governance?

**Planning in public**
Technical & content discussions in public on mailing lists, IRC & wikis
Conclusions from off-line meetings are made available to the public

**Development in public**
Internal and external development happens the same way – in public
Degree of influence derives from level of contribution

**Merit-based authority**
Authority to decide is given based on merits earned in the community, not by your employer

**Product info remains secret**
Information about un-released products private, all other is in the open
Early June 2010
Make public intent announcement.
Open mailing lists and wiki for public discussion about the Open governance model.

2009
Open Qt to external contributions.

Decide the open governance model – target state defined for change project

End of 2010
Change project executed.
Move to the new governance.

Qt Roadmap to Open Governance
Benefits?

• Drive Qt for projects needs
• Have a hand in the control of Qt’s evolution
• Freedom to innovate
• Fast path from contribution to productization (not relay on us)
MeeGo working model

Leveraging industry players
Way of Working:
Applying proven open source processes
MeeGo Organization Structure

The Linux Foundation

MeeGo Technical Steering Group

Program Office

- Core Program
- Handset Program
- Netbook Program
- IVI Program
- Tablet Program
- Connected TV Program

- Technology Development
- Distribution Development
- UI Design
- Release Engineering
- Quality Assurance
- Support (IT, Legal, Documentation, Localization)

Community Office

- Handset WG
- Netbook WG
- IVI WG
- Tablet WG
- Connected TV WG

© 2010 Nokia
Working Groups

• Working Groups are Delegates of the TSG

• WG responsibility
  – Defines the requirements
  – Release Management for the vertical UX
  – SW component licensing within overall MeeGo policy
  – Appoint leaders and maintainers for WG projects
  – Conflict resolution for WG projects

• Defines the vertical UX compliance profile specification

• Liaison with institutions within the vertical industry
Technical Steering Group

• Membership and guidelines
  – Initial membership is Intel and Nokia, serving as co-chairs
  – Membership expands based on meritocracy, contributions, and commitment to MeeGo; expect to add 3 to 8 members to the TSG over the next year
  – All meetings of the TSG are open, and all TSG decisions and meeting minutes are published on the MeeGo project website

• TSG responsibilities
  – Appoint Project leaders, maintainers, and WG members
  – SW Compliance, Licensing and Trademark use
  – MeeGo Platform Advocacy and Outreach
  – Conflict resolution for projects
  – Represents the MeeGo project
## Summary

<table>
<thead>
<tr>
<th>Qt</th>
<th>MeeGo</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Cutting Edge graphic and web-enabled SDK</td>
<td>➢ Complete open OS stack for different device type</td>
</tr>
<tr>
<td>• Cross-platforms applications development across desktop, web, mobile and embedded platforms and different screen size</td>
<td>➢ large pool of applications and services, through Intel AppUp and Nokia OVI Store</td>
</tr>
<tr>
<td>• Rich ecosystem of developers and applications/services already existing</td>
<td>➢ More innovation, smaller overhead, quicker time-to-market</td>
</tr>
</tbody>
</table>

### Both running under Open-Source model

**Benefit each other**
MeeGo Today

Netbook + Handset + IVI
First complete UX Profile
MeeGo v1.0 Netbook

Visually rich user experience

Instant access to calendar, tasks, appointments, and social networks through the home screen.
MeeGo Handset UX Day 1

Running on both Moorestone and N900

Enhanced Qt framework - MeeGo touch UI framework
Genivi select MeeGo

- GENIVI will be an active member of the MeeGo IVI working group.
- Because MeeGo is the best distribution for user centric devices where speed, size, power consumption and usability are key factors.
- Chosen by BMW, GM, Hyundai
<table>
<thead>
<tr>
<th>Date</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30 March 2010</td>
<td>Code open of MeeGo 1.0 OS</td>
</tr>
<tr>
<td>26 May 2010</td>
<td>MeeGo 1.0 Netbook UX complete</td>
</tr>
<tr>
<td>June 30 2010</td>
<td>Code open of MeeGo 1.1 Tablet and Handset UX</td>
</tr>
<tr>
<td>Q4</td>
<td>MeeGo 1.1 Core OS complete</td>
</tr>
<tr>
<td></td>
<td>MeeGo 1.1 Handset UX, Tablet UX, Netbook UX, In-Car-Infotainment UX complete</td>
</tr>
</tbody>
</table>
Join the MeeGo project