

Using Qt for GUI Tasks and Graphics in R

Deepayan Sarkar, Michael Lawrence

Computational Biology
Fred Hutchinson Cancer Research Center

July 10, 2009

R and Qt

- *R*: primarily command-line (Read-eval-print loop)
 - Allows GUI front-ends to embed *R*
 - Facilitates bindings to external libraries; e.g.
 - libxml, *tcltk*, *RGtk2*
- Qt: powerful C++ Application and UI framework
 - Mature, cross-platform, high-performance
 - Latest release (Qt 4.5) is GPL/LGPL on all platforms
 - Good candidate for binding to R

Focus areas

- Improve access to *R* documentation
 - Difficult to search (Java search engine?)
 - Inaccessible when R is running
 - Multiple matches not handled gracefully
 - S4 method documentation hard to find
- Enable Qt GUI programming
 - Beneficial for many problems
e.g., domain-specific GUI, graphical debugger
 - Should be accessible to R programmers
- Graphics (both static and dynamic)
 - R graphics powerful for static graphics, but slow (grid!)
 - Not designed for dynamic manipulation

R and Qt

- *RKward*: Qt/KDE based front-end embedding *R*
- *RQt*
 - Proof-of-concept Omegahat project (Duncan Temple Lang)
 - Long-term focus on automatic generation of bindings
- *qtinterfaces*
 - Recently registered R-forge project
 - Initial developers:
Deepayan Sarkar, Michael Lawrence, Hadley Wickham
 - Goal:
Create a coherent collection of R packages that provide an interface to the Qt application and UI framework, with a focus on enabling GUI development and advanced graphics.

Some Demos

- Using Qt Assistant to view R documentation
- “Object viewer” with applications
- An *R* graphics device implemented using Qt’s Graphics/View framework
- Mosaiq: a high-level graphics system not unlike lattice, but completely independent of the R graphics engine.
- RGtk2 driving a dynamic Qt view

Qt Assistant

- Standalone Qt library documentation viewer
- Can also be used as viewer for third-party static HTML files
 - Requires preprocessing (add keywords, register, compile)
 - Once processed, much more powerful help interface than anything R has
- *But*, R documentation is dynamic!
- Incompatible models, but we can get snapshots, almost as useful

Qt GUI programming

- Creation and memory management of widgets and other Qt objects (as external pointers)
- Manipulate these objects using R functions
- Connect Qt signals to arbitrary R functions
- Examples:
 - “Object viewer” with applications
 - Data import wizard

Graphics

- Based on the Scene Graph model; supports layers and layouts
- Fast implementation of low-level drawing
 - Custom OpenGL layer
 - Default Qt abstraction (output to PDF, SVG, pixmap, etc.)
- Callbacks for painting and events (everything happens in R)
- Examples:
 - R graphics device based on a scene graph
 - Mosaiq: a high-level graphics system similar to lattice

An animated tour

- Contributed by Hadley and Bei Bei
- An RGtk2 GUI control driving a Qt canvas