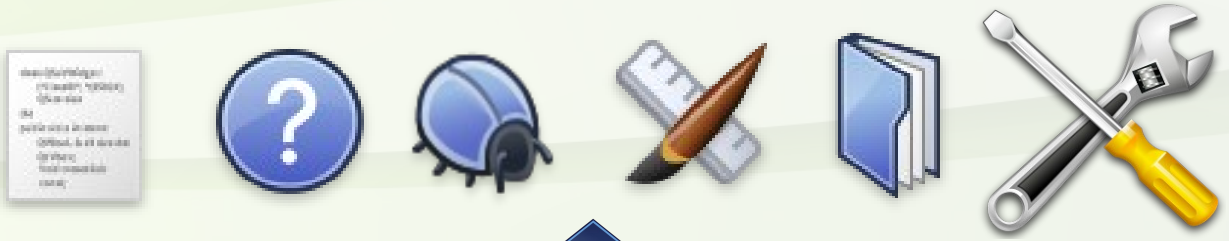
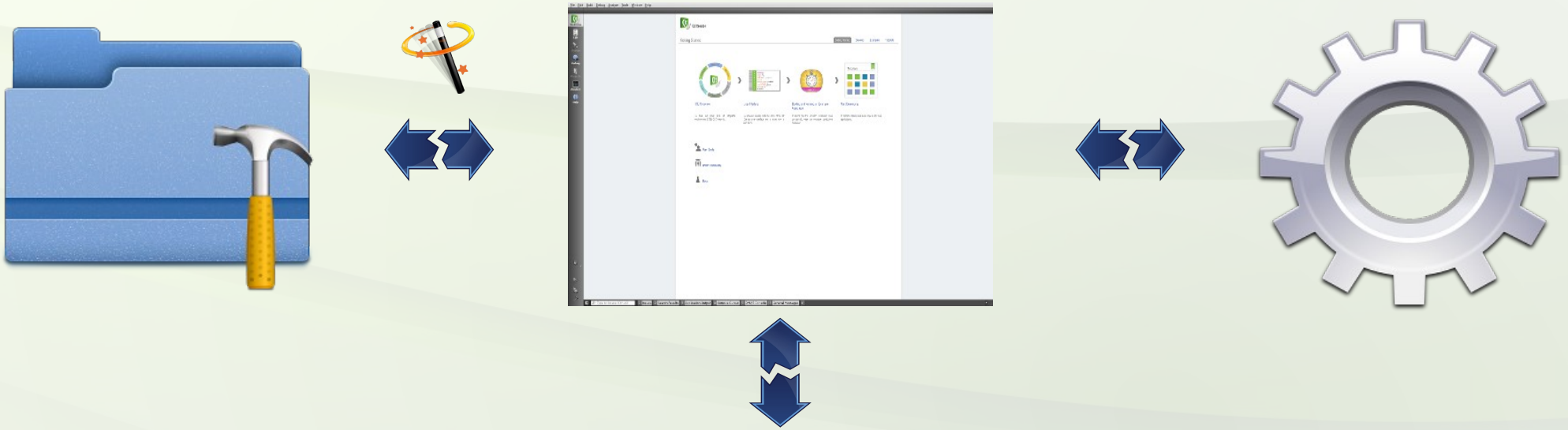
Extending Qt Creator
(without writing code)

Tobias Hunger





Configuration

Configuration

Qt Developer Days 2013

User configuration

- `~/.config/QtProject/(QtCreator*|qtcreator/*)`
- Can be changed by "`-settingspath <path>`" argument
- sqlite db + `.ini` file
- XML files (and more) in `qtcreator` subfolder
- Leave alone

- Possible exception: Sessions (`*.qws`)

Configuration

Qt Developer Days 2013

System wide configuration

- `../share/qtcreator/QtProject/*`
- Same as above (without sqlite file)
- XML files, `.ini`-file
- Use `sdktool` to edit XML files

Configuration

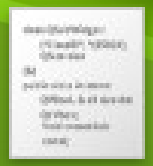
Qt Developer Days 2013

Project configuration

- `.user` and `.shared` file
- XML files
- No tool for `.shared` file, leave `.user` alone

- **Do not check in `.user` files!**

Built-in Tools

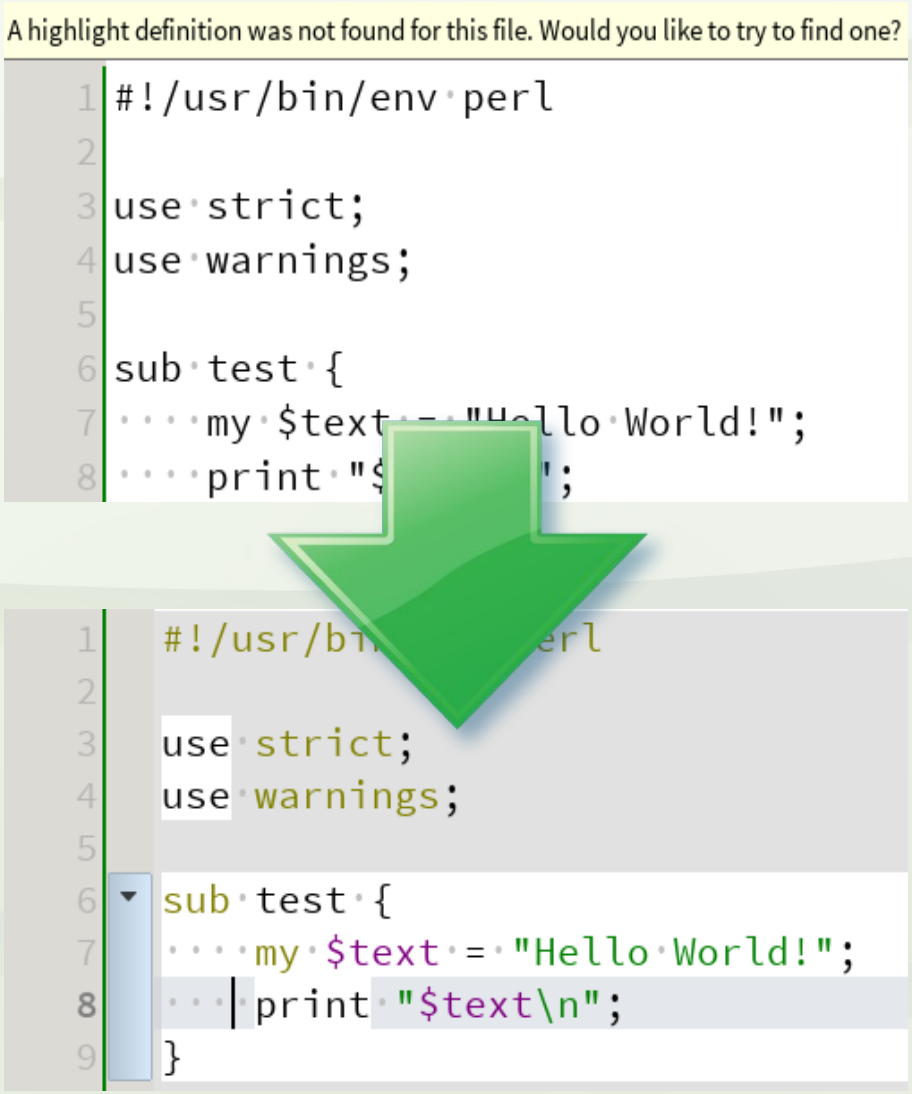


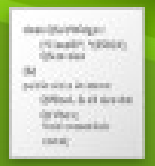
Editors

Generic Highlighters

- Configure via
  `Tools>Options>`
  `Text Editor>`
  `Generic Highlighter`

- or by putting files into
  `../share/qtcreator/`
  `generic-highlighter`





Editors

Macros

- Custom complex edit operations
- Configure via `Tools>Macros`
- No global configuration, user configuration in

  `…/QtProject/qtcreator/macros` (binary files)

Documentation

Qt Developer Days 2013

Custom Documentation

- Configure via
  `Tools>Options>Help>Documentation`

```
#include <iostream>

int main(int argc, char *argv
{
    std::cout() << "tmp" << s
    appPath
    printf("\nQt Creator Debu
    printf("Running query pro
    qDumpObjectData440(1, 42,
```

std::basic_ostream F1

- or by putting
  `[Help]`
  `InstalledDocumentation=/path/to/dir`
  (Comma separated list of paths) into `QtCreator.ini`

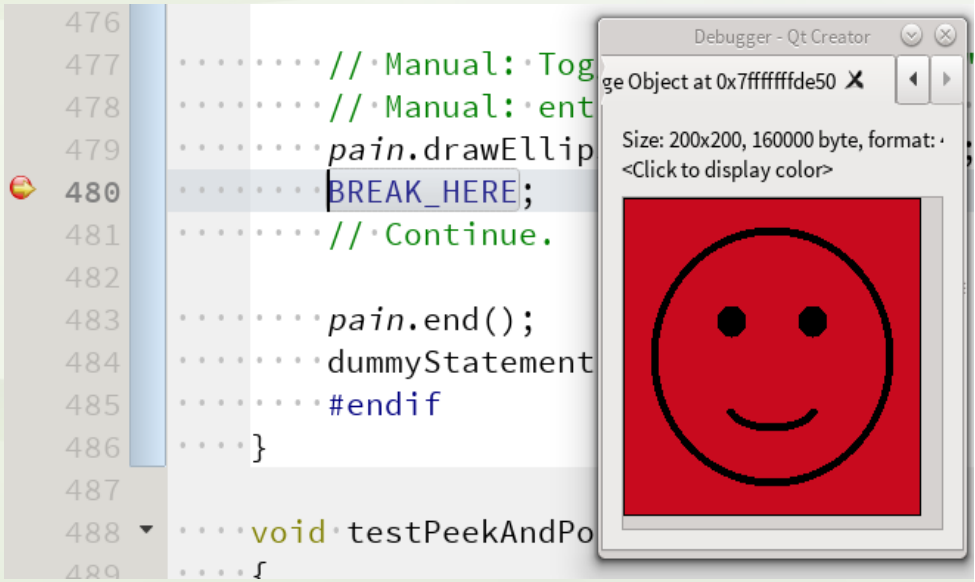
Debugger

- GDB/LLDB

  Python code in `shared/qtcreator/dumper/qttypes.py`

  or register own files in
  `Tools>Options>Debugger>GDB>Additional Startup Commands`

  - **`qtdump__`**`type__name(d, value)`
  - **`qtedit__`**`type__name(d, value)`



- CDB

  C++ code in `src/libs/qtcreatorcdbext` (Qt Creator sources!)

Designer

Qt Developer Days 2013

- Designer plugins for custom widgets

  Pitfall: Plugin needs to be built for Qt Creator, **not** for your project!

- Qml Designer offers similar feature for custom QML components

Projects

- Custom Build-/Clean-/Deploy Steps
- Run Custom Executables
- Environment
- Variable Substitution:
  - `%{CurrentBuild:Name}`,
  - `%{CurrentProject:FilePath}`,
  - `%{CurrentDocument:Row}`,
  - `%{CurrentKit:Id}`,

  ... many more!

External Tools

External Tools

Qt Developer Days 2013

- Run stand-alone tools from Qt Creator
  - Configure executable, arguments, stdin, etc.
    - Many variables available
  - Output can be displayed or replace selected text or update the editor

- Edit using `Tools>External>Configure...`
- Drop files into `share/qtcreator/externaltools`

Task Lists

- Import lists of “Build Issues” into Qt Creator
  - Quickly navigate issues with `F6`/`Shift-F6`
- Watches file for changes
- `file\tline\t(warning|error)\tmessage`
  (`'\n'` is expanded in message)

Code Generation

Snippets

Add custom code snippets, modify existing snippets to suit your needs

- add via `Tools>Options>Text Editor>Snippets`
- XML files in `share/qtcreator/snippets`

Wizards

Create custom (sets of) files

- `wizard.xml`: Metadata + file list + description of UI fields
- Templates of other files to generate

Create subdirectory for your wizards in `share/qtcreator/templates/wizards`

You need to do more?

Qt Developer Days 2013

Qt Creator Plugin Development

Aurindam Jana
Tobias Hunger

Documentation

- Qt Creator Manual:
  http://qt-project.org/doc/qtcreator/index.html
- Extending Qt Creator Manual:
  http://doc-snapshot.qt-project.org/qtcreator-extending/extending-index.html

- Qt Creator mailing list:
  http://lists.qt-project.org/mailman/listinfo/qt-creator
- IRC:
  #qt-creator on freenode.net

Thank you!

tobias.hunger@digia.com
hunger on IRC (freenode.net)