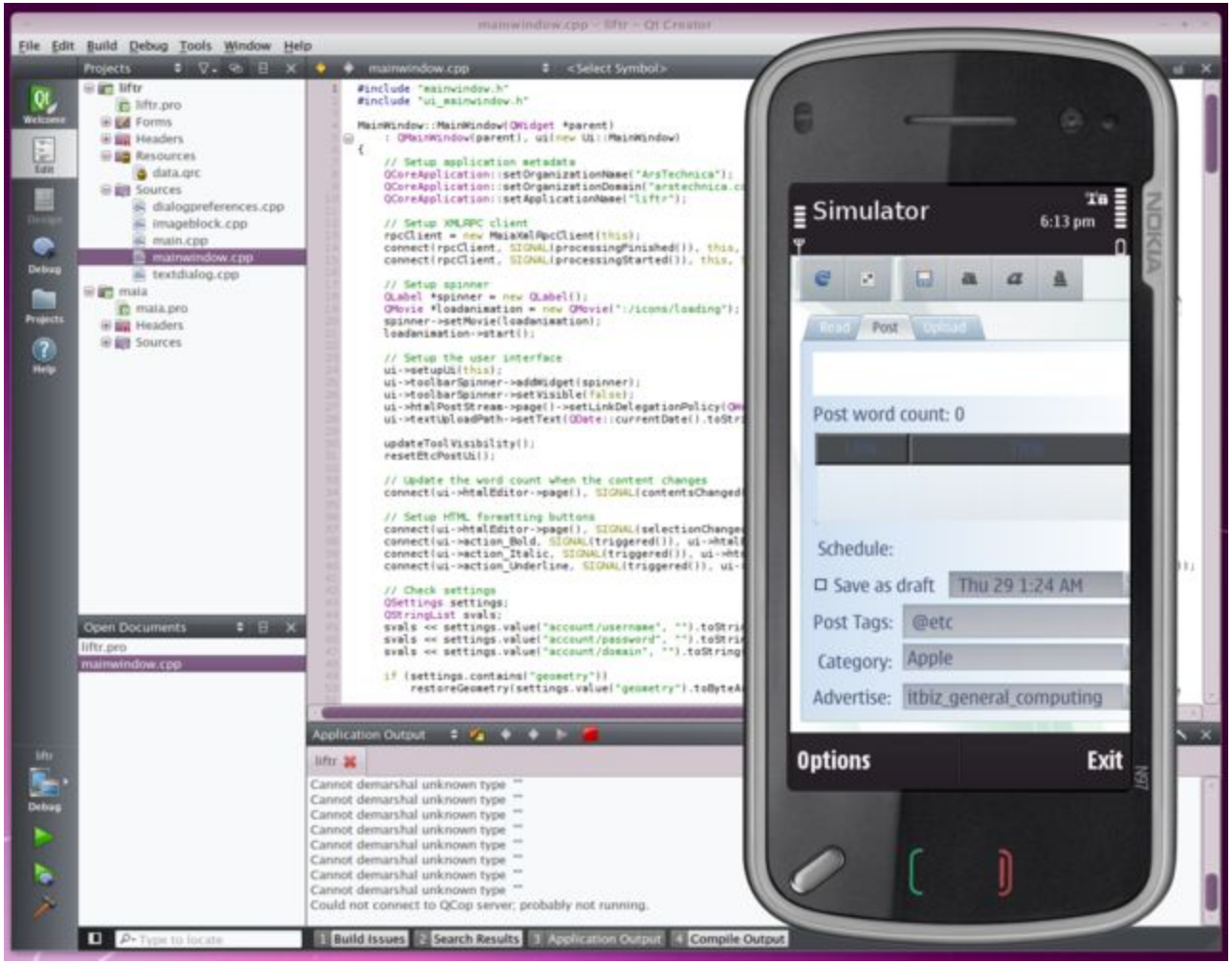
Journey to the land of containers

Introduction

- Yuvraaj Kelkar
- Journey of two build systems to the land of containers
- Embedded Linux Conference 2017, Portland

Journey 1: My first ever mobile app

- Just for fun: A cross-platform app for Symbian (circa 2008)
- First toolchain: Nokia Qt SDK for Symbian


mainwindow.cpp - liftr - Qt Creator
File Edit Build Debug Tools Window Help
Projects
liftr
liftr.pro
Forms
Headers
Resources
data.qrc
Sources
dialogpreferences.cpp
imageblock.cpp
main.cpp
mainwindow.cpp
textdialog.cpp
maia
maia.pro
Headers
Sources
Open Documents
liftr.pro
mainwindow.cpp
Application Output
Cannot demarshal unknown type
Could not connect to QCop server; probably not running.
Simulator
6:13 pm
Post
Post word count: 0
Schedule:
Save as draft
Thu 29 1:24 AM
Post Tags:
@etc
Category:
Apple
Advertise:
itbiz_general_computing
Options
Exit
NOKIA

Journey 1: My first ever mobile app

- Just for fun: A cross-platform app for Symbian (circa 2008)
- First toolchain: Nokia Qt SDK for Symbian
- Next toolchain: Linux Qt SDK

Download – Übersicht | Qt Project

qt-project.org

Register | Sign in

Qt Project

Downloads Documentation Forums Wiki Groups Blogs | Qt Digia

Search

Qt is a cross-platform application and UI framework for developers using C++ or QML, a CSS & JavaScript like language. Qt Creator is the supporting Qt IDE. Qt Cloud Services provides connected application backend features to Qt applications.

Qt, Qt Quick and the supporting tools are developed as an open source project governed by an inclusive meritocratic model. Qt can be used under open source (GPL v3 and LGPL v2.1) or commercial terms.

1. Get started

- Download
- Getting started
- Learning guides

2. Learn more

- Learning videos
- Wiki
- Documentation

3. Join us

- Forums
- IRC Channels
- Contributions

Log in

Username: Password: Log in

Keep me logged in for two weeks

Forgot your password?

Save time and log in using OpenID

New to Qt?
Get started here

5.3 available now

True cross-platform framework
for desktop, embedded and mobile

Learn about Qt 5.3

Get Creative with Qt Enterprise Embedded

Journey 1: My first ever mobile app

- Just for fun: A cross-platform app for Symbian (circa 2008)
- First toolchain: Nokia Qt SDK for Symbian
- Next toolchain: Linux Qt SDK
- Next toolchain: Maemo/Harmattan Qt SDK


Simulate
Generic
Storage
Network
Location
Contacts
Messaging
Sensors
Scripting
Application
View
Device Maemo Fremantle
Rotate Device
Zoom
Native size
Native resolution
NOKIA
N900
Simulator
Remove
Remove
Remove
New Item
Application Output
shoppinglist
Starting /home/yngve/Programming/shoppinglist-build-simulator/shoppinglist...
Messaging : Added storage configuration for account 10
Messaging : Migrated content data for account 10 to version 101
Build Issues
Search Results
Application Output
Compile Output

Journey 1: My first ever mobile app

- Just for fun: A cross-platform app for Symbian (circa 2008)
- First toolchain: Nokia Qt SDK for Symbian
- Next toolchain: Linux Qt SDK
- Next toolchain: Maemo/Harmattan Qt SDK
- Next toolchain: Blackberry Cascades

Momentics File Edit Source Refactor Navigate Search Project Run Window Help Sun 2:58 PM

QML Editing - example1/assets/main.qml - QNX Momentics IDE - /Users/jperih/ndk-10.0.9-workspace

BlackBerry 10 Native SDK 10.0.9.1673

Project

- 169.254.0.1
- 192.168.11.132
- 192.168.2.106
- example1
  - JavaScript Resc
  - Binaries
  - Includes
  - src
  - arm
  - assets
    - main.qml
  - translations
  - x86
  - bar-descriptor

*main.qml

BB10 Dev Alpha (1280 x 768)

```
// Default empty project template
import bb.cascades 1.0

TabbedPane {
    sidebarState: SidebarState.Hidden
    Tab {
        title: "Hub"
        Page {
            Label {
                text: "This is the Hub Page"
            }
        }
    }
    Tab {
        title: "Notifications"
    }
    Tab {
        title: "Text Messages"
    }
}
```

TabbedPane > Tab 38%

Outline

- TabbedPane
  - Tab
    - Page
      - Label
  - Tab
  - Tab

Search For

QML Properties

Tab

ID

General

Description	
Content	Page {}
New Content Available	☐
Unread Content Count	0
Image Source	
Title	Hub
Enabled	☑

Search For

C A

Panes

Controls

Search For

Writable Insert 13 : 1

Journey 1: My first ever mobile app

- Just for fun: A cross-platform app for Symbian (circa 2008)
- First toolchain: Nokia Qt SDK for Symbian
- Next toolchain: Linux Qt SDK
- Next toolchain: Maemo/Harmattan Qt SDK
- Next toolchain: Blackberry Cascades
- Next toolchain: Android Necessitas

TappyChickenMap

File Edit Window Help

TappyChicken

Search For Help

Modes

Save | Source Control | Content | Marketplace | Settings | Blueprints | Matinee | Build | Play | Launch

Search Classes

Recently Placed
Basic
Lights
Visual Effects
BSP
Volumes
All Classes

World Outliner

Search...

Type

World
Edit BP_Ma
Brush
PlayerStart

View Options

Devices

✓ GT-N7000 (00194a023f311f)
AC-KOMPUTER

Launching a game on a different device will change your default 'Launch' device in the toolbar

iOS Support

Project Launcher...
Device Manager...
Supported Platforms...

Project Settings

General Settings
Input
Navigation Mesh
Navigation System
Network
Physics
Rendering
Tutorials
User Interface

Platforms

Android
Android SDK
HTML5
iOS
Linux

Platforms - Android SDK

Settings for Android SDK (for all projects)

Set as Default | Export... | Imp

Search

SDKConfig

Location of Android SDK (the directory u	C:/NVPACK/android-sdk-windows
Location of Android NDK (the directory u	C:/NVPACK/android-ndk-r9c
Location of ANT (the directory usually c	C:/NVPACK/apache-ant-1.8.2
Location of JAVA (the directory usually	C:/NVPACK/jdk1.6.0_45
SDK API Level (specific version, 'latest',	matchndk
NDK API Level (specific version or 'lates	android-19

Content Browser

Add New | Import

Search Folders

Content
Audio
Blueprints
Effects
Fonts
Maps
Materials
Meshes
SetupFiles

Audio | Blueprints | Effects | Fonts | Maps | Materials | Meshes | Setup Files

8 items

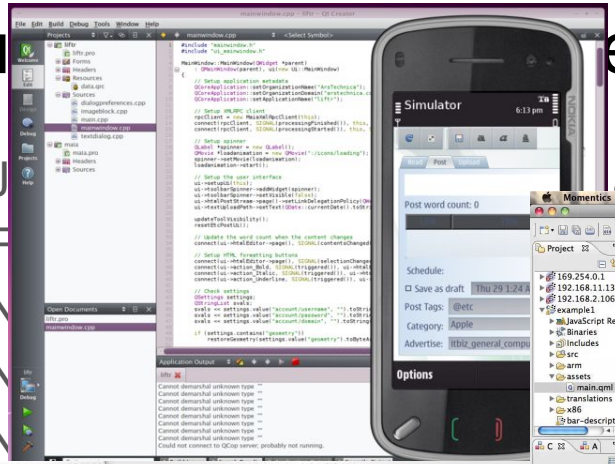
View Options

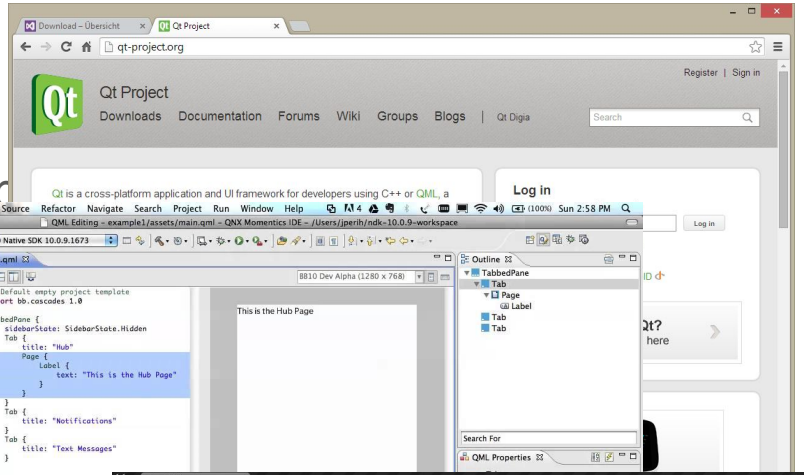
Tester aplikac... | Ask a questio... | Epic Games L... | TappyChicke... | ProjectLaunc... | Project Settings

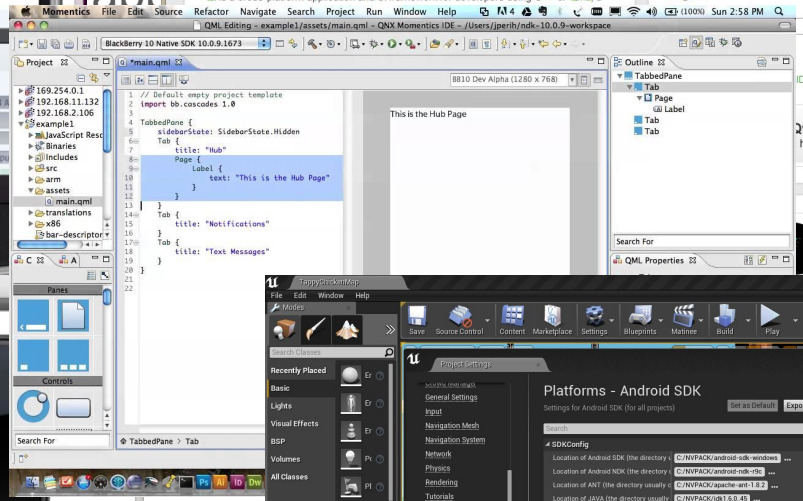
PL 22:57

Jou er

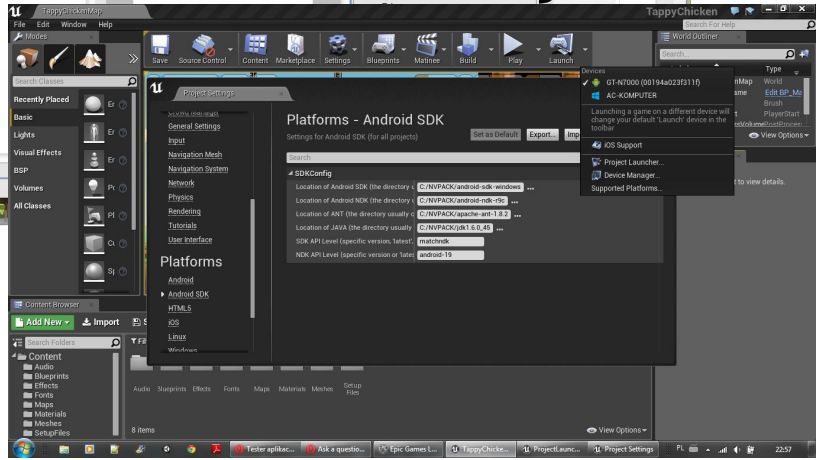
- J
- F
- N
- N
- N


Simulator
Post word count: 0
Schedule:
Save as draft
Post Tags:
Category:
Advertise:
Options


Download – Übersicht
Qt Project
qt-project.org
Register | Sign in
Qt Project
Downloads
Documentation
Forums
Wiki
Groups
Blogs
Qt Digia
Search
Qt is a cross-platform application and UI framework for developers using C++ or QML, a
Log in


Momentics
File Edit Source Refactor Navigate Search Project Run Window Help
QML Editing - example1/assets/main.qml - QNX Momentics IDE - /Users/jperih/ndk-10.0.9-workspace
BlackBerry 10 Native SDK 10.0.9.1673
Project
*main.qml
// Default empty project template
import bb.cascades 1.0
TabbedPane {
sidebarState: SidebarState.Hidden
Tab {
title: "Hub"
Page {
Label {
text: "This is the Hub Page"
}
}
}
Tab {
title: "Notifications"
}
Tab {
title: "Text Messages"
}
}
This is the Hub Page
Outline
TabbedPane
Tab
Page
Label
Tab
Tab
QML Properties
Panes
Controls
Search For
TabbedPane > Tab


Simulate
Generic
Storage
Network
Location
Contacts
Messaging
Sensors
Scripting
Application
View
Device Maemo Fremantle
Rotate Device
Zoom
Native size
Native resolution
NOKIA
Simulator
Remove
Remove
Remove
New Item
Application Output
Starting /home/jmpxe/Programming/shoppinglist-build-simulator/shoppinglist...
Messaging : Added storage configuration for account 10
Messaging : Migrated content data for account 10 to version 101


TappyChicken
File Edit Window Help
Save
Source Control
Content
Marketplace
Settings
Blueprints
Matinee
Build
Play
Launch
Modes
Recently Placed
Basic
Lights
Visual Effects
BSP
Volumes
All Classes
Platforms - Android SDK
Settings for Android SDK (for all projects)
Set as Default
Export...
General Settings
Input
Navigation Mesh
Navigation System
Network
Physics
Rendering
Tutorials
User Interface
Platforms
Android
Android SDK
HTML5
iOS
Linux
Windows
Content Browser
Add New
Import
Content
Audio
Blueprints
Effects
Fonts
Maps
Materials
Meshes
SetupFiles
8 items
Devices
GT-N7000 (00194a023f3110)
AC-KOMPUTER
iOS Support
Project Launcher...
Device Manager...
Supported Platforms...
World Outliner
Search For Help
View Options

Aaargh!

- Keep track of all toolchains
- ... on multiple machines
- … while trying to keep base OS updated
- … and some toolchains just cannot work with newer OSes.

When exactly am I supposed to focus on my app?

Virtualization to the rescue

- One VM for each toolchain
- No side-loading required
- No stepping on each other
- Whatever LTS level each toolchain wants

Everything works again. Yay!

About those VMs...

- So many VMs!
- 20GB HDD, 1-4 GB RAM each
- Desktop could run only one or two VMs at a time
- Wimpy laptop was wimpy, couldn’t even run one
- Irritant: Develop, check it in, pull on virtual machine, compile there.
- Acceptable for final build, annoying during development
- More problematic because I was trying for feature parity on multiple targets

Virtualization is a gateway drug

Buy cheap hardware, pay the price

- HDD overheated
- Then died
- Lost all my VMs
- Re-install everything
- Backups?

Denial or Zen?

Pause and recap: What do I want?

My build system(s) need to be:

- Independent of each other
- Independent of the underlying host and hardware
- Easily usable from multiple machines
- Easy to start and stop
- Not be heavy on system requirements
- No specialized hardware or ops work

Enter the Docker

Enter the Docker

- Install each toolchain in its own image.
- Independent of each other and the host hardware
- Use from wherever: desktop / laptop / wife’s laptop / AWS / GCE / Azure
- Speed of start/stop is incredible compared to VM
- CPU/memory overhead is negligible
- Storage cost was 100 MB compared to 20 GB for VM
- No need for specialized server hardware or virtualization software
- No more babysitting VMs!

Dockerfiles and Docker Hub

- Dockerfiles are awesome!
- Check in the build environment into source control
- Change and revert using git
- Reproducibility: No matter where I compile
- Peer review, open source
- One docker run command to compile the code
- Integrating into an IDE was a single `docker run` command per target

GVApi_login.cpp [uv-2017-02-06] - qgvdial - Qt Creator

File Edit Build Debug Analyze Tools Window Help

Welcome
Edit
Design
Debug
Projects
Help

Manage Kits...
Import Existing Build...

Active Project

desktop_linux

Build & Run

- Desktop
  - Build
  - Run
- Desktop Qt 5.7.1 GCC 64bit
  - Build
  - Run

Project Settings

- Editor
- Code Style
- Dependencies
- Clang Static Analyzer

Build Settings

Edit build configuration: Debug | Add | Remove | Rename...

General

Shadow build: ☑

Build directory: /home/uv/code/qgvdial/qgvdial/build-desktop_linux-Desktop-Debug | Browse...

Build Steps

qmake: qmake desktop_linux.pro -spec linux-g++-64 CONFIG+=debug | Details

Make: make -j4 in /home/uv/code/qgvdial/qgvdial/build-desktop_linux-Desktop-Debug | Details

Custom Process Step: docker run --rm -i -v `git rev-parse --show-toplevel`:/tmp/src accupara/qgvdial_ | Details

Command: docker | Browse...

Arguments: run --rm -i -v `git rev-parse --show-toplevel`:/tmp/src accupara/qgvdial_qt5_amd64 bash -c 'cd /tmp/src/qgvd

Working directory: %{buildDir} | Browse...

Custom Process Step: docker run --rm -i -v `git rev-parse --show-toplevel`:/tmp/src accupara/qgvdial_ | Details

Command: docker | Browse...

Arguments: run --rm -i -v `git rev-parse --show-toplevel`:/tmp/src accupara/qgvdial_qt5_amd64 make -C /tmp/src/qgvdial

Working directory: %{buildDir} | Browse...

Add Build Step

Clean Steps

Make: make clean in /home/uv/code/qgvdial/qgvdial/build-desktop_linux-Desktop-Debug | Details

Add Clean Step

Build Environment

Use **System Environment** | Details

desktop_linux
Debug

Type to locate (Ctrl+K) | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 Debugger Console | 6 General Messages

Dat `docker run` command tho’

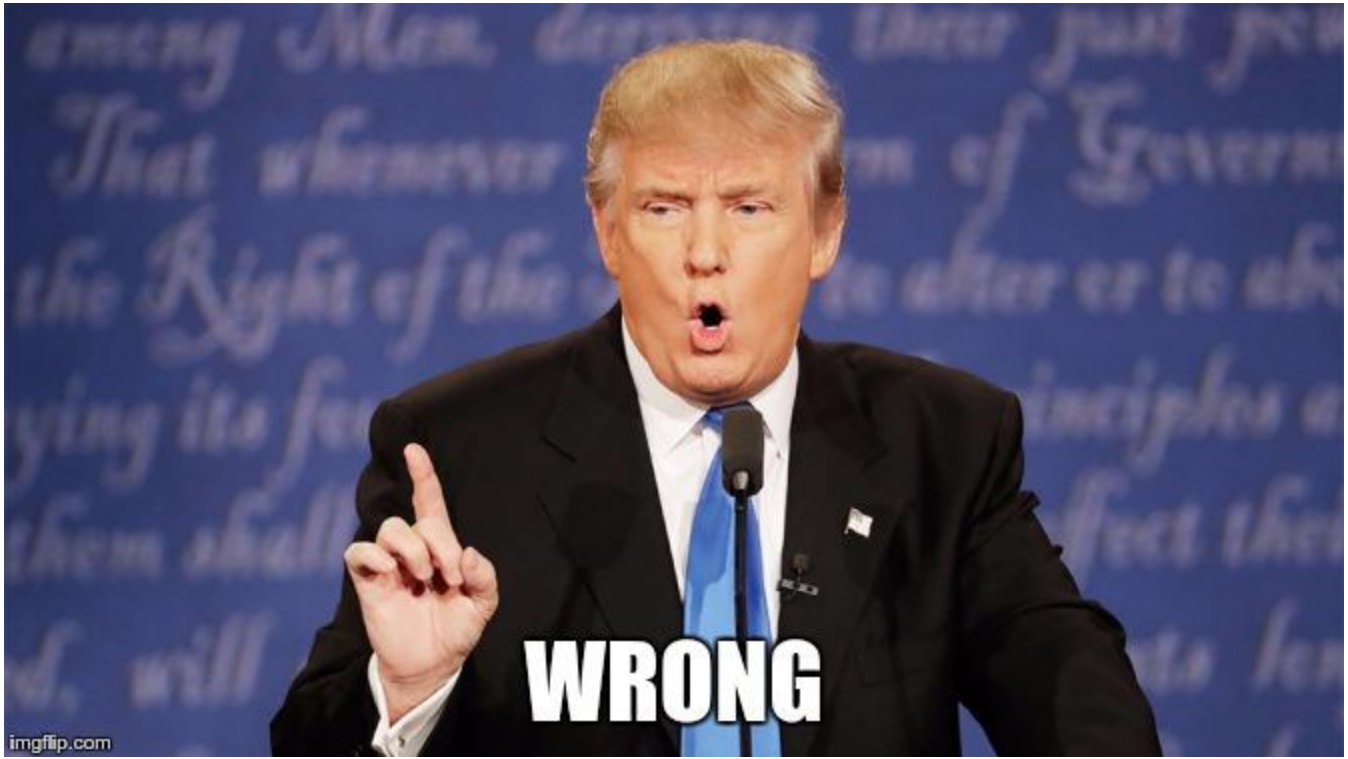
```
docker run \
    --rm -i \
    -v `git rev-parse --show-toplevel`:/tmp/src \
    accupara/qgvdial_qt5_amd64 \
    make -C /tmp/src/qgvdial/`basename %{buildDir}` -j4
```

Journey 1: Conclusions

- Use the right tool for the job
- The best solutions just work and get out of the way
- Even a simple mobile application benefits from a disciplined build environment
- Build systems invariably become complex and opaque
- Prioritize where to focus efforts by using existing tools

Journey 2: Customer environment

- Migrate a customer build environment into a container
- RHEL 6 build systems: Why? Legacy reasons.
- Should be simple right… ?


WRONG
imgflip.com

Journey 2: Customer environment

- Migrate a customer build environment into a container
- RHEL 6 build systems: Why? Legacy reasons.
- Should be simple right… ?
- Just install the dependencies, there’s nothing to it!


WRONG
imgflip.com

Journey 2: Customer environment

- Migrate a customer build environment into a container
- RHEL 6 build systems: Why? Legacy reasons.
- Should be simple right… ?
- Just install the dependencies, there’s nothing to it!
- The customer will know everything about **their own** build system


I'VE NEVER BEEN SO WRONG
MEMECREATOR.EU

Journey 2: Customer environment

Journey 2: Customer environment

- Create Dockerfile based off RHEL6 base image
- Get a list of all installed packages: `rpm -qa > rpm_qa.txt`
- Copy all the directories that were not in packages.
- Add hacks: `ANT_OPTS="-Xms1024m -Xmx1g -XX:MaxPermSize=2048m"`
- Add more hacks: `make SHELL='bash -x'`
- One step forward, 5 steps back
- Remove hacks: `sh` and `bash` are subtly different
- Sudden leap forward. First response: Disbelief.
- Clean up

Journey 2: How was this useful?

- Build & release team
  - Could see their own build system represented in code
  - Any modifications by developers were instantly visible
  - Modifications could be peer reviewed and then reverted if necessary
  - In other words: Infrastructure as code!
- Developer workflow and onboarding
  - No more “21 step wiki page”
  - BYO-OS
  - No restriction on “changing” the build system
- Enabled significant other improvements
  - Contact us

Sharing is caring

- GitHub repo to share container images
  - Linux, Qt apps, rsync, Android apps, Qemu, Qt library
  - https://github.com/accupara/docker-images
  - http://get.accupara.com/pup.py
- All containers available on the Docker Hub
  - https://hub.docker.com/r/accupara/
- Contribute: Pick your favourite project. Create a build container!
- Get in touch
  - Slack: https://accupara-community.slack.com
  - Email: contact@accupara.com
  - Twitter: @accupara

THANK YOU

Q & A