Free Software Automotive stack(s) that run on available hardware

with a demo on the Raspberry Pi 2

Jeremiah C. Foster • 30.01.2016

What is this talk about?

Patrick Ohnewein

Head of Free Software & Open Technologies Department at TIS innovation park

When I speak with Patrick he often asks me; “Can I run Free Software on my car?” The answer is, *it depends*.

It depends on having the right hardware and being able to get around the encryption automakers use on the boot image. I’ll try and outline what’s available today.

Outline

- Example of an IVI system with components
- Cars that run Linux
  - IVI (In-Vehicle Infotainment) vs. a complete automotive stack (ECU, RTOS, etc.)
- Example of an automotive system
- Overview of available hardware
- automotive specific engineering
- boot time and other challenges
- How to build on the Rpi2
- Resources (git repo URLs, wikis, etc.)


Main menu
18:38
91.2 MHz
Multimedia
Radio
Telephone
Navigation
Office
ConnectedDrive
Vehicle information
Settings
MODE
1
2
3
4
5
6
7
8
A/C
21°
AUTO
MAX A/C
OFF

Cars that run GNU/Linux

- BMW i3 2013
- Nissan
  - Infiniti Q50 2014
  - Infiniti Q30/QX30 2016
- Cadillac XTS, CTS 2013
  - Cadillac will be moving to Android/Linux
- Tesla Model S
- Toyota Lexus IS 2014
  - Coming soon: Volvo, PSA, JLR

Automotive software

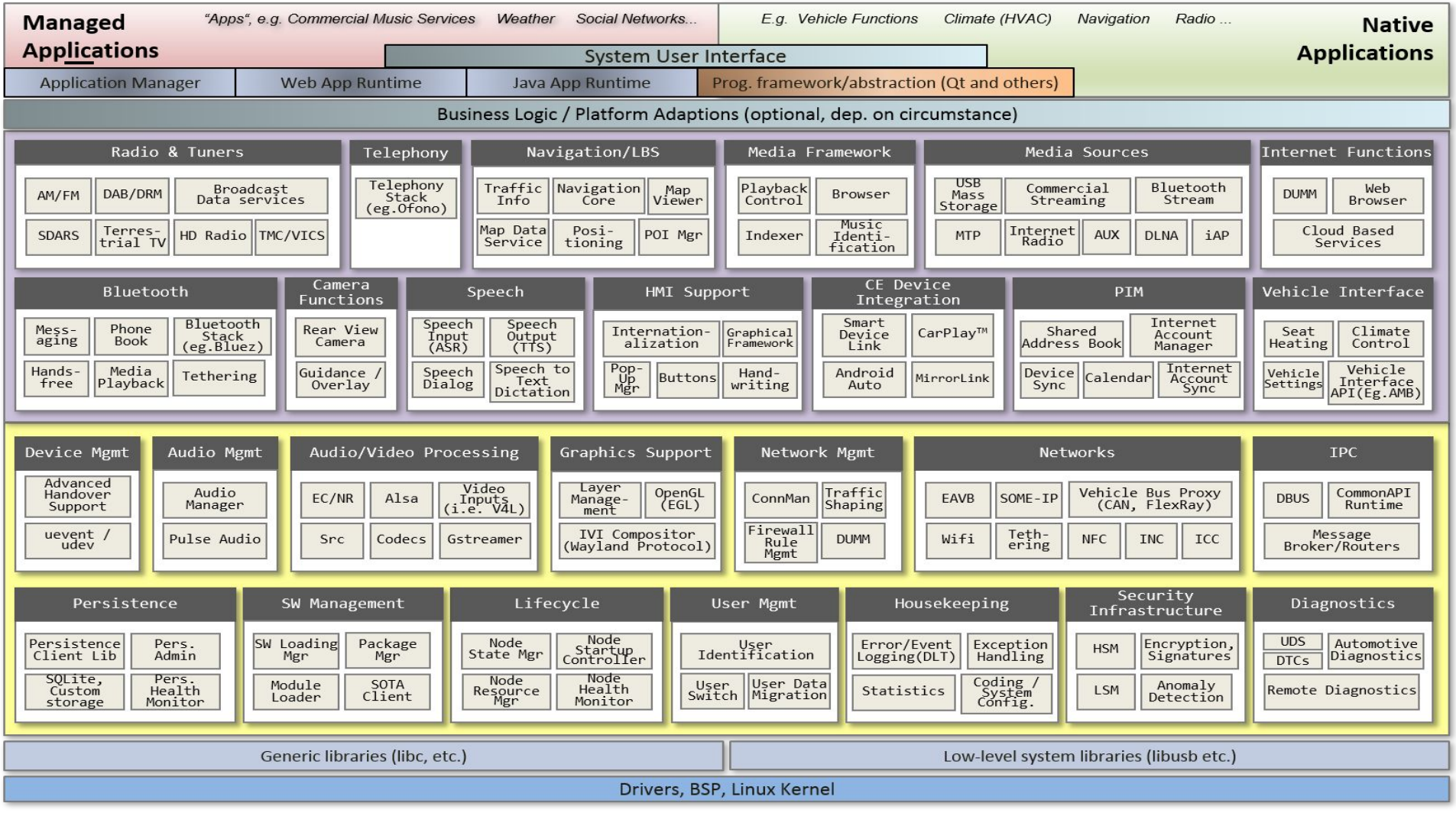
IVI

- Focus is on In-Vehicle infotainment and rich media
- Complex systems in a demanding environment, but not safety-critical
- Largely composed of commodity software components
- GNU/Linux becoming more widely used
- Uses automotive specific networking like MOST and EAVB

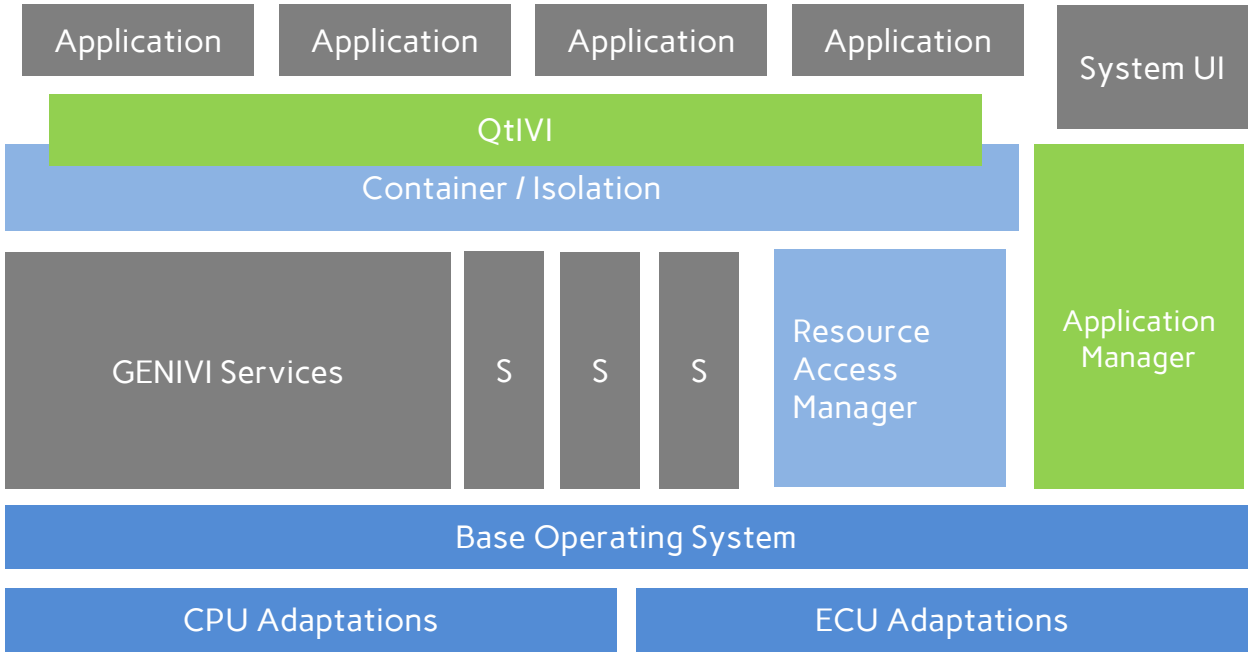
Complete stack

- Often has safety-critical and specific boot time requirements
- Widely regulated and certified (ISO 26262)
- Extremely complex system with ~200 million loc
- RTOS and/or virtualization widely used.
- GNU/Linux relatively new to this domain
- Automotive specific buses CAN and electrical systems AUTOSAR

In-Vehicle Infotainment


Managed Applications
"Apps", e.g. Commercial Music Services
Weather
Social Networks...
E.g. Vehicle Functions
Climate (HVAC)
Navigation
Radio ...
Native Applications
System User Interface
Application Manager
Web App Runtime
Java App Runtime
Prog. framework/abstraction (Qt and others)
Business Logic / Platform Adaptions (optional, dep. on circumstance)
Radio & Tuners
AM/FM
DAB/DRM
Broadcast Data services
SDARS
Terres-trial TV
HD Radio
TMC/VICS
Telephony
Telephony Stack (eg.Ofono)
Navigation/LBS
Traffic Info
Navigation Core
Map Viewer
Map Data Service
Posi-tioning
POI Mgr
Media Framework
Playback Control
Browser
Indexer
Music Identi-fication
Media Sources
USB Mass Storage
Commercial Streaming
Bluetooth Stream
MTP
Internet Radio
AUX
DLNA
iAP
Internet Functions
DUMM
Web Browser
Cloud Based Services
Bluetooth
Mess-aging
Phone Book
Bluetooth Stack (eg.Bluez)
Hands-free
Media Playback
Tethering
Camera Functions
Rear View Camera
Guidance / Overlay
Speech
Speech Input (ASR)
Speech Output (TTS)
Speech Dialog
Speech to Text Dictation
HMI Support
Internation-alization
Graphical Framework
Pop-Up Mgr
Buttons
Hand-writing
CE Device Integration
Smart Device Link
CarPlay™
Android Auto
MirrorLink
PIM
Shared Address Book
Internet Account Manager
Device Sync
Calendar
Internet Account Sync
Vehicle Interface
Seat Heating
Climate Control
Vehicle Settings
Vehicle Interface API(Eg.AMB)
Device Mgmt
Advanced Handover Support
uevent / udev
Audio Mgmt
Audio Manager
Pulse Audio
Audio/Video Processing
EC/NR
Alsa
Video Inputs (i.e. V4L)
Src
Codecs
Gstreamer
Graphics Support
Layer Manage-ment
OpenGL (EGL)
IVI Compositor (Wayland Protocol)
Network Mgmt
ConnMan
Traffic Shaping
Firewall Rule Mgmt
DUMM
Networks
EAVB
SOME-IP
Vehicle Bus Proxy (CAN, FlexRay)
Wifi
Teth-ering
NFC
INC
ICC
IPC
DBUS
CommonAPI Runtime
Message Broker/Routers
Persistence
Persistence Client Lib
Pers. Admin
SQLite, Custom storage
Pers. Health Monitor
SW Management
SW Loading Mgr
Package Mgr
Module Loader
SOTA Client
Lifecycle
Node State Mgr
Node Startup Controller
Node Resource Mgr
Node Health Monitor
User Mgmt
User Identification
User Switch
User Data Migration
Housekeeping
Error/Event Logging(DLT)
Exception Handling
Statistics
Coding / System Config.
Security Infrastructure
HSM
Encryption, Signatures
LSM
Anomaly Detection
Diagnostics
UDS
DTCs
Automotive Diagnostics
Remote Diagnostics
Generic libraries (libc, etc.)
Low-level system libraries (libusb etc.)
Drivers, BSP, Linux Kernel

Example stack



FOSS automotive stacks

GENIVI

- Consortium of automotive companies with BMW as a founder
- Spends ⅓ of its budget on FOSS development
- > 100 companies are members
- Fairly mature software base
- Built with Yocto and Baserock

Automotive Grade Linux

- Linux Foundation project
- Mostly populated by Japanese companies though Ford is a member
- Released a demo image at CES
- Built with Yocto

FOSS automotive stacks

Openivi

- Created by start-up in the automotive and telematics industry
- Still early stages
- OpenIVI Mobility is a complete system for rapidly prototyping mobility concepts,
- Qt/HTML

Tizen IVI

- Linux Foundation project
- Automotive code from Tizen mostly incorporated into AGL distro
- Uncertain future for IVI category of Tizen despite being in production in passenger vehicles

Available hardware

Renesas Porter Board

Renesas Silk Board

Raspberry Pi 2

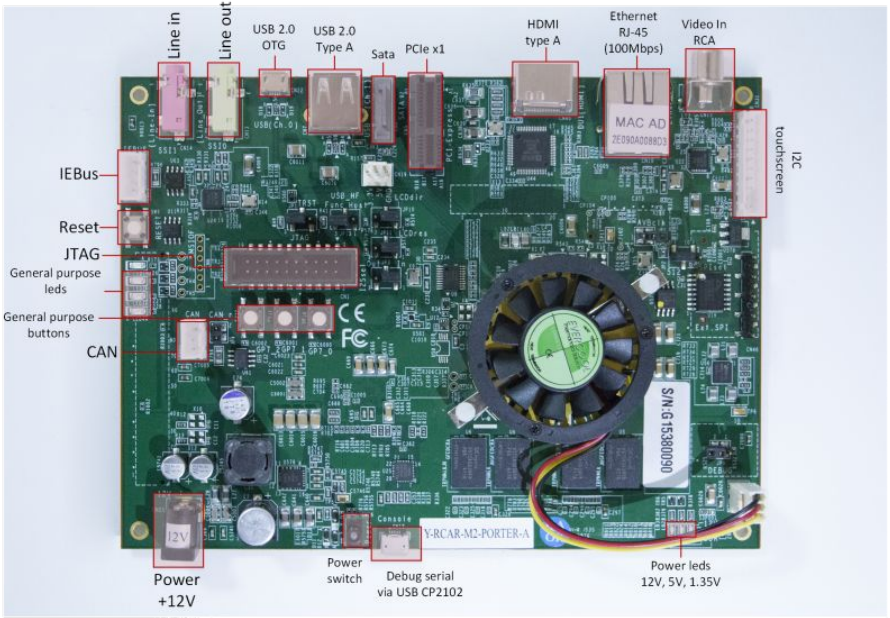
Minnowboard Max

Wandaboard

Jetson TK1

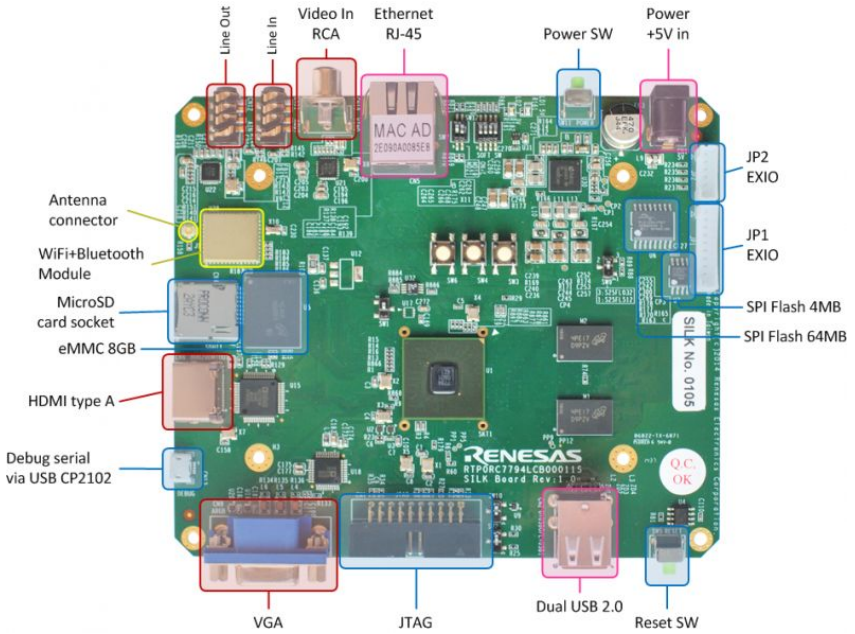
Renesas Porter board

- Renesas is a large automotive silicon vendor from Japan
- Automotive sample boards also available
- Widely used hardware in automotive, particularly in Asia
- 1.5 GHz ARM dual core Cortex-A15
- 2 GB DDR3 memory (dual channel)
- BSP on GitHub
- Available: http://www.digikey.com/product-search/en?keywords=Y-RCAR-M2-PORTER
- ~360 USD



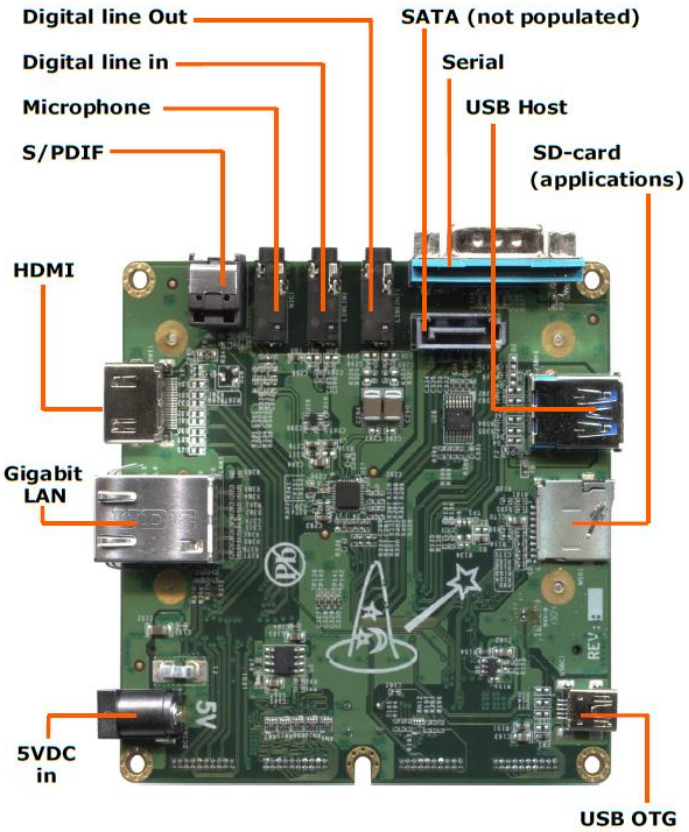
Renesas Silk board

- R-Car E2 SoC
- ARM Dual Core Cortex-A7
- GPU: PowerVR SGX540
- 1 GB DDR3 memory
- Works in the community, in AGL, GENIVI, LTSI kernel project, etc.
- Available: http://www.digikey.com/product-search/en?keywords=Y-RCAR-E2-SILK-A
- ~312 USD



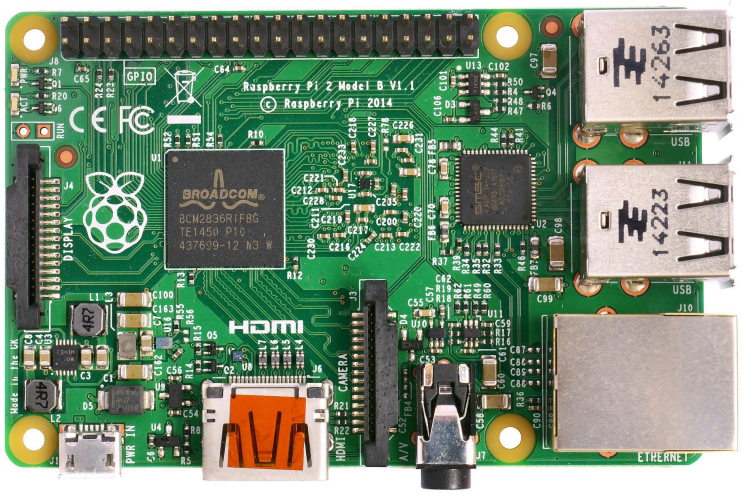
Wandaboard

- Freescale i.MX6 Duallite
- ARM Dual Core Cortex-A9
- GPU: Vivante GC 880 + Vivante GC 320
- 1 GB DDR3 memory
- iMX is widely used in the automotive industry (Freescale is now part of NXP)
- Available: http://www.wandboard.org/buy
- ~99 USD (For the mid-range unit)



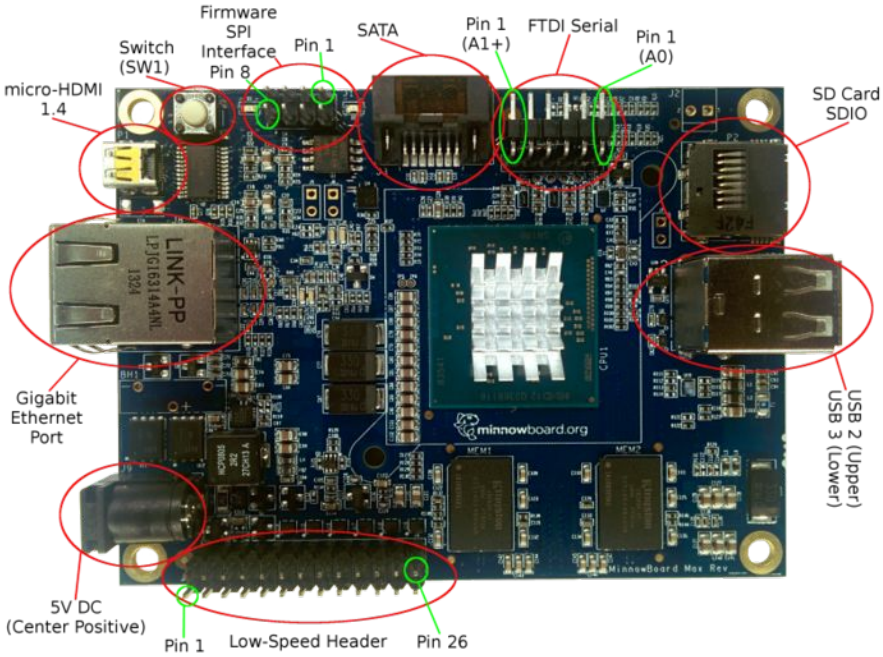
Raspberry Pi 2

- Backed by the Raspberry Pi foundation
- Broadcom CPU
- 900 MHz Quad-core Cortex-A7
- 1 GB LPDDR2 SDRAM memory
- Available: anywhere (Rpi magazine, Adafruit, Amazon, etc.)
- ~35 USD



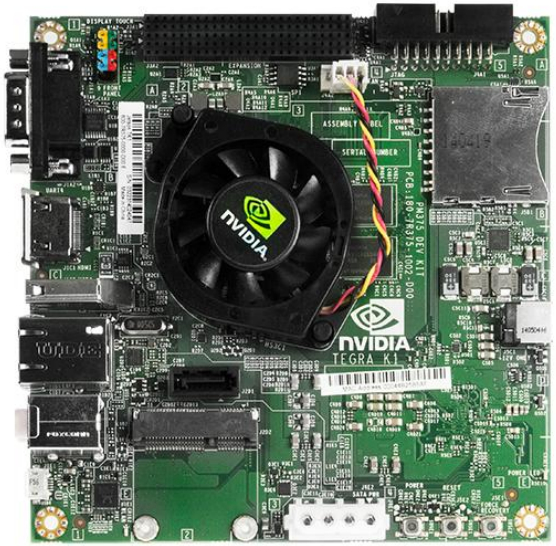
Minnowboard Max

- 64 bit Intel Bay Trail Atom
- Dual core 1.33 GHz
- Integrated Intel HD Graphics with Open Source hardware-accelerated drivers for Linux OS
- “Owned” by minnowboard.org foundation
- Debian GNU, Ubuntu, Fedora, Mint
- Yocto Project Compatible
- Available: http://wiki.minnowboard.org/Where_to_buy
- ~140 USD

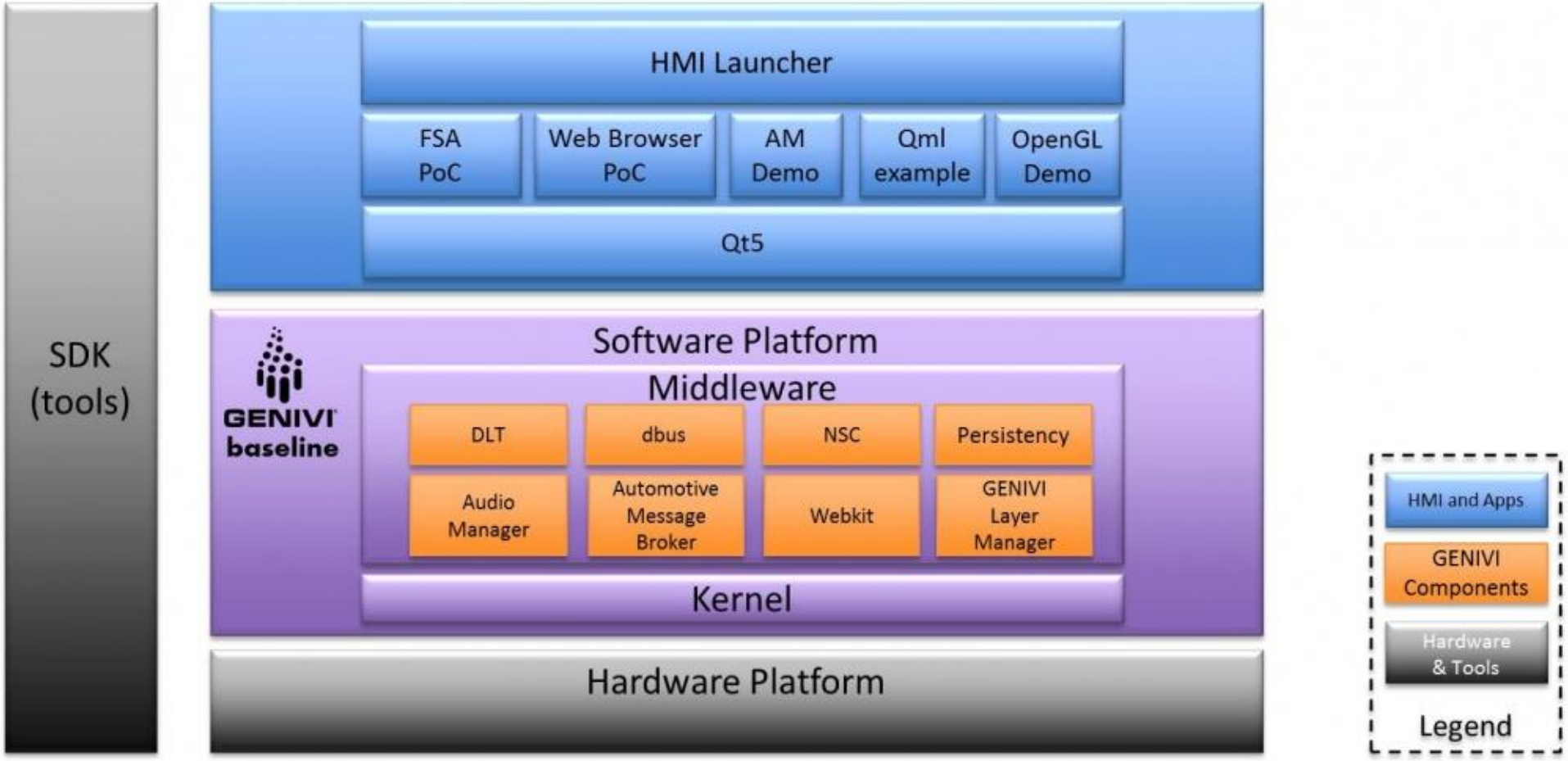


Nvidia Jetson TK1

- NVIDIA Tegra K 1
- Quad-Core ARM Cortex-A15 "r3"
- 2.3 GHz Max clock speed
- DDR3L and LPDDR3 memory up to 8 Gig
- Available from NVIDIA’s “store”
- ~200 USD



- Holds nearly all of the GENIVI components including dependencies and subsystems
- Evolving into a complete SDK or ADK
- A good starting point for the latest code


SDK
(tools)
HMI Launcher
FSA
PoC
Web Browser
PoC
AM
Demo
Qml
example
OpenGL
Demo
Qt5
Software Platform
GENIVI
baseline
Middleware
DLT
dbus
NSC
Persistency
Audio
Manager
Automotive
Message
Broker
Webkit
GENIVI
Layer
Manager
Kernel
Hardware Platform
HMI and Apps
GENIVI
Components
Hardware
& Tools
Legend

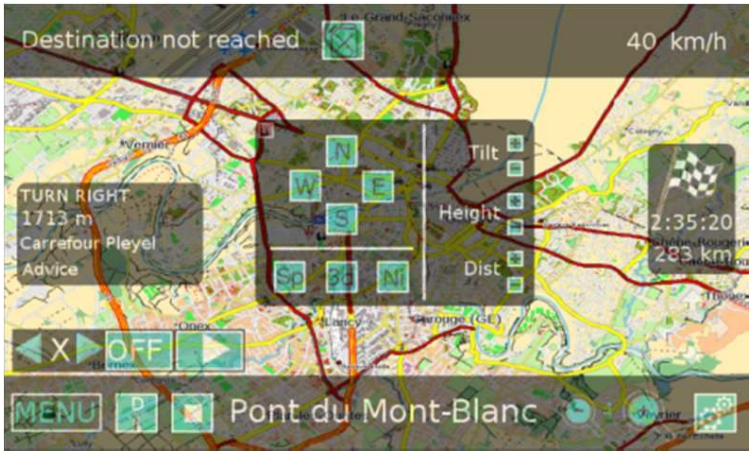
Fuel Stop Advisor

Proof of concept exercising numerous parts of GENIVI subsystems

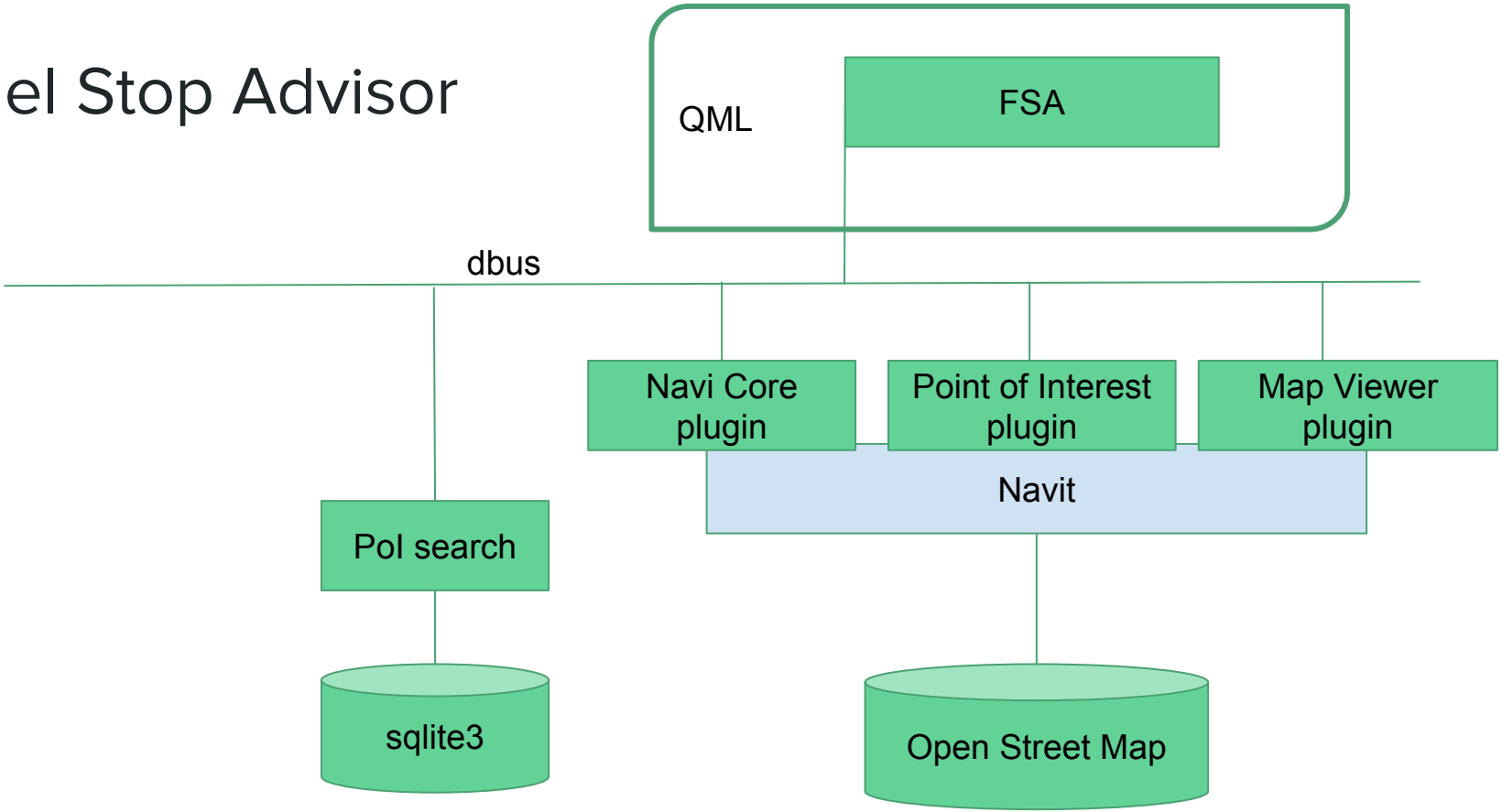
Enhanced tank distance based on the fuel consumption on the route ahead

Warning if destination not reached

Proposal of reroute to a refill station



Fuel Stop Advisor



Map viewer

dbus interface provided;

- org.genivi.mapviewer.MapViewerControl

Entire dbus interface documented and available online; http://git.projects.genivi.org/?p=lbs/navigation.git;a=blob_plain;f=doc/map-viewer/MapViewerAPI.pdf;hb=f0ddb754ad4e16d8f650485a610818c06e0ceac3

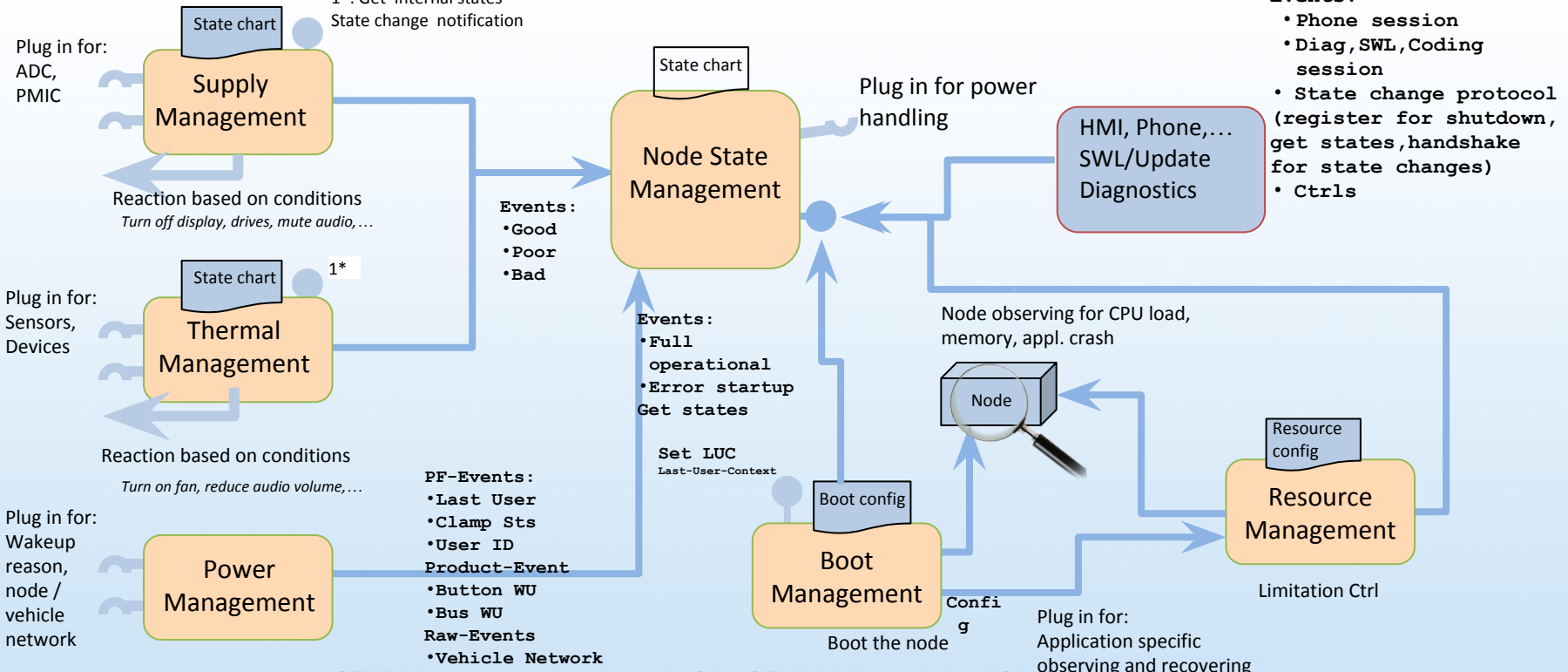
W3C positioning PoC

The PositionWebService is a simple proof of concept (PoC) showing how positioning information provided over D-Bus by the GENIVI EnhancedPositionService can be accessed within a web browser.

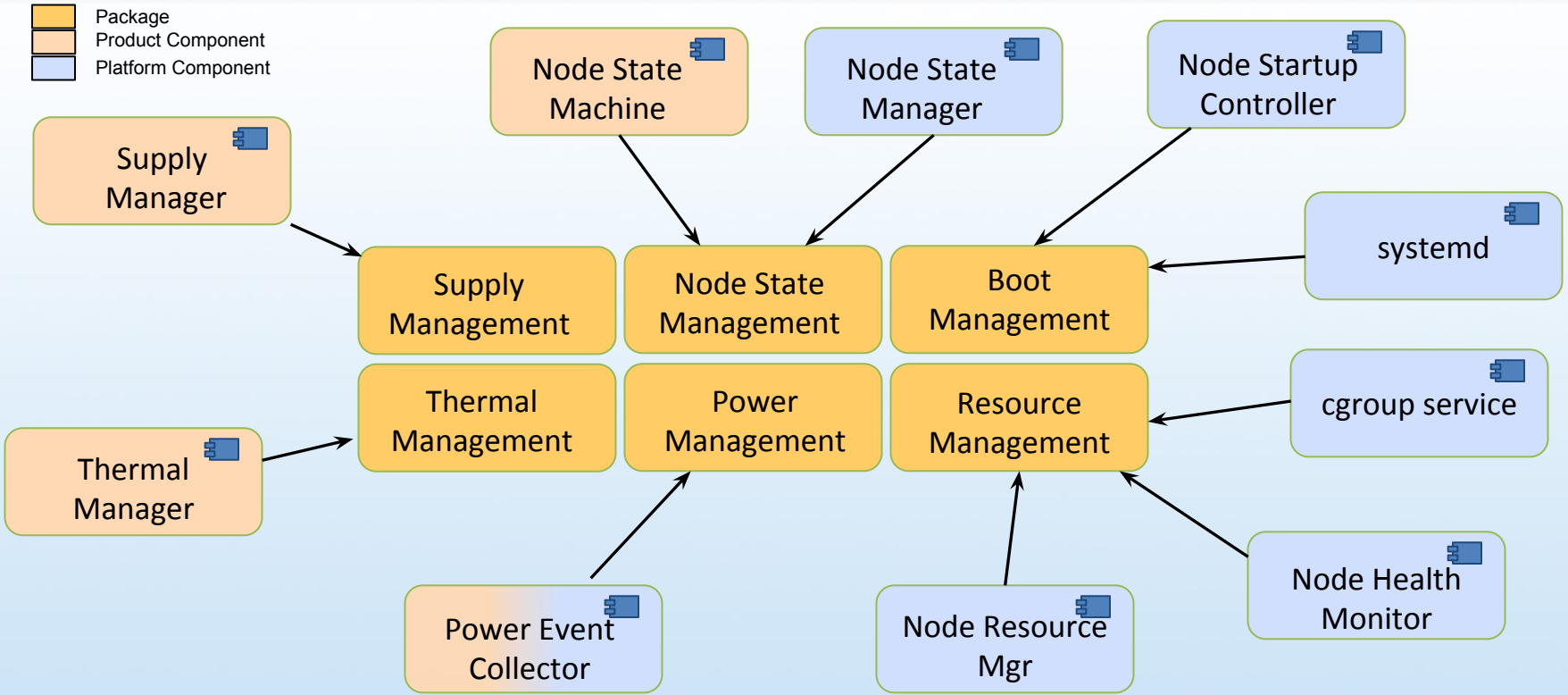
This PoC was developed to investigate how to match the already defined positioning dbus interface with the Web API being defined by the W3C

The translation D-Bus <-> JavaScript is realized using a FireBreath NPAPI plugin.

Lifecycle Overview

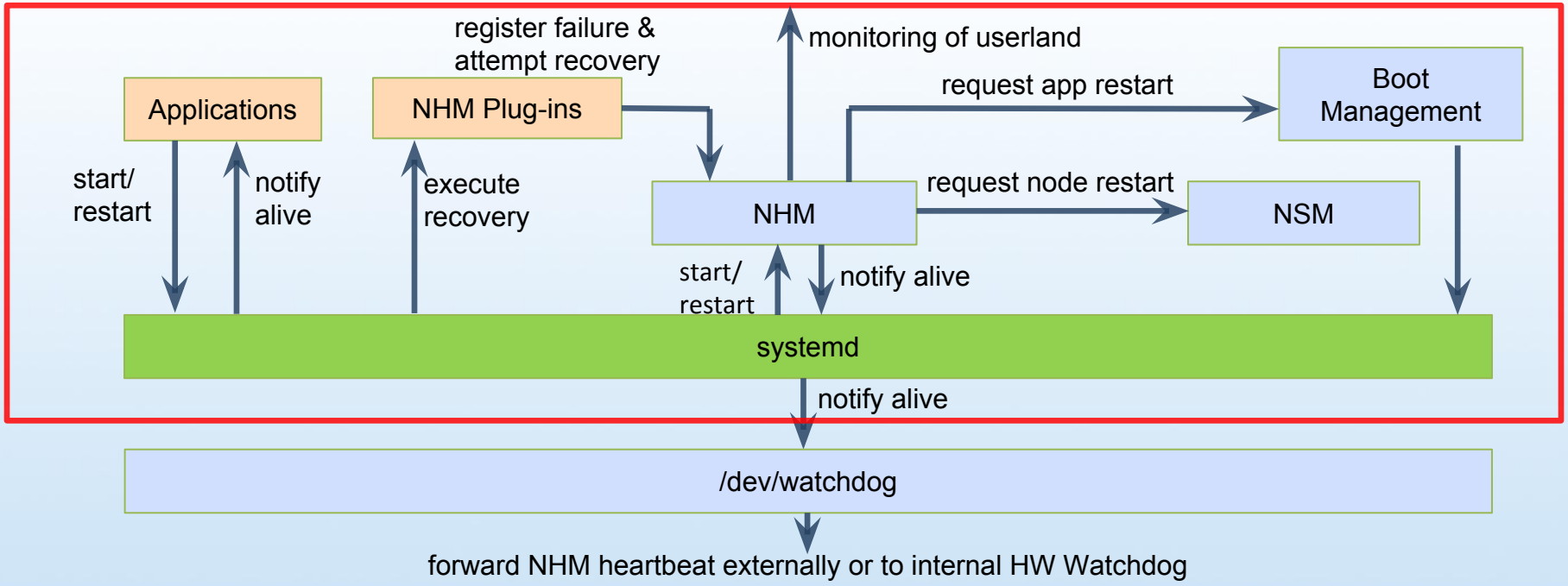


Lifecycle Manifest



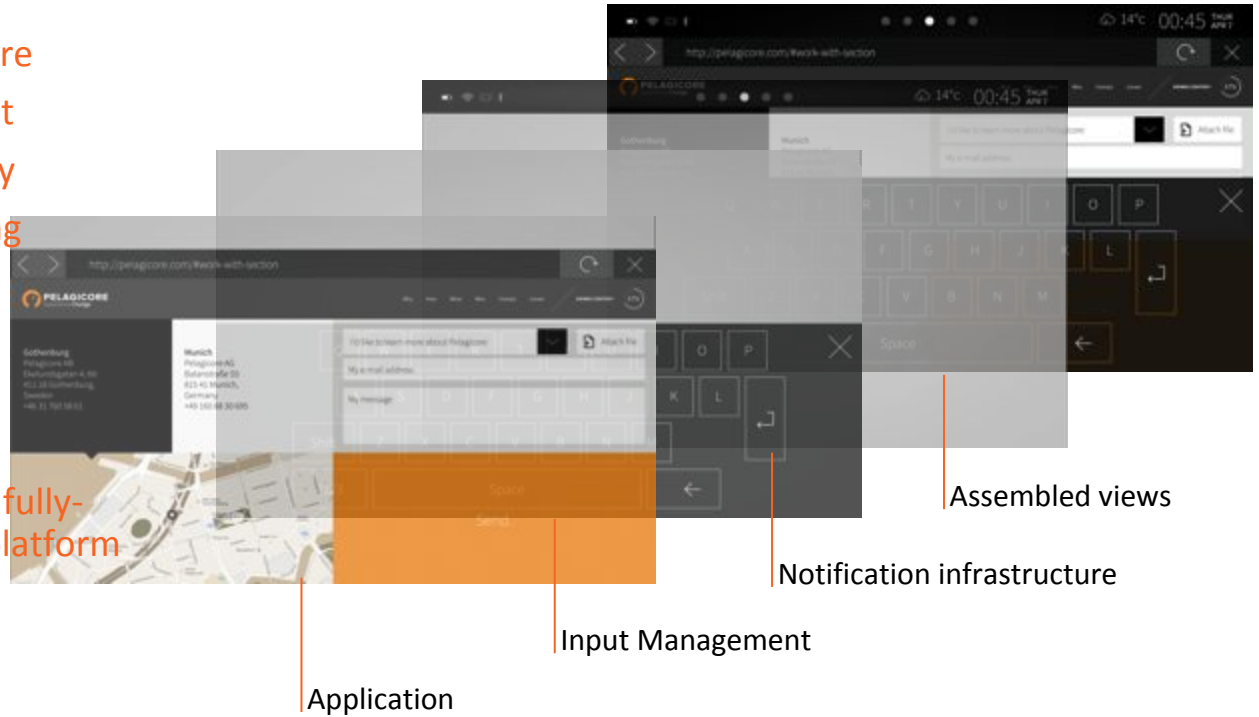
Health Management

Health Management will ensure that the node runs in a stable and defined manner. To do this it is planned to have the following multi layered observation system and escalation strategy:



Qt Compositing using Wayland

- Modern, multi-process architecture
- Application Lifecycle Management
- Security model to protect integrity
- Hardware accelerated compositing using Wayland
- OpenGL and HTML applications can be seamlessly composited
- Elevates Qt from being a UI and application framework to being a fully-fledged automotive UI software platform



Automotive Challenges

- Sudden loss of power
- Boot time requirements
- Aborted shutdown requirements
- FLASH wear
- Latency requirements
- Expected life of product

- Length of projects
- Size of projects
- Complex supplier relationships
- Purchasing processes
- …

Apps in Cars

- Remember MirrorLink?
- Who owns the data?
- Native applications
  - Large demand for this
  - Possible to add functions during the vehicle life-time
  - Matches the customer expectations
- Side effects
  - Partitioning the UI in exchangeable parts
  - Smaller updates

Apple CarPlay

android auto

Legal Challenges

- There is a difference between building a screen into a car and bringing a screen into the car
- Safety requirements
- Driver disruptions
- Driver workload management
- Driven by liability and legal requirements

1. Install 'repo'

The first thing to do in order to use this manifest, is to install the 'repo' tool wrapper, and that needs to be done on each machine (or user).

The following instructions can be used:

```
$ curl https://dl-ssl.google.com/dl/googlesource/git-repo/repo > /tmp/repo
$ chmod a+x /tmp/repo
$ sudo mv /tmp/repo /usr/local/bin/
```

Alternatively, if you don't have 'administrative' permission, or prefer to install in a user $HOME folder, you can do something along these lines:

```
$ mkdir ~/bin
$ curl https://dl-ssl.google.com/dl/googlesource/git-repo/repo > ~/bin/repo
$ chmod a+x ~/bin/repo
$ export PATH=~/bin:$PATH
```

Do not forget to add ~/bin permanently to your PATH.

2. Fetch all git trees

Initialize your local working repository:

```
$ mkdir -p ~/projects/genivi-rpi2
$ cd ~/projects/genivi-rpi2
$ repo init -u https://github.com/amirna2/genivi-manifest.git -b master
```

Checkout all project trees:

```
$ repo sync
```

3. Run the build setup script (this will create a build folder)

```
$ source ./buildenv/meta-ivi-rpi-init-build-env
```

4. Edit conf/bblayers.conf and conf/local.conf

For build/conf/bblayers.conf

```
BBLAYERS ?= " \
  /home/anathoo/projects/genivi-rpi2/poky/meta \
  /home/anathoo/projects/genivi-rpi2/poky/meta-yocto \
  /home/anathoo/projects/genivi-rpi2/poky/meta-yocto-bsp \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-openembedded/meta-oe \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-openembedded/meta-ruby \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-ivi/meta-ivi \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-ivi/meta-ivi-bsp \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-genivi-demo \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-qt5 \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-raspberry \
  "
BBLAYERS_NON_REMOVABLE ?= " \
  /home/anathoo/projects/genivi-rpi2/poky/meta \
  /home/anathoo/projects/genivi-rpi2/poky/meta-yocto \
  /home/anathoo/projects/genivi-rpi2/poky/../meta-ivi/meta-ivi \
  "
```

For build/conf/local.conf

```
MACHINE ??= "raspberrypi2"
GPU_MEM = "128"
CORE_IMAGE_EXTRA_INSTALL += "wayland weston"
LICENSE_FLAGS_WHITELIST  += "commercial"
PREFERRED_VERSION_weston ?="1.6.0"
MULTI_PROVIDER_WHITELIST += " \
             virtual/libgl \
             virtual/egl \
             virtual/libgles2 \
             virtual/mesa \
             "

#Comment out to avoid bitbake error with some GPLv3 licensed components
#INCOMPATIBLE_LICENSE ?= "GPLv3"
```

5. Start the build

```
$ bitbake -v genivi-demo-platform
```

6. Flash image on the SD card

Replace sdX with the correct device ID

```
$sudo umount /dev/sdX
$sudo dd if=./tmp/deploy/images/raspberrypi2/genivi-demo-platform-raspberrypi2.rpi-sdimg  of=/dev/sdX bs=128M
$sync
```

GENIVI

GENIVI provides a standard FOSS interface following community best practices;

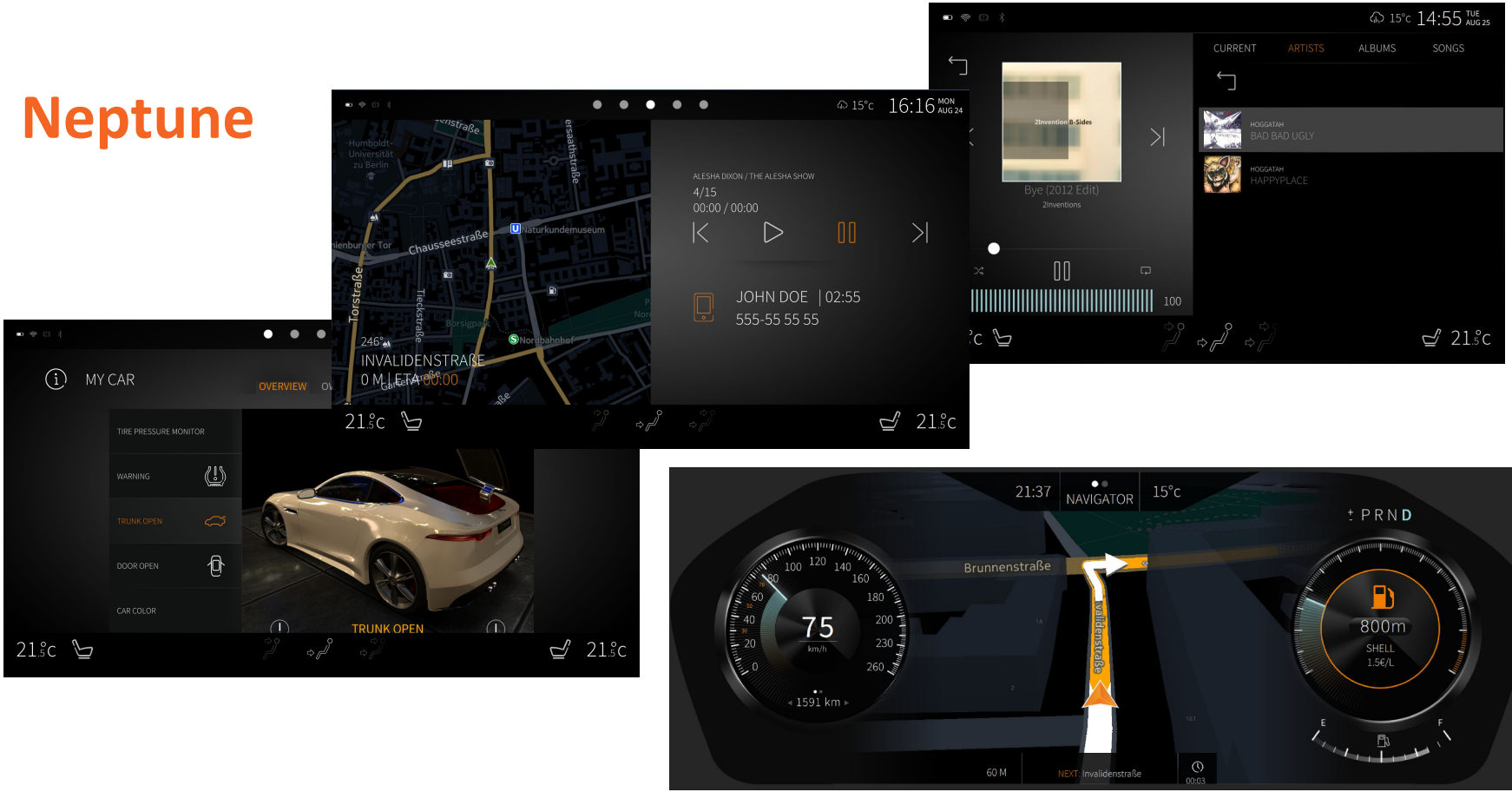
- World clone-able git repos: http://git.projects.genivi.org
- GENIVI Demonstrator Platform: https://at.projects.genivi.org/wiki/x/aoCw
- Mailing lists: https://lists.genivi.org/mailman/listinfo
- IRC: Freenode #automotive
- Wiki: http://wiki.projects.genivi.org/
- Web: http://projects.genivi.org/
- Propose a project: http://genivi.org/propose

Automotive Grade Linux

FOSS your ride

- Gerrit and git repos: https://gerrit.automotivelinux.org/gerrit/#/admin/projects/
- Mailing lists: https://www.automotivelinux.org/community/mailing-lists
- IRC: Freenode #automotive
- Wiki: https://wiki.automotivelinux.org/
- Demo: https://www.automotivelinux.org/news/news/2016/01/agl-shows-demo-ces-2016

Neptune


15°c 14:55 TUE AUG 25
CURRENT
ARTISTS
ALBUMS
SONGS
2Invention B-Sides
Bye (2012 Edit)
2Inventions
HOGGATAH
BAD BAD UGLY
HOGGATAH
HAPPYPLACE
100
21.5c
15°c 16:16 MON AUG 24
Humboldt-Universität zu Berlin
Chausseestraße
Naturkundemuseum
Torstraße
Tieckstraße
Nordbahnhof
246°
INVALIDENSTRAßE
0 M | ETA 00:00
ALESHA DIXON / THE ALESHA SHOW
4/15
00:00 / 00:00
JOHN DOE | 02:55
555-55 55 55
21.5c
21.5c
MY CAR
OVERVIEW
TIRE PRESSURE MONITOR
WARNING
TRUNK OPEN
DOOR OPEN
CAR COLOR
TRUNK OPEN
21.5c
21.5c
21:37
NAVIGATOR
15°c
PRND
Brunnenstraße
Invalidenstraße
75
km/h
1591 km
800m
SHELL
1.5€/L
60 M
NEXT: Invalidenstraße
00:03