Tizen IVI

Rusty Lynch
2012-10-12
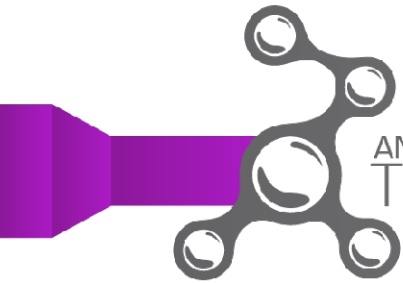
ANDROID FOR INTEL ARCHITECTURE INTEL LINUX WIRELESS GUPNP OPENSTACK POWRTOP LESSWATTS SYNCEVOLUTION
TIZEN XEN CONNMAN LINUX KERNEL YOCTO INTEL LINUX GRAPHICS K
OFONO SIMPLE FIRMWARE INTERFACE (SFI) ENTERPRISE SECURITY

Contents

This presentation will provide a brief introduction to Tizen IVI and then also introduce some of the areas undergoing active development, including:

- Centralized policy enforcement with Murphy
- Vehicle information access with the Automotive Message Broker
- Building a lightweight graphics stack using Wayland and Weston

In-Car-Infotainment (IVI)

- Does not include "dirty parts" of the car (i.e. engine controls)
- Can consist of multiple computer systems (ECU's) with multiple screens on each
- Common applications in today's car include:
  - Navigation
  - Satellite radio
  - Music player
  - Movie player
  - etc
- Traditionally a rigid RTOS but industry is moving towards a flexible / extensible OS with common off the shelf software

Tizen IVI Project

IVI on tizen.org

Downloads
http://download.tizen.org/previews/ivi/latest
Wiki Page
http://wiki.tizen.org/wiki/IVI

Email list
IVI@lists.tizen.org
http://lists.tizen.org/listinfo/ivi

Issue Tracking
http://bugs.tizen.org (Tizen IVI Project)

Register username to edit wiki or issues
https://www.tizen.org/user/register

Tizen IVI Development

- Regular snapshots available
  - Daily snapshots available (expect to bleed!)
  - Stable milestones available roughly once per month
- Tizen 2.0 Timeframe
  - Development happens in the profile/ivi gerrit git trees
  - As a package reaches Tizen 2.0 quality requirements then it will be migrated from profile/ivi to the common Tizen packaging area
- Future Tizen 3.0 goals
  - Address changes in core Tizen packages making it easier to support a rich variety of vertical platforms
  - Introduce some of the innovations from IVI development to other Tizen verticals

Requirements Gathering

- Automotive Grade Linux Working Group
  - *"The Workgroup will facilitate widespread industry collaboration that advances automotive device development, providing a community reference platform that companies can use for creating products. "*
- GENIVI
  - *"GENIVI® is a non-profit industry alliance committed to driving the broad adoption of an In-Vehicle Infotainment (IVI) open-source development platform."*

GENIVI® Compliance

"The GENIVI compliance program provides a set of specifications for GENIVI member companies to measure their products and services. Those that meet the specifications may be registered as GENIVI compliant..."

Specific Component = SC, Abstract Component = AC, Placeholder Component = PC

P1 = Mandatory P2 = Optional, must be fully disclosed if implemented For Placeholder Component requirements: P3 = Optional, does not have to be disclosed

Murphy: Intro

Murphy is a resource policy manager. It orchestrates shared resource usage in complex environments with limited user interaction capabilities. Murphy's mission is to follow and understand the current system state and then make decisions triggered by events. The decisions are communicated to enforcement points which in turn change the system state accordingly.

License: BSD

Language: C

Source: https://github.com/otcshare/murphy

Murphy: Examples of resource types

- Audio, such as the permission to play audio, audio routing and volume
- Video, such as the permission to place a window to a certain screen
- System, such as CPU and memory share
- Power management, such as CPU throttling and batch task management
- Network, such as allocation of limited network resources
- Thermal, such as limiting resource use to keep the system under some temperature threshold

Murphy: Architecture
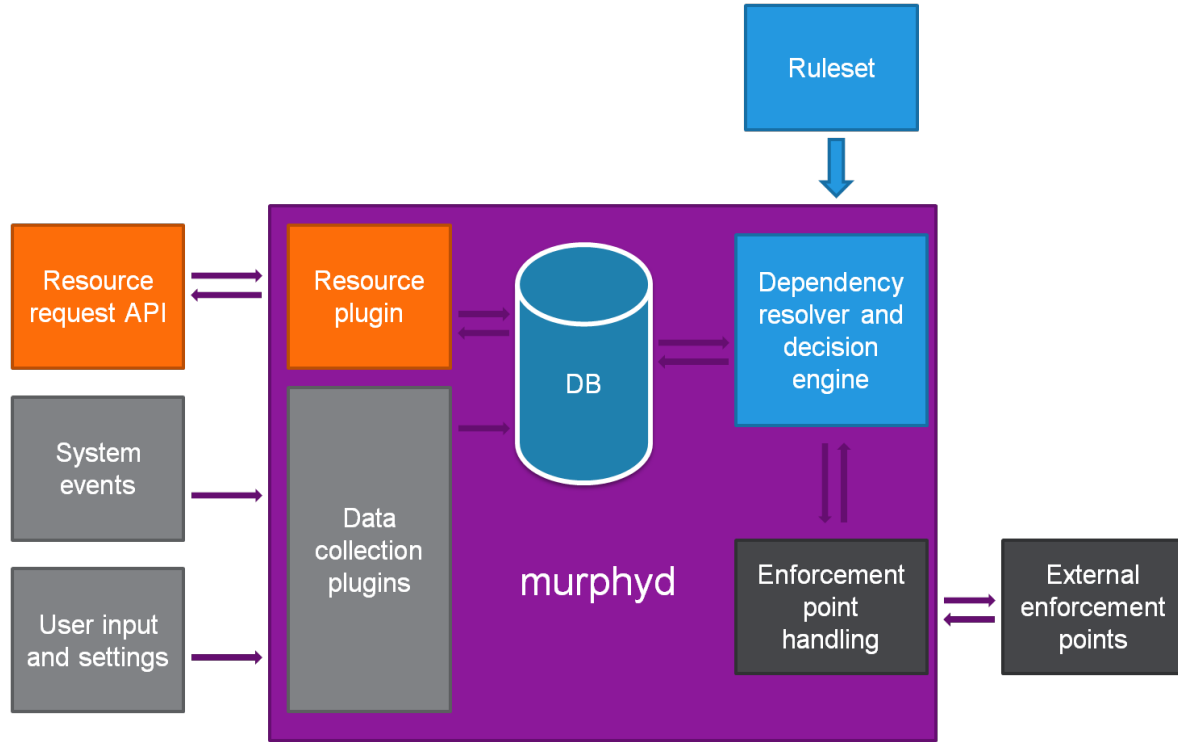
Example Usage Models

- Implementing resource "Zones"
- Ensuring applications meet driver safety compliance
- Support multiple ECU car architecture

Automotive Message Broker: Intro

Automotive Message Broker (AMB) is a vehicle network abstraction system. It brokers information from the vehicle to applications with an extensible source and sink plugin mechanism that accommodates chaining systems together.
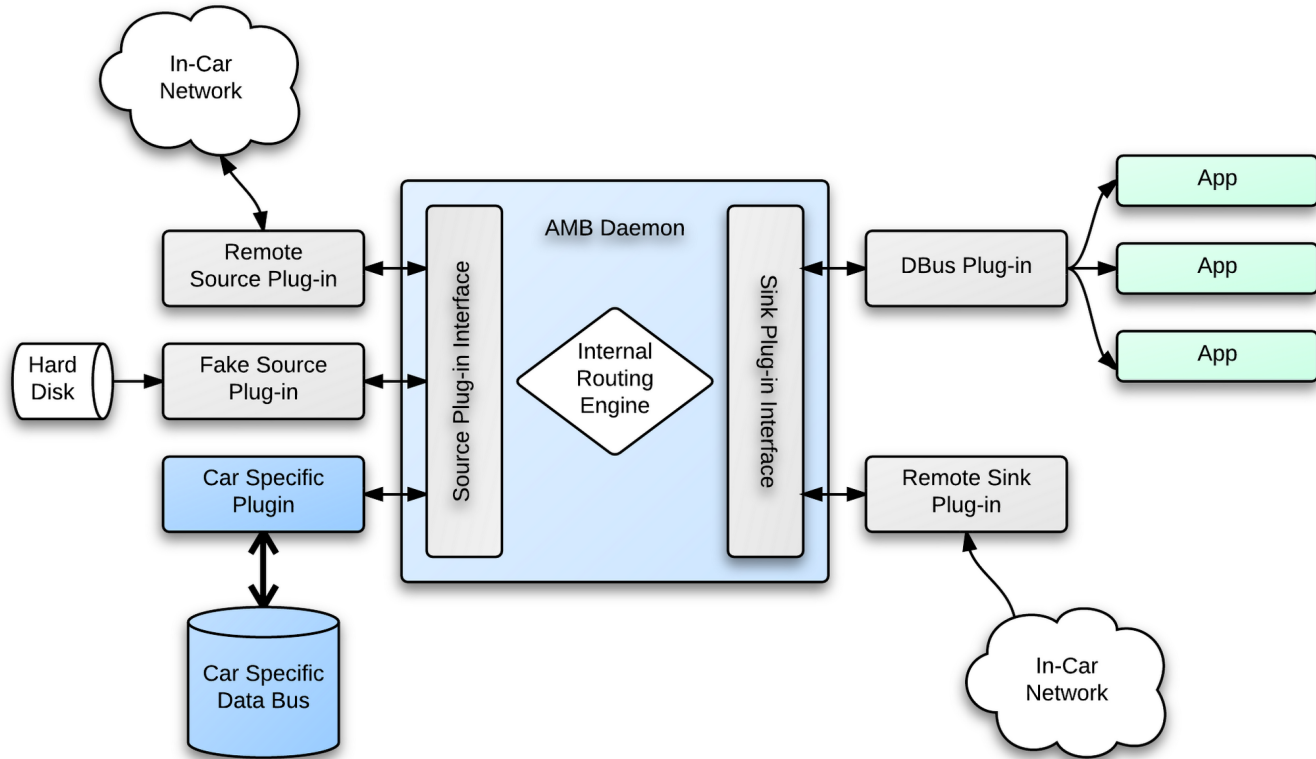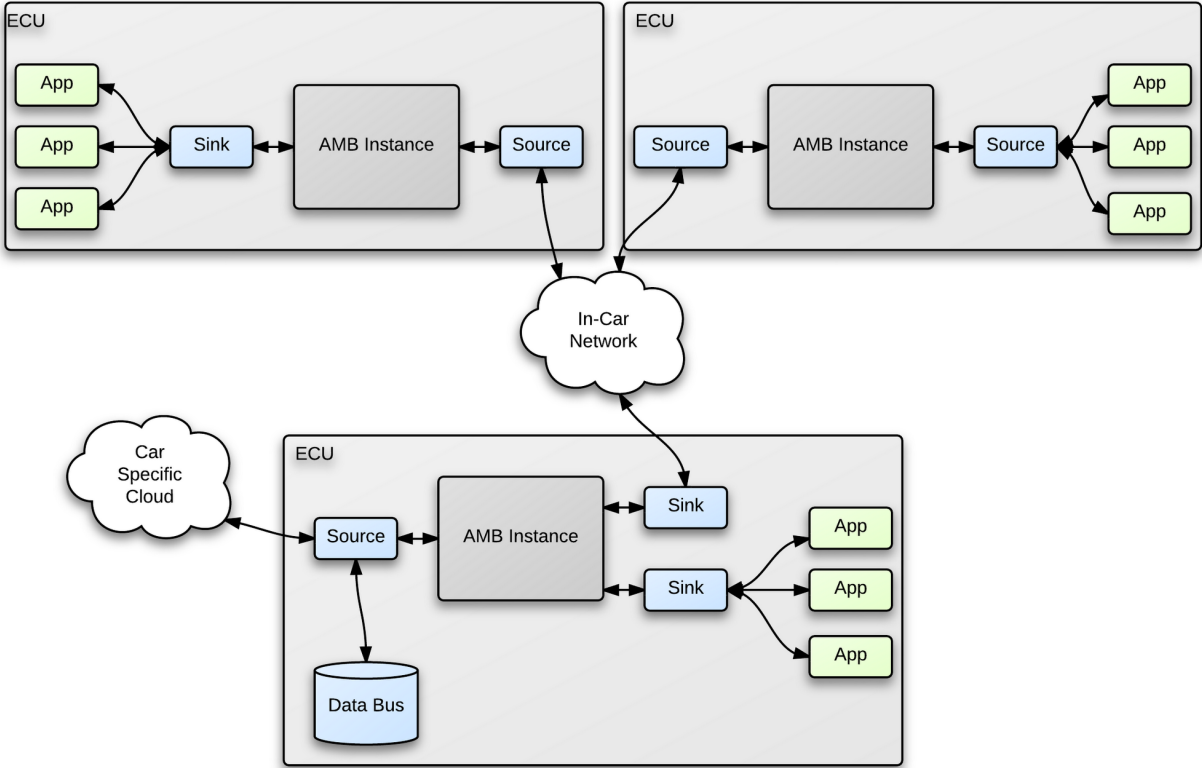
License: LGPL v2.1

Language: C++

Source: https://github.com/otcshare/automotive-message-broker

Project Page: https://01.org/projects/automotive-message-broker

AMB: Architecture

AMB: Car Level View

W3C Automotive API Implementation

Web is emerging in automotive. Car manufacturers and Tier-1s have very recently started to study the possibilities of Web and HTML5 to be used in application development and even to create flexible user interfaces (UI) in in-vehicle infotainment (IVI) systems.

In order to be able to provide automotive rich UI and applications, an access to IVI system provided data is needed. Ideally, this is done through standardized automotive Web APIs.

W3C Automotive API Goals

- Lightweight
  - The API is lightweight and provides getting and setting data items
- Initial Minimal Set
  - The initial proposed API implements an a minimal set of data items
- Query Support
  - The API implements a method for a Web application to query for supported data items, so graceful functionality degradation is possible

Wayland: Intro

Wayland is a protocol for a compositor to talk to its clients as well as a C library implementation of that protocol. The compositor can be a standalone display server running on Linux kernel modesetting and evdev input devices, an X application, or a wayland client itself. The clients can be traditional applications, X servers (rootless or fullscreen) or other display servers.
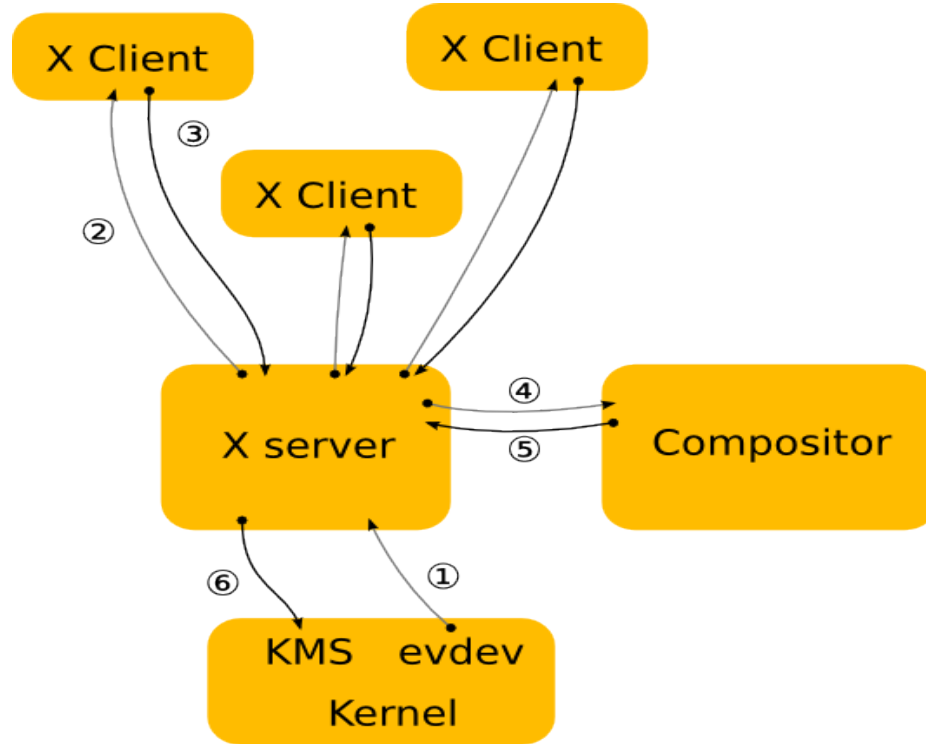
License: MIT

Language: C

Source: git://anongit.freedesktop.org/wayland/wayland

Project Page: http://wayland.freedesktop.org/
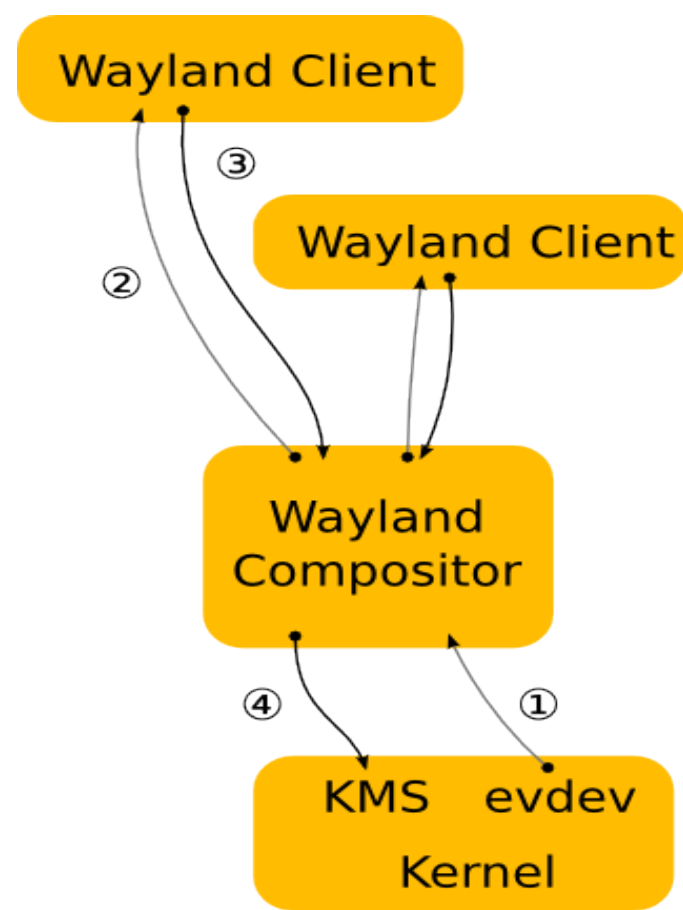
Wayland: Traditional Architecture

Wayland: Architecture

Wayland New Developments

- Remote Display Support
  - Streaming GL
  - Sharing content between ECU's
  - Display IVI content instrument panel
- Screen and Input Transformations
  - Enable common off the shelf monitors to be installed in landscape orientation
- GENIVI Layer Management API Support
  - Implementing an IVI shell for enabling industry innovation around standardizing layer management control APIs

Questions

JRE **INTEL LINUX WIRELESS** GUPNP OPENSTACK POWLRICP LESSWATTS KVM POKY
NNMAN YOCTO INTEL LINUX GRAPHICS MEEGO
SIMPLE FIRMWARE INTERFACE (SFI) ENTERPRISE SECURITY INFRASTRUCTURE

INTEL OPEN SOURCE
TECHNOLOGY CENTER