



# App FW – Working session

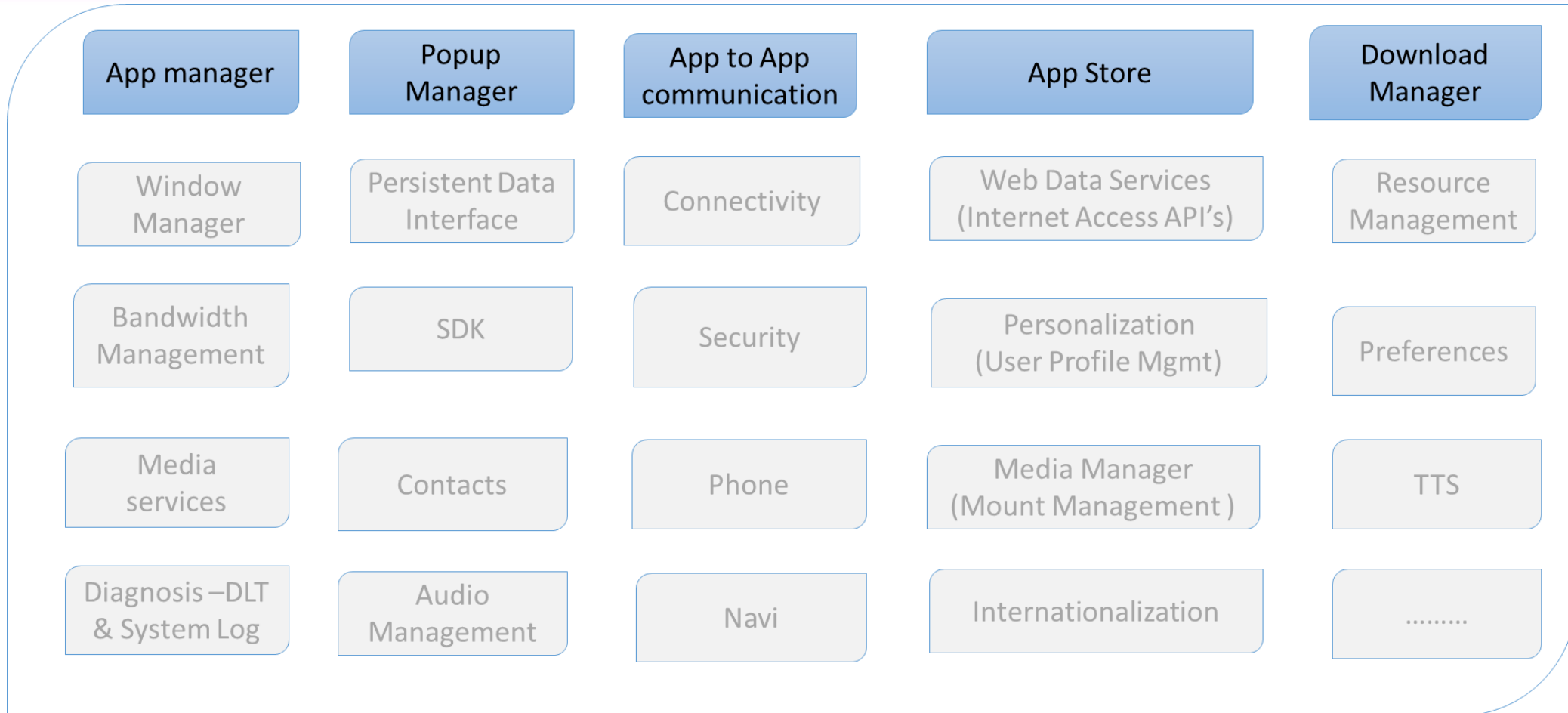
28-04-2016 | GENIVI Members

Gunnar Anderson, Gururaja N  
SAT Lead, Lead Architect  
Volvo, Bosch

# Agenda

- App FW – Status
- App manager comparative study
  - Tizen, Apertis and Qt Automotive
- Challenges in defining Unified API's – Johan Thelin, Pelagicore
- Security - Philip Withnall, Collabora
- Open discussion

# App FW – Functional Area



\*Note : This is not a representation of any component. Only depicting the functional Areas

# App FW Status



28-April-2016

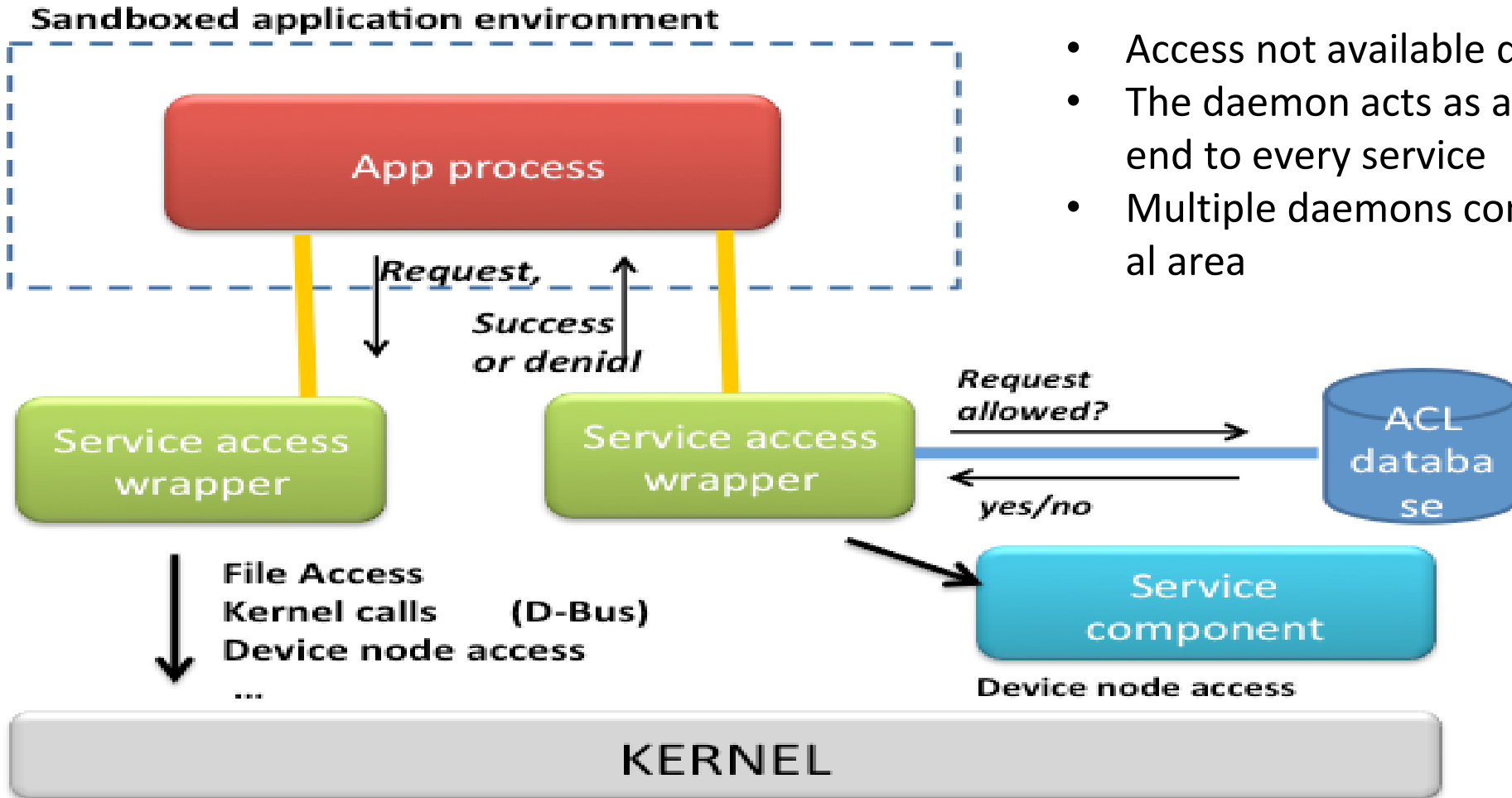
- Application framework consists of a software framework used by software developers to implement the standard structure of an application
- GENIVI Reference Architecture works with two different types of Applications
  - Managed Application
  - Native Application
- Application can be a UI based or without a UI

# Native Application

- Developed under OEM supervision or otherwise vetted
- “Trusted Application” principle for user data separation
- Security : Follow principle of least possible privilege to reduce damage in case of exploit
- Applications by nature are part of System SW update

- Downloadable apps
- Third party developed Apps
- Provide full sandboxing
- May reuse some already existing foreign frameworks
- Extended/alternate lifecycle strategy
- Extended/alternate user management strategy
- Extended/alternate persistence management strategy

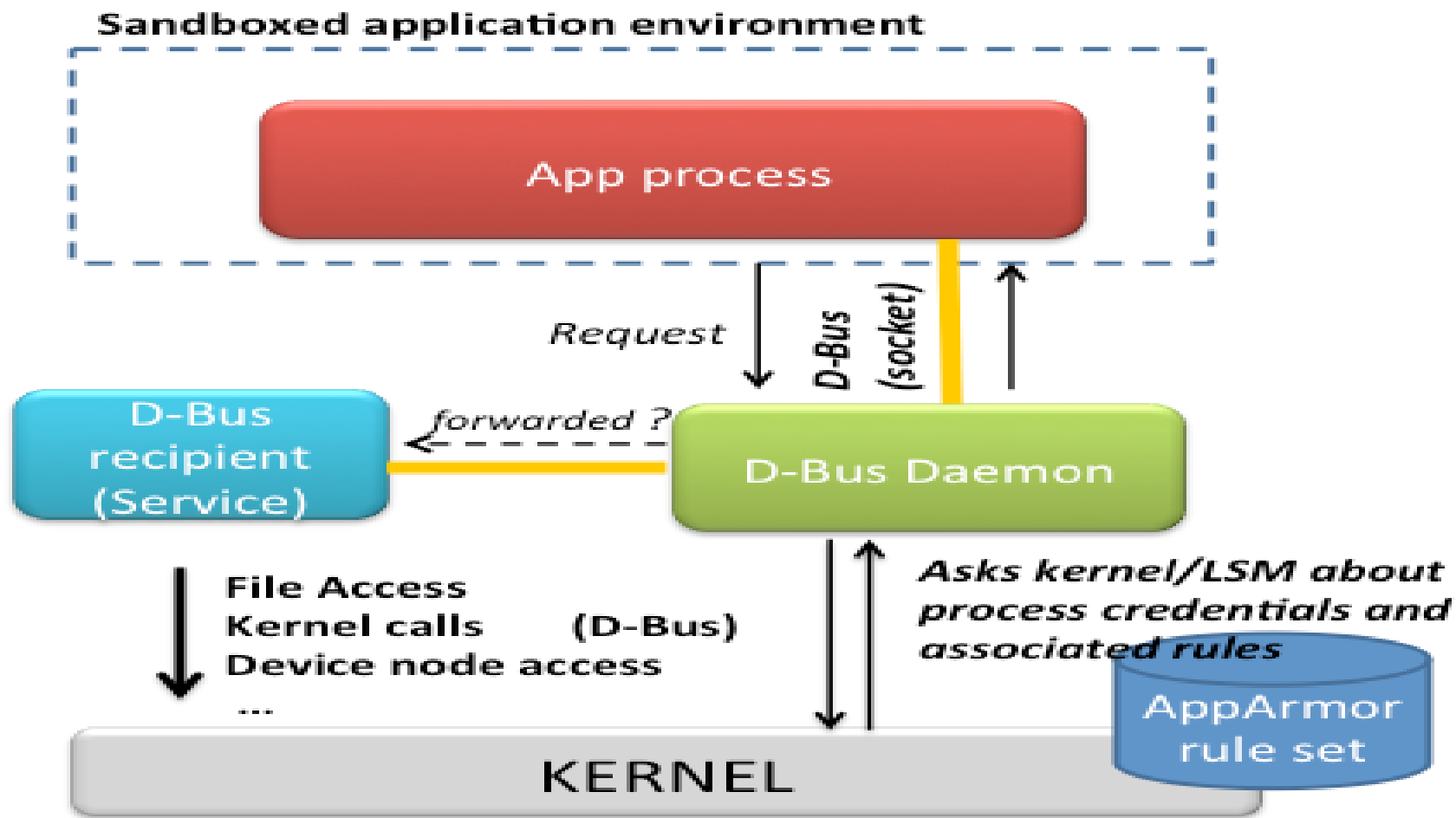
# Access Control mechanism



- Access not available directly by IPC
- The daemon acts as an access control front end to every service
- Multiple daemons controlling each functional area



# Access Control mechanism



# Agenda

- App FW – Status
- **App manager comparative study**
  - Tizen, Apertis and Qt Automotive
- Challenges in defining Unified API's – Johan Thelin, Pelagicore
- Security - Philip Withnall, Collabora
- Open discussion

# App Manager Comparative study

Tizen	Apertis	Qt Automotive	Comments
getCurrentApplication	getCurrentActiveApp	Defined by the System UI. Can support multiple Apps at once	<b>Similar</b>
kill	Apps are killed based on their states by app manager.	ApplicationManager::stopApplication(string id, bool forceKill).	Qt : Only available to System UI and via System DBus. Apps themselves can only launch other apps through mime-type <b>Similar</b>
Launch	LaunchNewApp ()	ApplicationManager::startApplication ApplicationManager::openUr	

# App Manager Comparative study

Tizen	Apertis	Qt Automotive	Comments
LaunchApp Control	openURI – launch an app based on mime type that it handles	Applications uses standard Qt interfaces, i.e. they are independent of Application Manager	Similar
findappcontrol	ApplaunchDatabase - Returns a list of apps and the MIMEs that they can handle	The ApplicationManager singlet on presents a model with the metadata regarding all installed apps and information regarding if the application is running or not.	Similar
getappcontext	----	----	Specific

# App Manager Comparative study

Tizen	Apertis	Qt Automotive	Comments
getAppsinfo get list of installed apps	Api's available for System UI to get such info	Available to System UI	Similar
getAppinfo Gets application information for a specified application ID.	getApplicationInfo	Available to System UI	Similar
getAppSharedURI Gets URI of read-only shared directory of application for a specified application ID.	----	----	Specific

# App Manager Comparative study

Tizen	Apertis	Qt Automotive	Comments
<b>getAppMetaData</b> Gets application meta data array for a specified application ID.	<b>getappinfo</b>	Available to System UI	<b>Similar</b>
<b>addAppInfoEventListener.</b> Change to list of installed apps	<b>signal:</b> <b>AppLaunchDatabaseUpdate</b>	Available through the object returned from <b>ApplicationManager::get.</b> Available to System UI	<b>Similar</b>
<b>removeAppInfoEventListener</b>	<b>available</b>	Yes, through Qt signals/slots.	<b>Similar</b>

# App Manager Comparative study

Tizen	Apertis	Qt Automotive	Comments
----	<p>GetGlobalSearchApps</p> <p>This API gives a list of apps that are supporting global search</p>	<p>This would be implemented as a tag in the manifest, e.g. another capability</p>	<b>Specific</b>
----	<p>Registermyapp</p> <p>The registration is needed as app manager is a dbus service</p>	<p>The ApplicationManager itself is not tied to a specific appstore implementation - as long as the package can be downloaded via HTTP, HTTPS or FTP</p>	<b>Specific</b>
removeAppInfoEventListener	available	Yes, through Qt signals/slots.	<b>Similar</b>

## Summary

- Feature wise there are similarity among all the three
- There are features influenced by the Architecture
- When it comes to API and parameter matching, there is quite a lot of differences

## Next steps

- Understand the challenges in deriving a common ground



# Agenda

- App FW – Status
- App manager comparative study
  - Tizen, Apertis and Qt Automotive
- Challenges in defining Unified API's – Johan Thelin, Pelagicore
- Security - Philip Withnall, Collabora
- Open discussion

Thank you