GENIVI's Role in & Roadmap for Driving Apps and HMI Integrations

HMI & Application Framework  EG
10.10.2012 / 10:30 – 11:30

Ashwin Khodiyar / Prashanth Halady
EG Lead / EG Lead Architect
Jaguar Land Rover / Robert Bosch
Agenda

• Motivation
• Scope & Goals
• Journey until now
• Overview of focus areas identified
• Brief outlook
• Addressing FAQs
Motivation

- Increase reuse and reduce costs for several products/model years by offering common non-differentiating services / behaviours for HMI
- Enable 3rd party application integration
- Support functional extension during life of product through upgradeable applications
The field of activity of the HMI- Application Framework EG is to define:

1. An HMI Framework
2. An Application Framework

To achieve the following goals:

1. Identify and specify platform requirements from HMI perspective
2. Identify and offer non-differentiating common services required by HMI applications
3. Enable upgrades & downloadable applications in the platform
The work of the HMI-Application Framework EG will:

- Ease up integration -> speed up development
- Provide an open and flexible architecture concept without restrictions to Design and Tooling.
- Enable 3rd parties to easily integrate applications.
- Offers a contact to other EGs to discuss HMI topics.
<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sep 2011</td>
<td>Stakeholder needs identification</td>
</tr>
<tr>
<td>Oct 2011</td>
<td>Discussion on scope @ AMM San Jose</td>
</tr>
<tr>
<td>Oct 2011</td>
<td>Outcome of AMM Setup of BoF</td>
</tr>
<tr>
<td>Nov 2011</td>
<td>Wiki page setup to collect input</td>
</tr>
<tr>
<td>Feb 2012</td>
<td>Publication of White paper outlining scope of HMI Framework</td>
</tr>
<tr>
<td>Mar 2012</td>
<td>BoF transformed to EG by GENIVI BOD</td>
</tr>
<tr>
<td>May 2012</td>
<td>HMI &amp; Application EG outlined scope of the EG in the Paris AMM</td>
</tr>
<tr>
<td>July 2012</td>
<td>Focus areas/projects identified in F2F meeting</td>
</tr>
</tbody>
</table>
• CE World concepts (iOS, Android…) drive end-user expectations with respect to flexibility of infotainment systems in function and personalization.

• This is reflected in a demand for an architecture change in modern IVI systems:
• Explanation of terms:
  – An HMI Application consists of the following Layers:
    • Graphical representation & rendering.
    • Graphical logic - Screen flow, etc.
    • Functional logic of the application, e.g. Media Player.
    • Connection to the middleware.

Note: This is an example for better understanding, not a binding architecture!
HMI & Application Framework EG Development Phases

**Need**
- Based on feature or technology to be supported
- Produces a scope description

**Vehicle Level**
- Produces vehicle-level actors, use cases and requirements

**Software Platform**
- Produces component architecture and software platform level requirements

**Placeholder Component**

**Abstract Component**
- Produces Component Specification (including the API and behavior)

**Specific Component**
- Produces component implementation

**Source Code**

---

19-Oct-12

GENIVI is a registered trademark of the GENIVI Alliance in the USA and other countries
Copyright © GENIVI Alliance 2012
HMI & Application Framework EG
Collaboration with other EGs

• HMI-Application Framework EG acts like a client:
  – Provide requirements and Use Cases from HMI perspective
  – Concept review from HMI perspective.

• Collaboration areas with:
  – Sys. Infrastructure EG (User management, Communication)
  – Media & Graphics (Layer Manager, …)
  – LBS (Speech, …)
  – Automotive EG (Software Manager, ..)
  – Korea EG & Networking EG (HTML5,..)
Current Projects

- Pop-up Management  
  Project Lead – Gururaja N - Bosch

- Plug-in Architecture  
  Project Lead – Eric Terree – Magneti Marelli

- Generic Speech Interface  
  Project Lead – David Kaempf - Continental

- HTML 5  
  Project Lead – Mark Rivers - Mentor
<table>
<thead>
<tr>
<th>Category</th>
<th>Project Lead</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Management</td>
<td>Ashwin Khodiyar – JLR</td>
</tr>
<tr>
<td>Translations &amp;</td>
<td>Gernot Resinger – Fujitsu</td>
</tr>
<tr>
<td>Internationalization</td>
<td></td>
</tr>
<tr>
<td>Communication</td>
<td>Gernot, Fujitsu &amp; Mark, Mentor</td>
</tr>
<tr>
<td>Registry</td>
<td>Pierre Sigrist – Visteon</td>
</tr>
</tbody>
</table>
**Description:**

Manage pop-ups from different sources and their priorities. The pop-up manager is responsible to show the pop-ups from all applications of the system according to current priorities. Therefore each application has to register to the pop-up manager and has to provide information about its pop-ups.

**Initial Scope:**

Common interface pop-up manager and applications

- Subscribing mechanism
- Define priority levels
- Define pop-up description format for different application sources

**Responsible:** Gururaja N, Bosch
HMI & Application Framework EG
Application plug-in architecture

Description:
Exchange major HMI application components during IVI system lifetime and also support downloadable applications. The HMI architecture proposal has to consider a plug-in concept to increase flexibility.

Initial Scope:
Identify major requirements and overlap of requirements with other EGs like Automotive EG (Software Management).
Areas of focus – Application life cycle management, Resource management, Security management, Communication mechanisms & Priority levels.

Responsible: Éric Terrée, Magneti Marelli
Description:

A new application can assume a standard interface to speech and voice recognition systems.

Initial Scope:

Identify requirements towards a unified Interface for Speech/voice components in the system.
Vendor agnostic abstraction layer
Integration of Voice recognizer & TTS engines

Responsible: David Kämpf, Continental
Description:
The HMI has to adapt to the actual user settings/profiles. To do so, it is necessary to inform the HMI about the actual user and to provide the appropriate profiles/data. These are the essential tasks for a user management.

Initial Scope:
Review of existing requirements and solutions from HMI Perspective
Proposal of additional requirements to SI EG

Responsible: Ashwin Khodiyar, Jaguar Land Rover
Description:
IVI system supports communication mechanism in between HMI modules and to the middleware tailored to the HMI needs.

Initial Scope:
Elaborate Requirements and Use cases for communication framework
Proposal for Data Binding concepts - A special focus should be on usecases like huge Lists and fast IPC mechanism

Responsible: Gernot Reisinger, Fujitsu & Mark Rivers, Mentor
Description:
Define a standardized way, how HTML 5 based applications are integrated (e.g., Data Binding) and installed in a system.

Initial Scope:
Common interfaces for HTML5 applications to other Genivi automotive components
Extend Web Vehicle API by Korean EG with interfaces needed by HMI FW, e.g. Popup Manager

Responsible: Mark Rivers, Mentor
Description:
Support / provide infrastructure to upgradeable / downloadable applications in terms of launch,

Initial Scope:
Define a manifest file/ contents, that contains all information, that is needed for installation and Application management (dependencies, needed resources (Audio/ Video subsystem), Home screen icon, Security restrictions, Last mode behavior..)

Interface with other EGs for interfaces/requirements for application launching and life cycle management

Responsible: Pierre Sigrist, Visteon
Use-Case:
Adaptive-HMI – Dynamically adapt HMI content to current driver state

Description:
The driver state manager keeps track about the cognitive workload and level of distraction of the driver. This information is available to all applications. The applications, define how to react on a propagated workload state, e.g. reduce HMI functionality or delay incoming external events like email.

Initial Scope:
- Define a common set of basic workload states
- Define a common interface to a Driver State Manager
- Reference implementation to support basic legal requirements
Components

Level 3 – Specific
Compliance based on source code

Level 2 – Abstract
Compliance based on interfaces
- Pop-up Manager

Level 1 – Placeholder
Compliance based on requirements
- App. Plug-In Architecture
- HTML5 Interface
- Speech Service

FOTON
04/2013
- App. Plug-In Architecture
- HTML5 Interface

GEMINI
10/2013
- Translation & Internationalization
- Driver Work-load Manager
- Distributed HMI Manager
1. What is inside the scope of the EG and what is outside especially as the topic of HMI is a differentiating factor for the OEM?

2. How do you differentiate between upgradeable and downloadable applications?
3. How is the EG playing a role towards integration of the HMI of a native and a downloadable application?

4. You have mentioned about downloadable apps. Which app. development technology will be supported?
THANK YOU