PLINTA - PLattform för säker INTegration av Användarfunktioner i bil.
Public Final Report

Vinnova Project within FFI - Fordonstrategisk Forskning och Innovation
Vinnova reference, Diarienr: 2012-01703

The PLINTA project was executed between 2013.01.01 and 2015.01.30
Project Partners: Semcon, Pelagicore, HiQ and Volvo Cars

Confidentiality
The content of this report is open for the public. It is the intended starting point for people interested in the PLINTA project.

Date 2015-01-30

FFI

Table of Contents

FFI in short

FFI is a partnership between the Swedish government and automotive industry for joint funding of research, innovation and development concentrating on Climate & Environment and Safety. FFI has R&D activities worth approx. €100 million per year, of which half is governmental funding. The background to the investment is that development within road transportation and Swedish automotive industry has big impact for growth. FFI will contribute to the following main goals: Reducing the environmental impact of transport, reducing the number killed and injured in traffic and Strengthening international competitiveness. Currently there are five collaboration programs: **Vehicle Development, Transport Efficiency, Vehicle and Traffic Safety, Energy & Environment and Sustainable Production Technology.**
For more information: www.vinnova.se/ffi

1 Executive summary

PLINTA - PLattform för säker INTegration av Användarfunktioner i bil - is an innovation project financed under the VINNOVA program - Fordonstrategisk Forskning och Innovation” (FFI). PLINTA is a project in the FFI Elektronik, mjukvara och kommunikation portfolio. The project partners are Semcon, Pelagicore, HiQ and Volvo Car Corporation.

The purpose of the PLINTA project is to address the problems related to use of mobile devices while driving and increase the competitive edge of Swedish companies within software based services for automotive. The suggested solution is to introduce an infotainment environment powerful and adaptive enough to be able to continuously coexist with mobile devices and a the same time maintain the position as the prefered user interface for the driver. The work in the project has resulted in:

- A flexible and open Infotainment Head Unit (IHU) system with a big touch screen has been built and integrated in a Volvo XC60 car. The system is based on a platform that enables development of applications that can easily be downloaded and installed. The applications can access and control vehicle specific functionality.
- Increased system security has been achieved by using a secure gateway concept between the vehicle’s core functionality and the consumer oriented infotainment functions. The concept can also be used to extend the vehicle with new hardware that can be used by applications in an easy a flexible way.
- An adaptable HMI has been developed. The HMI can handle different touch screen resolutions and different vehicle configurations without software changes. This is an initial step in the direction of lowering dependency between IHU development and vehicle programs. Minimizing driver distraction has been considered and the HMI can be tailored to achieve such requirements. Face tracking as a mean for driver distraction monitoring has been prototyped.
- An Infotainment Head Unit (IHU) remote control and remote driver identification application for an iPhone have been developed.
- The reference platform is cloud connected enabling personalization and prediction of driver intent based on combination of data from different servers with personal information. E.g., automatic extraction of event location from calendar and feeding the vehicle’s navigation system.
- The combination of a flexible development environment for the IHU and a secure connection to the vehicle’s core functionality enables easy development of new functions such as applications for ECO-driving.

It is widespread conception that many of the above mentioned enablers are already industrialized by the automotive industry. Actually this is far from the truth; adding functionality or new user experiences into todays vehicle infotainment systems is in reality a time consuming and costly task. Major barriers are

- Current agreements and commercial arrangement in Automotive industry are not developed for open platforms.
- The eco systems are not there yet. Too many platforms and no strong driving force with a clear business model. A consolidation of the many short lived IHU Programs into a few platforms with trusted long term evolution independent of vehicle program cycles is required.

To overcome these barriers, the hope is that the work in PLINTA can trigger and inspire the Swedish automotive industry to seek out the opportunities for bringing these ideas to life. Hence, an important part of the PLINTA project has been to demonstrate and show the possibilities of an open and flexible platform.

2 Background

The innovation options within automotive seem endless these days. Cars become cloud connected, bring multiple displays, myriads of sensors and cars even become autonomous. To leverage these options and deliver great User Experiences the OEM must shorten development cycle, lower the threshold for implementing new ideas and become adaptive to User Feedback.

The reference for connected user experiences are the continuously updated smartphones and it will always be tempting to use these in the car despite the fact that usage of smartphones distracts the driver. Apple Car Play and Google projection mode are initiatives that strive to make the smartphone the center in the infotainment system but still using the buttons and screens in the car for input and output to make usage safer and more intuitive.

It takes 3 years to develop a car that can live for 15-20 years and so far the industry has mainly delivered updates focused on handling bugs. Competition among Car OEMs is increasingly being focused on offerings in the digital domain and the dominant 17-inch touch screen in the Tesla has made the new in-car digital paradigm visible to the world.

To position the car as a safe and secure enabler of connected digital innovation new software architectures and development methods are required.

To offer more digital innovation the car industry has to lower the cost for infotainment development. A logic path is to share the development of non-differentiating software across the industry and concentrate resources on the the elements that are differentiating and seen as truly innovative. This calls for use of open source software components, cross industry standardization and strong strong development orchestration capabilities.

3 Objective

PLINTA aims to create concrete reference implementations and demonstrators of a modern infotainment platform based on an open and secure architecture. The PLINTA project is executed using agile development methods.

The demonstrators serve both as proof of concept and as fundament for promoting the enablers in the platform and thereby establish buy-in and momentum for taking these further towards industrialization.

The platform should lower the threshold for creation of any new User Experience and thereby contribute to strengthen Sweden on the international automotive market. The platform will contribute to the development of safer cars.

Key development objectives:

- Linux based infotainment platform based on open source components
- Reference HMI enabling safe usage and easy adaption of new features
- Explore options for smartphone symbiosis
- Concept for secure and managed access to the vehicle networks
- Explore options of innovation enabled by cloud connection
- Explore methods of project execution between the partners

4 Project Realization

The project is realized as a collaboration project between Volvo Car Corporation, HiQ, Semcon and Pelagicore. In accordance to the contractual undertakings for the PLINTA project each party takes responsibility for the implementation and integration of work items into a PLINTA reference system. Applied high level roles and responsibilities for each party is illustrated in figure 1.

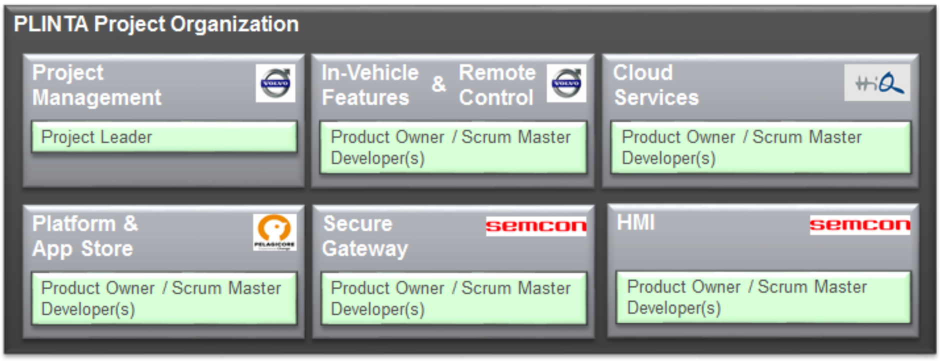

Figure 1: Project roles and responsibilities split onto PLINTA partners.

The execution of the project is done in accordance to the schedule illustrated in figure 2.

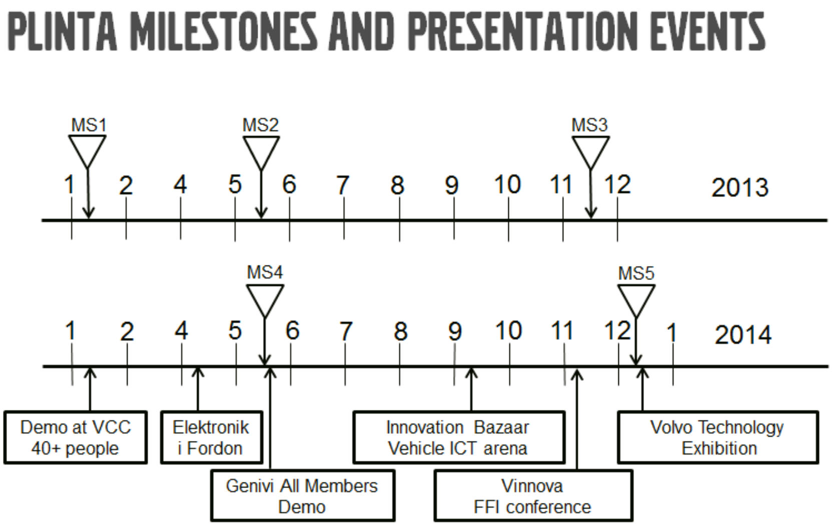

Figure 2: Project schedule.

The project is executed with an agile mindset based upon the agile manifesto:

> *"**Individuals and interactions** over processes and tools*
> ***Working software** over comprehensive documentation*
> ***Customer collaboration** over contract negotiation*
> ***Responding to change** over following a plan"*

http://agilemanifesto.org/

Other agile development principles applied within the project are illustrated in figure 3. However, due to the fact that there has been a dedicated scope for each PLINTA partner and this scope has been defined and agreed upon before project start, it has been difficult to fully embrace agile development methods and principles.

A project wiki hosted by Semcon is used for project documentation and administration of backlog.

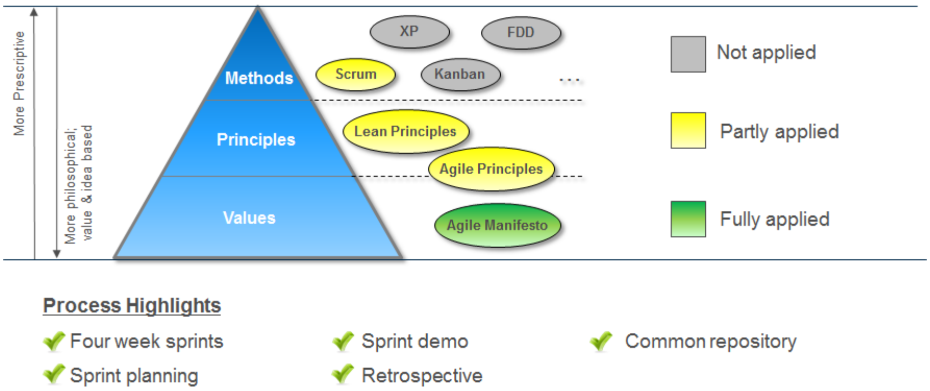

Figure 3: Agile principles applied within the project.

5 Results and Deliverables

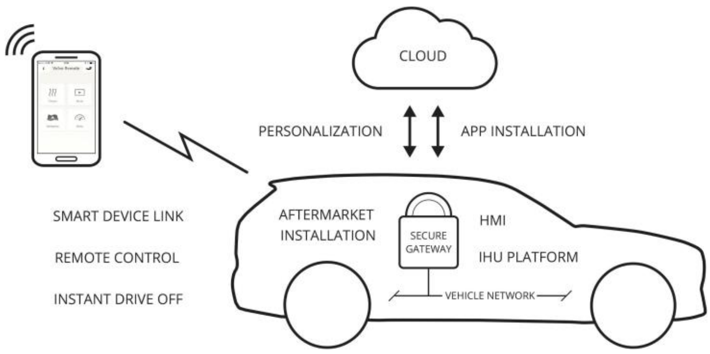

Figure 4 shows the PLINTA system and use case relations.

Figure 4: The PLINTA system overview and use cases

5.1 IHU Platform

The Infotainment Head Unit (IHU) platform in the PLINTA demonstrator was built on-top of a Yocto based platform supplied by Pelagicore. Through the inclusion of a IVI meta layer the platform included the components needed to be GENIVI compliant. On top of this common base service layer, the Pelagicore Application Manager (AM) was used to host the system user interface, manage application installations and to monitor the life-cycle of applications in the system. A high level overview of the layered architecture applied in the PLINTA demonstrator is given in figure 5. Some of the conceptual foundations that the platform is built upon is described below:

- **GENIVI Alliance -** A non-profit consortium whose goal is to establish a globally competitive, Linux-based operating system, middleware and platform for the automotive in-vehicle infotainment industry.
- **Qt -** A cross-platform application and UI framework for developers using C++ or QML, a CSS & JavaScript like language. Qt, Qt Quick and the supporting tools are developed as an open source project.
- **Yocto -** The Yocto Project is an open source project governed by Linux Foundation. The goal with the project is to produce tools and processes that will enable the creation of Linux distributions for embedded software that are independent of the underlying architecture of the embedded software itself as well as the underlying Board Support Layer.

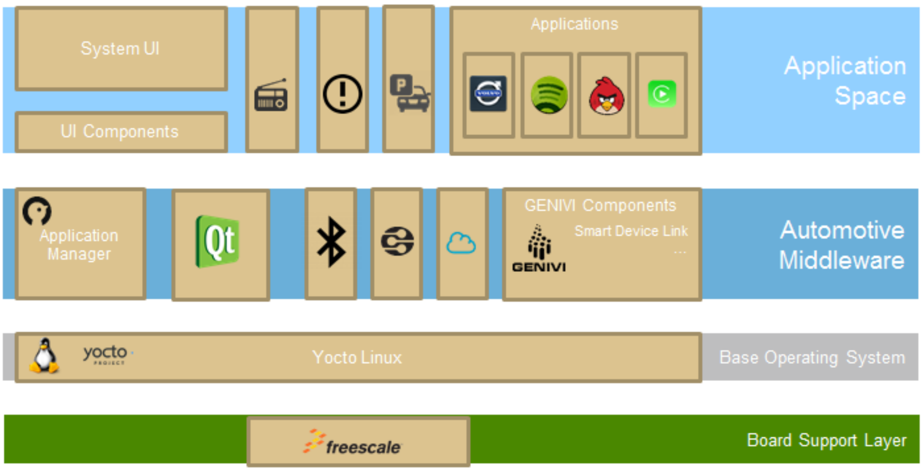

Figure 5: PLINTA reference system overview.

5.2 HMI

A frontend implementation for the project was created as a brand neutral user interface to demo functionality. The foundation of the interface lies on providing the user with quick and effortless access to the most important features and information, yet allowing in-depth access to all present categories and individual applications, permitting the user to spend more time and effort on primary and secondary tasks and less effort on lesser critical ones.

The de-cluttered and lucid information presentation in a tile based grid, see figure 6, prevents extended glance times and unnecessary interaction to accomplish common tasks. This keeps driver distraction at a minimum during usage. The interface therefore provides a safer and more proficient environment than current solutions for nomadic devices, by standardizing the visual presentation and interaction paradigms for all connected devices and applications.

Figure 6: PLINTA user interface, with its tile based grid.

Most regions are introducing guidelines and regulations to lessen driver workload and distraction consequence, by controlling and limiting the driver interaction with in-vehicle infotainment systems. The HMI must be able to support both OEM and international and national HMI Design Guidelines (e.g. ESoP, NHTSA, AAM, JAMA). The NHTSA guideline only emphasizes two vehicle modes: driving and standing still. This, in itself, provides complexities for the app developers. But the latest standards as defined within the Genivi project defines five different distraction levels or states.

A separate report[1] presents a HMI safety framework for how this complexity can be managed in order allow app developers to focus on their application, without being burdened by automotive environment and still be able to create innovative services that adhere to safety standard, as well as OEM guidelines. A second benefit is that different apps would behave similarly without requiring coordination between app developers, resulting in a more uniform user experience.A well behaving app, developed according to this framework, would be informed about the distraction level as well as changes to it and use this information to adapt the HMI in terms of e.g. animations, images, sounds, input modes, font sizes, button sizes and text lengths.

As a concrete example of the effect of this, an app could automatically present larger text and buttons as well as removing all scrolling text effects, as the traffic situation complexity increases (e.g. higher speed, driving at night, other vehicles).

5.3 Secure Gateway

A very important use case seen in modern vehicles that has been explored in the PLINTA project is the ability to access the vehicle's sensors and actuator from applications running in the infotainment environment. This includes access from both applications running in the native equipment – pre-installed or downloadable – as well as applications running in mobile devices connected to the vehicle for example via a wifi hotspot. Such connections require a high level of security to prevent that malicious or malfunctioning applications cause damage or tampers with the vehicle in unwanted ways. Failing to secure this could, in worst case, make the vehicle malfunction in ways that are potentially dangerous.

The secure gateway (SG) is a concept that addresses the needs for applications to access the vehicle network and, at the same time, providing a protection for malicious or malfunctioning applications. The core of the concept is a network consisting of three layers – a network layer, a messaging layer and a service layer. The layers help to enforce "separation of concerns" and increases security. A schematic picture is show below.

5.3.1 Network Layer

The base for the Secure Gateway is a TCP/IP-network running on wifi or Ethernet or a combination of both. Standard protocols for encryption such as TLS/SSL and certificates, easy configuration as defined in RFC 3927 and service discovery using mDNS (RFC 6762) have been used. These protocols are grouped together to form the lowest layer in the Secure Gateway concept – the network layer. The layer is schematically presented in the figure 7 below (blue color). Communication with vehicles networks, such as CAN or Flexray is an important part of the concept and is handled using network gateways (not to be confused by the Secure Gateway concept itself). In the network gateway the IP-network and the vehicle network meet and can communicate with each other, using the messaging and service layers in the Secure Gateway. Note that there is no direct translation between IP packets and vehicle network data format since the communication needs to be secured.

[1] Available in the Technical Report. Based on a concept developed in another Vinnova sponsored project - SICS.

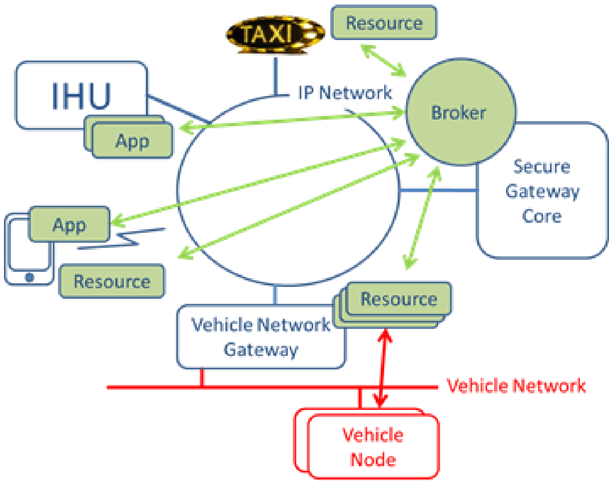

Figure 7: SG with network layer (blue) and messaging layer (green)

5.3.2 Messaging Layer

The communication in the Secure Gateway is based on a topic based publish- subscribe pattern where messages are sent from a publisher to one or more subscribers. But instead of the publishers knowing which subscribers to send the messages to, all communication is done via an entity called a broker. The subscribers register themselves with the broker and list the topics that they are interested in. All messages from publishers are tagged with a topic and then sent to the broker. Based on this information, the broker can determine which subscribers that shall receive the message and forward the message accordingly. The Secure Gateway uses MQTT as the messaging protocol and this forms the messaging layer that resides on top of the network layer. All messages are encrypted and a gatekeeper in the broker uses access control lists (ACL) to determine access rights.

5.3.3 Service Layer

The top layer in the Secure Gateway is the service layer. In this layer, there are two important actors – applications and resources. In short, a resource is a software component that represents a device, for example a media player or a temperature sensor. The resource provides API:s using the messaging layer (MQTT). Applications, installed for example in the IHU, can then use these API:s to read data or control actuators. To access devices on the vehicle network, resources on the vehicle network gateway are used. Certificates are used by the gatekeeper in the messaging layer to authenticate both resources and applications.

5.4 Application Installation

Options for user management and how to provide a reusable framework for user profiles, including private data such as account details and settings, have been explored within the project. It was found that the concept of sandboxing can provide clear user separation without affecting the applications. The configuration of the sandbox is done to correctly reflect user profile information. This makes it possible to combine common automotive applications such as Tuner and Navigation that provide seamless operation between users, while allowing for 3rd party applications on the platform without risking information leaks between user profiles. A sandbox is implemented by executing the software in a restricted operating system environment.

5.5 Personalization

During the spring 2014 the PLINTA project started a cooperation with Ericsson and developed functionality, based on the Ericsson Multiservice Delivery Platform (MSDP), that supported the storing and downloading of personal settings (personalization) for the vehicle. Integrating the cloud enables whole new ranges of functionality.

The following services where integrated or implemented:

- Streaming services, i.e. music apps like Rdio, where your personal playlists and authentication tokens for the services are downloaded.
- Personalized car climate settings.
- Google Calendar - keeping track of your appointments
- Google Maps - where a Google Calendar event can set your next destination, and Google Maps keeps track of where you are, the distance to your destination and when you get there.

5.6 Smart Device Link

Smart Device Link is an open source competitor to similar technologies like Mirrorlink and Apple Car play. It enables control of smartphone apps from a car HMI using voice and touch physical buttons. The work involved understanding the source code, designing an interface layer so that it could be used in an HMI based on QT instead of a HTML5 based one and writing the actual code.

An implementation on the PLINTA platform was demonstrated at the GENIVI All Member Meeting in Gothenburg. The strategy was to release the developed code for smart device link back to the Genivi project. However in meantime Genivi project had released newer source code which made the VCC contribution obsolete.

The open source project responsible for Smart Device Link has continued to release newer versions of the source code that improve the functionality. However it has limited support by automotive OEMs. This is partly due to bad quality of the source code and poor documentation but also because Apple and Google have released competing projects that offer similar functionality as Smart Device Link.

5.7 Remote Control

To test the developed platform and explore aspects of smartphone symbiosis use cases identified during VCC user clinics in China was identified. These cases was chosen since it would secure a brough audience for the PLINTA project.

Back seat hosting enabling a back seat passenger to control climate, media and navigation functionalities within the car from a smartphone or a pad.

A remote control application was created for iPhone/iPad. When the passenger enters the car the iPhone connects to the car’s WiFi hotspot, the remote control will automatically find the message broker with in the car, and try to connect, (Connection handling can be seen under the section Secure gateway). After a successful connection, the application will check if the resources it can control are present. If they are present they can be controlled.

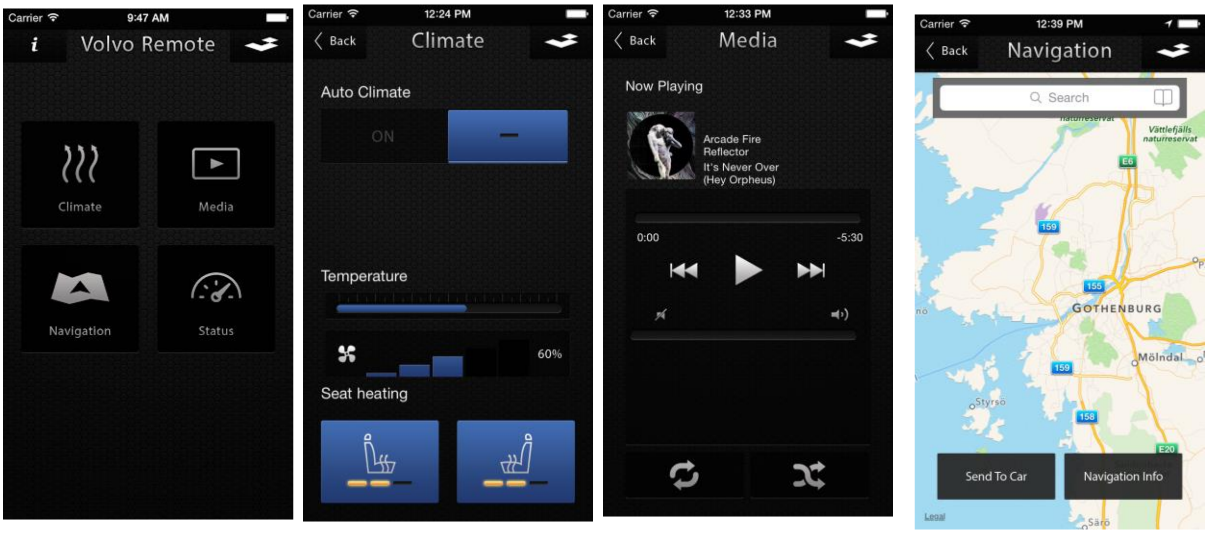

Figure 8: Graphical user interface for remote control.

The remote control is made in iOS, and consist of four different screens as illustrated in figure 8:

- **Screen 1**, adaptive home screen for remote control application
- **Screen 2**, the climate view, give the user the possibility to control some climate settings from on the rear climate section.
- **Screen 3**, the media view, this view act as a remote control for the infotainment media system . In the Plinta project we implemented only some basic functionality, but it will be possible to extent this quite heavily.
- **Screen 4**, the navigation view, inputting an address to a built-in navigation system can be very cumbersome and leads to driver distraction. Using the capabilities of Google Maps or iOS to search for an location enable easier search for the driver and support from a passenger.

Instant Drive Off is a use case identified in China. The idea is that the driver enters the car and all personal settings and destination are already known by the car.

Today the car user is identified via the car key. The range on the wireless transmitter in the car key is set to limit power consumption and the cost is quite high.

Bluetooth Low Energy (BTLE) is a newer sub-standard within Bluetooth 4.0. It can be used in a device to support keyless access. The goal was to identify the User, startup the infotainment system, and preload the user settings.

In PLINTA an iPhone is used as Bluetooth LE device with the intention to replace the physical key. The iPhone is constantly advertising its presence and after a successful connection, the server will start the authentication process. After a successful identification, the infotainment system will be informed about the approaching user and preload all settings from the cloud. The system will be completely operational when the user opens the door - ready for instant drive off.

Google accounts for three users were created, containing different calendar events on different locations. After identification the driver's personal credentials are fetched from the cloud, and used to login to the Google account. The Google API is used to fetch the calendar, and check for calendar event with location information. The location information, if available, is provided to the navigation system, and one or more route is planned.

5.8 Face tracking

A face tracking system was added to the PLINTA platform.The intent was to enable the infotainment system to be aware of the driver's attention level and use it to take appropriate action like shutting down distracting applications. The face tracking was implemented using a Web camera and the OpenCV library.

OpenCV (Open Source Computer Vision Library) is a library of programming methods and algorithms focused on real time computer vision. Mainly focused on real-time image processing, object detection, face recognition algorithms, and image manipulation.

Face detection is an important part of face recognition as the first step of automatic face recognition. Thus face and eye detection was achieved using a detector and a trainer, from the OpenCv library, based on Haar feature-based cascade classifiers, an approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

While the system is in use, it will search for faces and eyes in the image. If the eyes are closed or the face is out of range, more than 2.5 seconds, the system sends a warning. The system is able to learn the faces, by collecting images with the driver, and using EigenFaces algorithm to recognise faces. Feature that can be used for driver profiling or for security measures. Face recognition can also be used to identify the driver and thereby personalize the car.

5.9 Aftermarket Installation

Using the concepts of the Secure Gateway, it is possible to develop new aftermarket solutions. In the project this is demonstrated with a taxi sign. A taxi sign is mounted on the vehicle and connects automatically to the Secure Gateway. A taximeter application is then downloaded and installed in the IHU. Using the messaging protocols (MQTT), the application can find and use the API that the taxi sign provides. This (rather naïve) example demonstrates a simple, but effective way of extending a system with new functionality that can be used for more elaborate functions.

5.10 Delivery to FFI-goals

The FFI sub-program goals to increase the knowledge and reduce development and industrialization time are addressed in the following ways:

- Electrical Architecture for embedded and connected systems
  - The PLINTA reference system can serve as an open and flexible platform to be used as a basis for further development of In-Vehicle Infotainment systems. The PLINTA platform and the agile ways of working in the project have influenced the VCC strategy for infotainment development significantly.
  - The modular and open environments used in the secure gateway as well as the IHU platform enable the leveraging of external development to reduce time to market.
  - Connection to cloud and cloud computing.
  - Adding flexibility into the architecture to be able to add new functions
- Technologies for "green, safe and connected functions"
  - Proof of concept that face tracking can be used as element in a driver distraction system.
  - A concept for how application development that takes driver distraction into consideration can be simplified
- Digital user interfaces
  - A reference user interface preventing driver distraction and promoting standardization between connected devices

    - A framework for allowing third parties to create modern and consistent HMI solutions while adhering to OEM and safety standards
- Growth in Sweden
    - By using open and flexible platforms, the PLINTA project shows how to increase the speed and volume of innovation. This lowers implementation costs and allows more developers to work autonomously on the real platform.

6 Dissemination and publications

The results from the project have been demonstrated at several occasions to trigger a pre-selected audience to meet and discuss the solutions and findings. Pre-selected audience where people representing a domain which have to buy-in to findings and recommendations to establish the momentum required to move closer to industrialization. Typically architects, HMI designers, User Experience concept developers, application developers, safety specialists at Volvo Cars.

The PLINTA project has proactively approached other projects and stakeholders to identify demo use cases which could trigger a brought audience in parallel with demonstrating the capabilities of the PLINTA platform. Example are the use cases Instant Drive off and Remote Control.

Other examples where the experiences from the PLINTA project have been shown are through Pelagicore's work in GENIVI, for instance in EG HAF, but also through the open sourcing of components developed during the project (e.g. ivi-logger). The PLINTA demonstrator was also shown during the GENIVI AMM hosted in Gothenburg in May 2014.

6.1 Public Presentation

Work from the project has been presented at:

- Vehicle ICT Arena, Lindholmen, Göteborg 4. September 2014,
- Genivi all member meeting in Göteborg 20-23. May 2014
- Elektronik i Fordon, Göteborg 8-9 April, 2014
- Vinnova FFI conference, Göteborg Lindholmen 6. November 2014
- Demos at Volvo Cars for specifically invited persons from different departments, with different domain background, several events.

In addition work has also been presented at (non public):

- Volvo Cars Technology Exhibition 2-4 December, 2014

6.2 Knowledge and results dissemination

The HMI Safety Framework, described in the Software Architecture Report, is based on a concept developed in another Vinnova sponsored project - SICS. It might be worthwhile following up on further development of the Safety API, developed in the SICS project, and consider incorporating that functionality into the PLINTA context, should we choose to implement the HMI Safety Framework.

Results from the project concerning the Secure Gateway will be reused and further developed in the context the Vinnova sponsored project “Second Road fas 2”.

7 Conclusions and future research

7.1 Conclusions

The conclusions from the PLINTA project is that it is possible to use an open and flexible environment for the development of new functionality in the infotainment area. It is also the PLINTA partner's views that there exist a conception that much of this is already industrialized by the automotive industry and available today. However, this is not the case. Adding functionality or new user experiences into todays vehicle infotainment systems is in reality a time consuming and costly task due to

- Current agreements and commercial arrangement in Automotive industry are not developed for open platforms.
- The eco systems are not there yet. Too many platforms and no strong driving force with a clear business model. A consolidation of the many short lived IHU Programs into a few platforms with trusted long term evolution independent of vehicle program cycles is required.

To overcome these barriers, the hope is that the work in PLINTA can trigger and inspire the Swedish automotive industry to seek out the opportunities for bringing these ideas to life. Hence, an important part of the PLINTA project has been to demonstrate and show the possibilities of an open and flexible platform.

7.2 Future work

Below are a few ideas for future research and work based on the challenges found during the project.

- Support the development of new business models that adapts to the conclusions from the project.
- More work is needed to fit the secure gateway into the vehicle system architecture in order to industrialize the concept in a good way.
- More work in the Public Key Infrastructure (PKI) area is needed. Parts of the security in the Secure Gateway concept relies on certificates and the different methods how to distribute and protect the certificates today are not sufficient.
- The face tracking systems integration to the IHU should be further explored. Current Active Safety systems are executed in a very closed and secure environment but options for using face tracking to adjust options in the HMI are relevant.
- Fully implemented app container solution according to the findings from this project with automatic app validation, third party app deployment etc.

8 Participating Parties and Contact Persons

Anders Bengtsson
HiQ Göteborg

Jacob Juul
Volvo Cars Corporation

Johan Kristensson
Semcon Sweden AB

Alwin Bakkenes
Pelagicore AB