Design and implementation of an ITS station to bridge automotive and IoT systems

Master Thesis in Computer Science and Networking

Supervisors: Prof. Piero Castoldi, Dr. Paolo Pagano
Candidate: Michele Carignani
BACKGROUND
BACKGROUND

• **Internet of Things** and Wireless Sensor Networks technologies
BACKGROUND

• Internet of Things and Wireless Sensor Networks technologies
• Connected cars and Cooperative Intelligent Transport Systems
OUTLINE

BACKGROUND

• **Internet of Things** and Wireless Sensor Networks technologies
• Connected cars and **Cooperative Intelligent Transport Systems**
OUTLINE

BACKGROUND

• **Internet of Things** and Wireless Sensor Networks technologies
• Connected cars and **Cooperative Intelligent Transport Systems**

DESIGN

• **Integration** of automotive and IoT
OUTLINE

BACKGROUND

• Internet of Things and Wireless Sensor Networks technologies
• Connected cars and Cooperative Intelligent Transport Systems

DESIGN

• Integration of automotive and IoT
• System hardware / software design
OUTLINE

BACKGROUND

• Internet of Things and Wireless Sensor Networks technologies
• Connected cars and Cooperative Intelligent Transport Systems

DESIGN

• Integration of automotive and IoT
• System hardware / software design

IMPLEMENTATION AND RESULTS
OUTLINE

BACKGROUND

• **Internet of Things** and Wireless Sensor Networks technologies
• Connected cars and **Cooperative Intelligent Transport Systems**

DESIGN

• **Integration** of automotive and IoT
• System hardware / software **design**

IMPLEMENTATION AND RESULTS

• **Facilities** implementation
OUTLINE

BACKGROUND

• **Internet of Things** and Wireless Sensor Networks technologies
• Connected cars and **Cooperative Intelligent Transport Systems**

DESIGN

• **Integration** of automotive and IoT
• System hardware / software **design**

IMPLEMENTATION AND RESULTS

• **Facilities** implementation
• **Results** achieved for the **URBELOG** project
BACKGROUND
“A global infrastructure [...] interconnecting physical and virtual things [...]” (ITU-T)
Internet of Things

“A global infrastructure [...] interconnecting physical and virtual things [...]” (ITU-T)

- 2014: 10 billion new processors
- 2% for GP, 98% embedded
- Worldwide market
- 158.6 billion $ by 2015
- steady growth of 7% in 5 years
“A global infrastructure [...] interconnecting physical and virtual things [...]” (ITU-T)

- 2014: 10 billion new processors
- 2% for GP, 98% embedded
- Worldwide market
- 158.6 billion $ by 2015
- steady growth of 7% in 5 years
Machine to Machine (M2M)

Connected devices (billions)

Source: http://www.ericsson.com/mobility-report
IoT Standard open stack
IoT Standard open stack

- **IEEE 802.15.4**
  - LR-WPAN
  - short-range,
  - low-data-rate,
  - efficiency (cost, power)

- **6LoWPAN**
  - IPv6 over Low Power
  - Minimal (code, memory)
  - End-to-end with Internet

- **CoAP**
  - REST-ful model
  - Lightweight HTTP-like
IoT Standard open stack

- **IEEE 802.15.4**
  - LR-WPAN
  - short-range,
  - low-data-rate,
  - efficiency (cost, power)

- **6LoWPAN**
  - IPv6 over Low Power
  - Minimal (code, memory)
  - End-to-end with Internet

- **CoAP**
  - REST-ful model
  - Lightweight HTTP-like
From the fully connected car...

- Connected car is a reality
- Hundreds of ECUs in the on-board networks
- Monitoring, diagnostics, infotainment (apps, music...)
- Mostly closed and proprietary
... to Cooperative ITS

systems “to support transportation of goods and humans with information and communication technologies in order to efficiently and safely use the transport infrastructure and transport means (cars, trains, planes, ships)”
Standardization

Release 1 of the standards jointly announced by ETSI and CEN on Feb 2014

Operational deployment started based on published standards

Release 2 features being discussed
ITS: EU Standards in force (1)
ITS: EU Standards in force (1)

- **ITS Station** (vehicular, road-side, central, personal)
• **ITS Station** (vehicular, road-side, central, personal)

• ~ ISO/OSI layered structure but:
ITS: EU Standards in force (1)

- **ITS Station** (vehicular, road-side, central, personal)
- ~ ISO/OSI layered structure but:
  - Full stack Management and Security
ITS: EU Standards in force (1)

- **ITS Station** (vehicular, road-side, central, personal)
- ~ ISO/OSI layered structure but:
  - Full stack Management and Security
  - Novel layer of Facilities
IEEE 802.11p
• Access for fast moving objects (notably vehicles)

GeoNetworking
• Provides addressing based on geographical position (geobroadcast, geomulticast, unicast)

Basic Transport Protocol
• Simple multiplexing and demultiplexing
• Best effort
ITS: Facilities
ITS: Facilities

- Facilities
  - Common Facilities
  - Domain Facilities
    - Application support
    - Information support
    - Communication support

- FA-SAP
- SF-SAP
- MF-SAP
- NF-SAP
ITS: Facilities

Provide (among others):
ITS: Facilities

Provide (among others):

- Time and Positioning
ITS: Facilities

Provide (among others):
• Time and Positioning
• Billing and payment
ITS: Facilities

Provide (among others):

• Time and Positioning
• Billing and payment
• Map service
ITS: Facilities

Provide (among others):

- Time and Positioning
- Billing and payment
- Map service
- Local Dynamic Map
ITS: Facilities

Provide (among others):

- Time and Positioning
- Billing and payment
- Map service
- Local Dynamic Map
- Decentralized Environmental Notifications
ITS: Facilities

Provide (among others):

- Time and Positioning
- Billing and payment
- Map service
- Local Dynamic Map
- Decentralized Environmental Notifications
- Cooperative Awareness
ITS: Facilities

Provide (among others):

- Time and Positioning
- Billing and payment
- Map service
- Local Dynamic Map
- Decentralized Environmental Notifications
- Cooperative Awareness
ITS: Facilities

Provide (among others):

- Time and Positioning
- Billing and payment
- Map service
- Local Dynamic Map
- Decentralized Environmental Notifications
- Cooperative Awareness
ITS and IoT integration: motivations

- **No current inclusion** of IoT into ITS-S
- The vehicle as a *smart environment* and *smart objects* for ITS
- Enabling **Smart Cities**
- Applications
  - Verified *QoS* in goods logistics
  - Optimized *routing* and management
  - ... among others
ITS and IoT integration: how

- Both **on-board** or **road-side** IoT
- Access: **IEEE 802.15.4**
- Network and Transport: UDP supported, **6LoWPAN, RPL** needed
- Facility: smart objects resources integrated within **CA messages**, ....
ITS and IoT integration: how

- Both **on-board** or **road-side** IoT

- Access: IEEE 802.15.4

- Network and Transport: UDP supported, **6LoWPAN**, RPL needed

- Facility: smart objects resources integrated within **CA messages**, ....

Currently new standard being edited by

[CNIT]
Integration requirements
Integration requirements

- Multi-MAC capabilities
Integration requirements

- Multi-MAC capabilities
- Constrained resources, need for performance, real time requirements
Integration requirements

- Multi-MAC capabilities
- Constrained resources, need for performance, real time requirements
- Easily extensible (many parties involved)
Integration requirements

• Multi-MAC capabilities

• Constrained resources, need for performance, real time requirements

• Easily extensible (many parties involved)

• Following the concepts and naming of ETSI standards
Integration requirements

- Multi-MAC capabilities
- Constrained resources, need for performance, real time requirements
- Easily extensible (many parties involved)
- Following the concepts and naming of ETSI standards
- Implementing standard communication protocols at any layer
Design

SELECTED HARDWARE

- Beagle Bone Black
- IEEE 802.15.4 tranceiver
- GPS/GPRS
- IEEE 802.11n
- CAN / OBDII
- IEEE 802.11p gateway (ITRI IWCUv5)
SELECTED HARDWARE

- Beagle Bone Black
- IEEE 802.15.4 tranceiver
- GPS/GPRS
- IEEE 802.11n
- CAN / OBDII
- IEEE 802.11p gateway (ITRI IWCUv5)
Design

SELECTED HARDWARE

- Beagle Bone Black
- IEEE 802.15.4 tranceiver
- GPS/GPRS
- IEEE 802.11n
- CAN / OBDII
- IEEE 802.11p gateway (ITRI IWCUv5)

DESIGNED AND DEVELOPED SOFTWARE (Thesis contribution)

- A **framework** for applications and services
- An event and I/O **orchestrator** (Implemented on embedded Linux)
- **Gateways** and **Facilities** implementation
Framework design

- Component based design
- Abstracting **events** management
- Introducing **Services**
- **ItsStation**
  - Wrap event loop
- Refer service and application registries
Framework design

• Component based design

• Abstracting **events** management

• Introducing **Services**

• **ItsStation**
  • Wrap event loop
  • Refer service and application registries
Framework design

- Component based design
- Abstracting **events** management
- Introducing **Services**
  - **ItsStation**
    - Wrap event loop
  - Refer service and application registries
Framework design

- Component based design
- Abstracting **events** management
- Introducing **Services**
- **ItsStation**
  - Wrap event loop
  - Refer service and application registries

![Diagram showing the relationships between ItsEventInjector, ItsService, ItsApplication, Service1, ServiceN, ServiceN Interface, and ServiceN Implementation1. The diagram illustrates the hierarchical nature of the design with directed associations and includes a small portion of code snippets demonstrating the implementation relationships.]
Framework design

- Component based design
- Abstracting **events** management
- Introducing **Services**
  - ItsStation
    - Wrap event loop
    - Refer service and application registries
Framework design

- Component based design

- Abstracting **events** management

- Introducing **Services**

- **ItsStation**
  - Wrap event loop
  - Refer service and application registries
IMPLEMENTATION
Framework implementation

- The core library of **nodejs**, "non-blocking I/O" in C language
- Portable wrapping over **epoll**, asynchronous model (asynchronous handling of file descriptors, timers)
- Thread pool, work dispatch, threading and sync primitives
Events management

- By means of the **ItsStation** (I/O) and **ItsEventInjector** (event engine)
- Abstracts using **std::Function**
- Different events handled asynchronously

**EVENTS API**

- `timer()`
- `periodically()`
- `asynchronously()`
- `shutdown()`
- `on()`
- `trigger()`
Example: CAN Service Interface

- Controlled Area Network: manufacturer in-vehicle network
- A service to retrieve information from the car such as RPM, speed, odometer

```cpp
class CanServiceInterface : public noes::StationService {
public:
    CanServiceInterface();
    CanServiceInterface(noes::ItsStation* s);
    virtual can_data_t get() = 0;
    virtual void print(can_data_t d) = 0;
};
```
Example: CAN Service implementation

- HW: Implemented via OBDII adapter over serial (i.e. /dev/ttyS1)
- SF: **Query periodically** and **cache data** structure

```cpp
OBDDIServiceImpl::OBDDIServiceImpl(noes::ItsStation *s):
    CanServiceInterface(s)
{
    try {
        update();
        this->periodically(1000, [this](int k){
            this->update();
        });
    } catch (SerialException &e) {
        // ...
    }
}
```
CA facility implementation

API

- **generate()**: collection and encapsulation

- **encode()**: serialization in ASN.1 Unaligned-PackedEncodingRules

- **decode()**: deserialization from bytes

- **start_dissemination()**: manage CAMs dissemination

- **send()**: request from GN/BTP
CA facility implementation

API

- **generate()**: collection and encapsulation
- **encode()**: serialization in ASN.1 Unaligned-PackedEncodingRules
- **decode()**: deserialization from bytes
- **start_dissemination()**: manage CAMs dissemination
- **send()**: request from GN/BTP

Implementation details

- Extension of CAM PDU

```
LoadingZoneContainer ::= SEQUENCE {
  vin UTF8String SIZE ((18)),
  engineStatus EngineeringStatus,
  authorizationID AuthorizationID,
  cargoVolume CargoVolume
}
```
CA facility implementation

API

- **generate()**: collection and encapsulation
- **encode()**: serialization in ASN.1 Unaligned-PackedEncodingRules
- **decode()**: deserialization from bytes
- **start_dissemination()**: manage CAMs dissemination
- **send()**: request from GN/BTP

Implementation details

- Extension of CAM PDU
  
  \[
  \text{LoadingZoneContainer} ::= \text{SEQUENCE}\{\text{vin UTF8String SIZE ((18)), engineStatus EngineStatus}, \text{authorizationID AuthorizationID, cargoVolume CargoVolume}\}
  \]

- ASN.1, asn1c, CAM_t type
BTP Gateway implementation

- Using 802.11p board in bridge mode
- Implemented ETSI Protocol Control Information (PCI)
- PCI serialization
- Gateway module on the IWCUv5

```c
typedef struct btp_pci_s {
    int btp_type;
    int packet_type;
    int store_and_forward;
    int latitude;
    int longitude;
    int macAddr_i[6];
    int dport;
    int sport;
} btp_pci_t;
```
BTP Gateway implementation

- Using 802.11p board in bridge mode
- Implemented ETSI Protocol Control Information (PCI)
- PCI serialization
- Gateway module on the IWCUv5

```
typedef struct btp_pci_s {
    int btp_type;
    int packet_type;
    int store_and_forward;
    int latitude;
    int longitude;
    int macAddr_i[6];
    int dport;
    int sport;
} btp_pci_t;
```
IoT Gateway implementation
IoT Gateway implementation

- PHY and MAC, native Linux available (wpan layer 2 socket)
IoT Gateway implementation

- PHY and MAC, native Linux available (wpan layer 2 socket)

- Contiki: “the operating system for the IoT”, containing uip (6LoWPAN) and CoAP implementations
IoT Gateway implementation

- PHY and MAC, native Linux **available** (wpan layer 2 socket)
- Contiki: “the operating system for the IoT”, containing **uip** (6LoWPAN) and CoAP implementations
- Implementation on ItsStation: Contiki as a Linux process + virtual tunnel interface (i.e. `/dev/net/tun0`)
IoT Gateway implementation

- PHY and MAC, native Linux **available** (wpan layer 2 socket)

- Contiki: “the operating system for the IoT”, containing **uip** (6LoWPAN) and CoAP implementations

- Implementation on ItsStation: Contiki as a Linux process + virtual tunnel interface (i.e. `/dev/net/tun0`)

- Therefore IoT is available usual **sockets** API (via libuv wrapper)
RESULTS
URBELOG Project

- Last mile logistic optimization:
  - real time tracking of goods to enforce good practices
  - reduce emissions
- Loading zone application
  - Parking slots real time discovery, booking
  - Parking slot status monitored and advertised
Parking slot use case

- URBELOG use case: real time tracking of parking lot status with vehicle authentication

- Smart cameras network (IoT segment)

- Vehicle acknowledgment and free spots advertisement via CAMs (802.11p + GeoNet)

- Notification of change status to Service Center (also notifies infringements)
URBELOG: Developed devices
IoT segment: **Smart camera**
- Exposes calculated data (EMPTY, BUSY) as a REST Resource
- Reachable over 6LoWPAN
URBELOG: Developed devices

IoT segment: **Smart camera**
- Exposes calculated data (EMPTY, BUSY) as a REST Resource
- Reachable over 6LoWPAN

Road Side ITS Station, **UrbelogRSU**:
- Monitors smart cameras (**IoT Gateway**)
- Listens to CAM dissemination (**BTP Gateway**)
- Implements **timers** (handshake timeout, parking expiration)
- Forwards events to the Smart City Center (**SCC Gateway, MQTT**)
URBELOG: Developed devices

IoT segment: **Smart camera**
- Exposes calculated data (EMPTY, BUSY) as a REST Resource
- Reachable over 6LoWPAN

Road Side ITS Station, **UrbelogRSU**:
- Monitors smart cameras (IoT Gateway)
- Listens to CAM dissemination (BTP Gateway)
- Implements timers (handshake timeout, parking expiration)
- Forwards events to the Smart City Center (SCC Gateway, MQTT)

Vehicular ITS Station, **UrbelogOBU**:
- Disseminates CAM (BTP Gateway)
- Reads vehicle data (rpm) to include in CAM (OBDII Service)
“A prototype bridge between automotive and IoT”

by Carignani et al. will be presented at
IEEE World Forum on Internet of Things,
Automotive workshop on Dec 14th 2015
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th
On field demo on Nov. 13th

Service Center

WSN

CAM - VANET

(11.p)
On field demo on Nov. 13th
On field demo on Nov. 13th

- WSN
- CAM - VANET
- Service Center
- MQTT (3G)
- Smart Bench
- Smart BusStop
- CAM - VANET (11.p)
Conclusions

• New system **integrating IoT and C-ITS** was implemented
Conclusions

• New system integrating IoT and C-ITS was implemented

• Feasibility of one ITS application
Conclusions

- New system **integrating IoT and C-ITS** was implemented
- **Feasibility** of one ITS application
- **Soundness** of the IoT/C-ITS architecture was demonstrated on field
Conclusions

- New system **integrating IoT and C-ITS** was implemented
- **Feasibility** of one ITS application
- **Soundness** of the IoT/C-ITS architecture was demonstrated on field
- The designed framework led to a **fast and easy implementation** on the URBELOG use case
Thanks for your kind attention

Design and implementation of an ITS station to bridge automotive and IoT systems
Candidate: Michele Carignani
ITS Scenario
ITS Scenario

Long haul communications
Long haul communications

- eCall
- Tracking
- Infotainment
ITS Scenario

Long haul communications
- eCall
- Tracking
- Infotainment

Dedicated Short Radio Communications
ITS Scenario

Long haul communications
- eCall
- Tracking
- Infotainment

Dedicated Short Radio Communications
- Cooperative safety
- Traffic efficiency
- Signage
- Advertisement
Abstract Syntax Notation ASN.1

- ASN.1, a joint standard of ISO, IEC, and ITU-T.
- Standard and notation that describes rules and structures for
  - representing,
  - encoding,
  - transmitting, and
  - decoding