TECHNOLOGY • RESEARCH HIGHLIGHTS • VISIONS • SCIENCE

142

Multi-Service Architecture for mobility services

Olli Pihlajamaa | Immo Heino | Armi Vilkman

VTT TECHNOLOGY 142

Multi-Service Architecture for mobility services

Olli Pihlajamaa, Immo Heino & Armi Vilkman

ISBN 978-951-38-8077-4 (Soft back ed.)
ISBN 978-951-38-8078-1 (URL: http://www.vtt.fi/publications/index.jsp)

VTT Technology 142

ISSN-L 2242-1211
ISSN 2242-1211 (Print)
ISSN 2242-122X (Online)

JULKAISIJA – UTGIVARE – PUBLISHER

VTT
PL 1000 (Tekniikantie 4 A, Espoo)
02044 VTT
Puh. 020 722 111, faksi 020 722 7001

VTT
PB 1000 (Teknikvägen 4 A, Esbo)
FI-02044 VTT
Tfn +358 20 722 111, telefax +358 20 722 7001

VTT Technical Research Centre of Finland
P.O. Box 1000 (Tekniikantie 4 A, Espoo)
FI-02044 VTT, Finland
Tel. +358 20 722 111, fax +358 20 722 7001

Kopijyvä Oy, Kuopio 2013

Multi-Service Architecture for mobility services

Olli Pihlajamaa, Immo Heino & Armi Vilkman.
Espoo 2013. VTT Technology 142. 99 p. + app. 71 p.

Abstract

The work done in the SUNTIO2 project and reported in this document aims at providing assets for realizing the Multi-Service Model for the creation, provision and supply of mobility services. It is continuation for the work done in PASTORI and SUNTIO projects in which the Multi-Service Model has been developed.

The central result of the work culminates to the set of requirements for the system realizing Multi-Service Model and the functional architecture for such a system. The requirements, and hence the functional architecture, are based on several sources. First, the participants of the SUNTIO2 project formed a representative selection of potential roles in the Multi-Service Model and were, thus, provided invaluable sources of stakeholder aspirations. In addition, discussions with numerous other stakeholders during the project complemented this view. The extensive survey of the state-of-the-art as well as past and on-going efforts in the field of ITS and mobility services gave also important input to the work. Furthermore, the analysis of two service use cases helped greatly in the concretization of the requirements.

The architecture for Multi-Service Facilitation (MSF) created on the base of the requirements introduces the high-level functionality required for the realization of the Multi-Service Model. It includes such central elements as: MSF Centre for server side functionality, MSF Client Framework for offering user interface for MSF Centre and building service frontends, MSF Service Framework for service providers to co-operate in service provision and B2B Marketplace to facilitate the business ecosystem.

Keywords Multi-Service Model, Intelligent Transportation Systems (ITS), mobility services, requirements, architecture, state-of-the-art

Preface

The Multi-Service Model is a Finnish concept for intelligent transport services produced by a heterogeneous value network or an enterprise cluster. The model combines different “silo-services” into a common service platform, which can have many different roles and actions from data delivery to complex service supply systems. Multi-service is mentioned in the new intelligent transport strategy of the Ministry of Transport and Communications of Finland as a new possibility for Finland to offer public and private services in a cost-effective way. VTT and numerous businesses and authorities have been involved in a wide range of projects and pilots related to multi-service.

The most important motivation for Multi-Service Model has been bringing different public and commercial services onto the same platform easily accessible and cost-effectively, which could enable the wide implementation and penetration of traffic related services. Other motivators for development include creating new businesses and jobs in Finland, combining the know-how and expertise of separate stakeholders into a somewhat bigger enterprise, helping SME companies export their services jointly, and helping all of us learn from each other how we can together create services that make the everyday travelling and transport more convenient, timely, predictable and safe.

The authors wish to acknowledge both the individuals and companies involved in this effort for their commitment and great ideas: Tom Warras and Kimmo Ahola from Tekes, Ilkka Tuominen from DNA, Matti Lankinen from Indagon, Juha Laakso from Infotripla, Markku Rauhamaa and Zhi Chun Honkasalo from Nokia Siemens Networks, Börje Nummelin and Tuomo Roivainen from Semel, Jouni Sintonen from Teliasonera, Teemu Vähäkainu from Tieto, and Sami Sahala from City of Helsinki. We also wish to thank Seppo Öörni and Marko Forsblom from the Ministry of Transport and Communications of Finland, Juuso Kummala and Kari Hiltunen from the Traffic Agency, and Mia Nykopp from the Traffic Safety Agency for introducing the multi-service idea to their respective organisations. Many thanks also to all the other companies and stakeholders in previous multi-service projects for their excellent ideas and contributions.

Espoo 11.11.2013

Armi Vilkman (project manager), Olli Pihlajamaa and Immo Heino

Contents

List of abbreviations

A2DP	Advanced Audio Distribution Profile
AASHTO	American Association of Highway and Transportation Officials
AECID	Adaptive Enhanced Cell Identity
AFLT	Advanced Forward Link Triangulation
AoA	Angle of Arrival
API	Application Programming Interface
APTA	American Public Transportation Association
ASN.1	Abstract Syntax Notation One
B2B	Business-to-business
B2C	Business-to-customer
BS	Base Station
C2C-CC	Car to Car Consortium
CAI	Commonly Agreed Business to Business Interface
CALM	Communication Access for Land Mobile
CAN	Controller Area Network
CDR	Call Data Record
CEN	European Committee for Standardization
CFC	Cellular Floating Car
CID	Cell ID
CN	Core Network
CO_2	Carbon Dioxide
CPU	Central Processing Unit
CS	Circuit Switched
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
CVIS	Cooperative Vehicle-Infrastructure System
DAP	Device Attestation Protocol
DATEX	Data Exchange

DCMA	Code Division Multiple Access
DHCP	Dynamic Host Configuration Protocol
DR	Dead Reckoning
DSP	Digital Signal Processor
DSRC	Dedicated Short Range Communication
EC	European Commission
E-CID	Enhanced Cell ID
ECU	Electronic Control Unit
EETS	European Electronic Toll Service
EGNOS	European Geostationary Navigation Overlay Service
EMV	Europay, Mastercard and VISA
ETC	Electronic Toll Collection
EU	European Unions
EV	Electronic Vehicles
FCC	Federal Communications Commission
FCD	Floating Car Data
FI-PPP	Future Internet Public Private Partnership
FRAME	Framework Architecture Made for Europe
FTP	File Transfer Protocol
FVD	Floating Vehicle Data
GDI	Graphics Device Interface
GDP	Gross Domestic Product
GloNASS	Globalnaya navigatsionnaya sputnikovaya sistema
GNSS	Global Navigation Satellite System
GSM	Global System for Mobile Communications
GTFS	General Transit Feed Specification
HFP	Hands-Free Profile
HGV	Heavy Goods Vehicle
HOV	High Occupancy Vehicle
HTTP	HyperText Transfer Protocol

IEEE	Institute of Electrical and Electronics Engineers
IFOPT	Identification of Fixed Objects in Public Transport
IoT	Internet of Things
IP	Internet Protocol
IPS	Intelligent Parking System
ISO	International Organization for Standardization
ITS	Intelligent Transport System
ITS-G5	Geo networking protocol for V2V and V2I communication
IVI	In-Vehicle Infotainment
IVMM	Interactive Voting based Map Matching
JARTIC	Japan Road Traffic Center
KAREN	Keystone Architecture Required for European Networks
LA	Location Area
LBS	Location Based Services
LCD	Liquid Crystal Display
LIN	Local Interconnect Network
LTE	Long Term Evolution
M2M	Machine to Machine
MAC	Media Access Control
MANET	Mobile ad hoc network
MDM	Mobility Digital Marketplace
MLIT	Ministry of Land, Infrastructure, Transport and Tourism
MOST	Media Oriented System Transport
MSF	Multi Service Facilitation
MTC	Metropolitan Transmission Commission
NMEA	National Marine Electronics Association
OBD	On Board Diagnostics
OBU	On Board Unit
OD	Origin-Destination
OEM	Original Equipment Manufacturer

OGC	Open Geospatial Consortium
OMG	Object Management Group
OpenLS	Open Location Services Interface Standard
OSGi	Open Services Gateway initiative
OTDOA	Observed Time Difference of Arrival
P2P	Peer-to-Peer
PKI	Public Key Infrastructure
PND	Personal Navigation Device
POI	Point of Interest
PT	Public Transportation
RA	Routing Area
RAN	Radio Access Network
RCA	Radio Corporation of America
RDSS	Regional Data Service Server
RF	Radio Frequency
RFID	Radio Frequency Identification
RSS	Received Signal Strength
RSTD	Reference Signal Time Difference
RTOS	Real Time Operating System
RTP	Real Time Protocol
RTTI	Real-Time Traffic and Travel Information
SATNAV	Satellite Navigation
SDI	Serial digital interface
SDK	Software Development Kit
SIM	Subscriber Identity Module
SIR	Signal to interference Ratio
SLA	Service Level Agreement
SME	Small and medium enterprises
SMTP	Simple Mail Transfer Protocol
SO	Service Orientation

SOA	Service Oriented Architecture
TCP	Transmission Control Protocol
TDOA	Time Difference of Arrival
TEKES	Teknologian ja innovaatioiden kehittämiskeskus
TIC	Traffic information management
TISP	Traffic Information Service Provider
TPEG	Transport Protocol Experts Group
UI	User Interface
UK	United Kingdom
UML	Universal Modelling Language
UMTS	Universal Mobile Communication System
UPnP	Universal Plug and Play Protocol
URI	Uniform Resource Identifier
USA	United States of America
USB	Universal Serial Bus
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
VANET	Vehicle Area Network
VICS	Vehicle Information Communication System
VNC	Virtual Network Computing
W3C	Word Wide Web Consortium
WAAS	Wide Area Augmentation
WAVE	Wireless Access in Vehicular Environment
WFS	Web Feature Service
WiFi	Wireless Fidelity (wireless internet)
WLAN	Wireless Local Area Network
WMS	Web Map Service
xFCD	Extended Floating Car Data
XML	Extensible Markup Language

1. Introduction

The Multi-Service Model is a Finnish concept proposed originally by Vilkman et al. (2010, 2011). It aims at providing a one stop shop for end users to access mobility services, and an efficient business ecosystem for multi-actor service networks providing mobility services or service elements that can be utilize in the building of services. Vilkman et al. (2010) lists the basic features and requirements of the Multi-Service Model:

- a wide array of services that are modular and modifiable (enabling service bundling based on user needs)
- ensuring information security and privacy as well as fulfilling other legislative requirements
- open and transparent architecture solutions and interfaces
- the use of harmonized solutions and reliable technology
- utilization of existing components and solutions
- independence of the technology used
- terminal-independence.

The Multi-Service Model is not bound to any specific technological solution but defines merely a way to create, provide and supply services in a way where formerly separate, detached services and their providers in their silos become part of a greater whole through service networks and common ways to supply services to end users. The purpose of this document is to introduce a technology independent, high-level architecture that enables the realization of this kind of model.

The rest of the document starts in Chapter 2 with an introduction of Intelligent Transportation Systems (ITS) and some of their central enablers in order to give a view on the landscape where the Multi-Service Model is supposed to be realized.

Chapter 2 examines the Multi-Service Model itself and introduces the core value chain as well as main actors that are involved in the creation, provision and supply of the mobility services. In this point we introduce the concept Multi-Service Facilitation (MSF) that refers to the means or enablers that support the realization of the Multi-Service Model in the service creation, provision and supply – the main topic of this document. In addition, Chapter 2 also summarizes the benefits of the multi-service approach and introduces potential business examples. The topics in this chapter provide necessary background for the business requirements and high-level stakeholder aspirations that are later on in the document taken into account in the architectural requirements.

Chapter 3 surveys the past and on-going efforts that are relevant for or similar to the Multi-Service Model. The 511 service from North-America as well as ITS efforts in Japan starts the chapter to bring in ideas from outside of Europe. Examples of significant European efforts follow and finally the chain of Finnish efforts are summarized in order to give some light to the evolution of the Finnish national ITS scene. The related work, shortly introduced in this chapter, forms an important contribution to the requirements for the architecture as they have used vast amount of resources to analyse the aspirations and needs that are related to the mobility services and their provision.

Chapter 4 concentrates on the consolidation of the requirements for the Multi-Service Facilitation. It first documents the sources of requirements as well as the methodology used for the requirements collection. After that, it dives deeper in the requirement sources and lists central stakeholder aspirations that were brought up during the SUNTIO2 project. This is followed by a summary of two representative use cases, floating car data (FCD) collection and intelligent parking service that were analysed in order to concretize and prioritize the stakeholder aspirations and high-level requirements found in the documentation of other efforts. This chapter is concluded by the summary of requirements for the Multi-Service Facilitation.

After the requirements, Chapter 5 finally proposes a functional architecture for Multi-Service Facilitation. The chapter starts with the definition of the architectural context that presents the external actors and corresponding external interfaces of the MSF system. After definition of the architectural context, follows the definition of the architecture itself and description of its functional components. Core data model related to the architecture is then presented in a conceptual level. The chapter is concluded with a discussion on the realization alternatives for the architecture.

The main document is concluded by key findings from the project period, suggestions for the future work and a list of simplified steps towards the realization of the Multi-Service Model in the service creation, provision, supply and use.

Attached in the end of the document are four appendices that report some of the extensive background work done in SUNTIO2 to support the requirement collection and creation of the MSF architecture.

The first two appendices include use case analyses used to deduce concrete requirements for the MSF architecture. The first case (Appendix A) surveys thoroughly the state of the art of floating vehicle data (FVD) collection, data utilization and future directions. The second use case (Appendix B) describes a future parking service that is not yet existing but could be realized using the enabler of the Multi-Service facilitation element proposed in this document.

Appendix C is related to the FVD use case presented in Appendix A. During the project several participants expressed their interest to a common framework and data set for FCD collection as it does not yet exist in wide practical use anywhere.

It was seen that such a common FVD data set would enable creation of a FVD market place that could provide a good example of the realization of the Multi-Service Model in B2B traffic data collection and provision. This appendix examines the state-of-the-art and makes a proposal for a minimum data set for the FVD data for the basis of the further formal definition.

Appendix D complements Chapter 3 by introducing some of the commercial approaches that resembles the Multi-Service Model.

2. Intelligent Transport Systems (ITS)

2.1 Definition of ITS and motivation for its use

Intelligent transport systems (ITS) and services apply information and communication systems and services (advanced technologies in electronics, communications, computers, control and sensing and detecting) to all kinds of transportation systems in order to improve the safety, efficiency and service of traffic and transport. The key element in ITS is real-time information of traffic or transport situations. (Aakre et al. 2012).

The objectives of using ITS include improving traffic safety, relieving traffic congestion, improving transportation efficiency, reducing air pollution, increasing energy efficiency, and promoting the development of related industries. ITS can be grouped in different ways, including:

1. Demand Management
2. Traffic Management
3. Driver assistance and vehicle control
4. (Commercial) Fleet Management and operation.

Demand management can contain ”soft” methods like information, public transport (PT) services including PT information services, pre-trip information, parking information and payments, car-pooling and pre-booking, city car clubbing etc. and “tough” methods like pricing, electronic payments, road-tolling, congestion charging, restrictions for certain vehicle types or parking etc. (which all may be based on driver or vehicle identification).

Traffic management is usually in the hands of the authorities and includes roadside equipment, traffic lights, traffic guidance and control, route guidance, railroad level-crossing systems, incident management etc., which are controlled from traffic management centres.

Driver assistance and vehicle control systems include driving information during travel, emission and energy consumption monitoring and feedback, route and Point Of Interest (POI) guidance, travel service information, incident alerts, eCalls and other emergency services, back-up collision prevention, side collision prevention, intersection collision prevention, and visual improvement of traffic accident prevention. Driver assistance systems can be different for different vehicle types;

e.g. ambulances and police vehicles need different systems from normal passenger cars. This category also includes emerging cooperative systems where vehicles use different sensors and monitoring devices and can communicate with other vehicles as well as central systems.

Fleet Management and operation can include all professional services either for public transport or goods transport, and hauling from individual vehicle management to whole fleet and cargo management (including e.g. electronic customs clearance of commercial vehicles, automatic security roadside inspections, security service and maintenance services).

2.2 Enablers of ITS

2.2.1 General

The main technology enablers for intelligent services or multi-services fall into three groups: a) devices, b) software, applications and services and c) communications. End-users have to have a device or platform for the applications to receive and send information. This also applies to the road environment; there have to be monitoring systems to measure the status of the weather or traffic. Then the monitored data must be transmittable through a mobile communication network to a server, and the processed information or service to the end-user. Both software and applications are needed in all chain rings for value-adding services (data gathering, processing, mediating and providing information).

Examples of key elements for service delivery to road users, travellers or hauliers are:

- Monitoring devices and applications (installed in a vehicle or smartphone, or on the road or roadside)
- Devices that can be connected to the web or a central system or can contact other devices in real-time or at given time intervals
- Media or device that provides information (smartphone, vehicle display, navigator, web page etc.)
- Notification apps for personal or user notifications
- Location of the vehicle or traveller
- Information on the character, type and status of the road link (or rail, sea route, metro line etc.): location, speed limit, lanes, width, tunnelling, equipment etc. (some information can be static or slowly changing but may also be variable and time-related)
- Information on the status of road weather and road (rail, sea route, metro line etc.) including fog, slipperiness, snow, disturbances, accidents etc.
- Information on the status of maintenance: snow ploughing, road works etc.

- Mathematical modelling for creating information from raw or monitored data; requires also historical data, contact with other databases etc.
- Communication systems, e.g. wireless networks.

2.2.2 Devices

In-vehicle devices, intelligent road-transport equipment and their related applications are usually designed, built and used for a certain purpose. For example, eCall, an automated emergency call from a vehicle to a local emergency response centre in case of a serious accident, has been defined and standardized as a single service needing special vehicle equipment, although it uses many of the basic elements: monitoring the vehicle and vehicle location, software and modems for making the call and sending the location and other data to a central system (the emergency rescue centre's call centre). Pan-European eCall is planned to be a mandatory and regulated service and will need a dedicated and certified in-vehicle device.

The same situation exists in road tolling. The European Electronic Toll Service (EETS) tries to tackle the problem with separate non-compliant road tolling devices and roadside systems. It is both undesirable and inconvenient to have to acquire a separate road-tolling device for each country in order to pay heavy-goods-vehicle tolls, such as on the route from Finland to Spain, which has four or five different tolling schemes with separate agreements, devices and bills.

The third regulated device is the electronic tachograph. These devices have been harmonized to function similarly in different countries, but they have been developed as a single service, not for integration with e.g. road tolling, emergency services or fleet management services.

Smartphones have been seen to develop as a common platform for different transport-related services. The progress there has been of interest for example in "vehicle-smartphone" cooperation-developing organizations like Genivi Alliance and the Car Connectivity Consortium. Common efforts have been enthusiastic and promising. Also in many general open-platform studies, good ideas have emerged for tackling the challenges of silo services. The vehicle industry and PND (personal navigation device) and map-providing companies are jointly trying lower the separation between vehicle and PND industries. Mobile operators are perhaps the best connective partners in the service chains to urge greater cooperation and interoperability of different services and industry. Motives for interoperability are also helping to develop public transport fare payment to include more means for travellers: payments can be made increasingly with EMV cards (normal credit card with a chip) and smartphones.

2.2.3 Wireless Networks

Currently the most widespread wireless communication technology is the Universal Mobile Telecommunication System (UMTS) based on the CDMA (Code Division Multiple Access) access principle. The UMTS architecture was designed to offer both voice and data services. Data services are based on High Speed Downlink Packet Access (HSDPA) and High Speed Uplink Packet Access; combined these are called High Speed Packet Access (HSPA). Current HSPA offers data rates (peak data) of up to 14 Mbit/s in the downlink and 5.76 Mbit/s in the uplink.

LTE (Long Term Evolution), initiated in 2004, is an enhancement of UMTS. LTE reduces latency for packets, and improves the spectral efficiency of HSPA three- to fourfold in the downlink and two- to threefold in the uplink by using a different channel coding. LTE offers a peak data rate of 100 Mbps for downlink and 50 Mbps for uplink (20 MHz bandwidth) and the system is IP-optimized, i.e. the focus on services is in the packet switched domain. LTE has already been deployed in several countries worldwide and hundreds of devices support it (mostly smartphones and tablets). LTE Advanced is an enhancement to LTE that improves throughput and reduces latencies even further. The peak data rate for downlink is 1 Gbps and uplink 500 Mbps.

In the future, mobile traffic is expected to increase dramatically with the use of entertainment services like music and video (e.g. streaming type services). Another source of such growth is Machine-to-Machine (M2M) and Internet of Things (IoT) applications. Unfortunately the LTE/LTE-A network architecture is not optimally suitable for these types of short message traffic.

Traffic applications use UMTS, LTE/LTE-A infrastructure networks for common traffic services, but also ad-hoc networks are needed for Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) related services. Mobile ad-hoc networks (MANETs) are a self-configuring infrastructure-less networks of mobile devices connected by wireless connections. MANETs are brought to traffic as vehicle-area networks (VANETs), a tool for improving road safety and supporting new services (Faezipour et al. 2012). Several European Union projects like COOPERs and COMeSafety have studied V2V and V2I communications based on wireless network technologies.

Standardization bodies are also working with VANETs. The ISO approach to VANETs is CALM (Communication Access for Land Mobile). ISO TC 204/Working Group 16 has specified a common architecture, network protocols and communication interface definitions using various access technologies including cellular, satellite, infrared, 5 GHz microwave, 60 GHz millimetre-wave, and mobile wireless broadband. EU projects like CVIS (Cooperative Vehicle-Infrastructure System) has studied practical issues related on this architecture vision.

The European automobile industry’s Car to Car Consortium (C2C-CC) is aiming also to develop a de facto standard for VANETs using the C2CNet protocol, which provides multi-hop communication based on geographical routing (Mohammad et al. 2011).

The USA-originated IEEE 1609 WAVE defines the architecture, organization, management structure, communication model, security mechanisms and physical access (Faezipour et al. 2012). IEEE P1609.1 contains the key components of the WAVE system architecture by defining the communication formats and data storage formats. IEEE P1609.2 specifies secure message formats and IEEE P1609.3 network layer protocols and routing-related issues.

Conceptually, VANETs are straightforward, but their design and deployment are technically very challenging. To solve physical layer radio medium sharing problems, a dedicated short range communications system (DSRC) is adopted by all standardization organizations. DSRC is a one-way or two-way short- to medium-range wireless communication channel specifically designed for automotive working in the 5.9 GHz band. DSRC is standardized as 802.11p within the IEEE 802.11 working group, and is also a part of the IEEE P1609 wireless access in a vehicular environment (WAVE, Wireless Access in Vehicular Environment) standard family. 802.11p is used as a basis for the ITS-G5 (geo networking protocol for V2V and V2I communication) standardization in Europe.

2.2.4 Ubi information processing

Technical advances reduce the costs of sensors, actuators and processors, allowing them to be used in any real world object. The process whereby electronic systems communicate and share information autonomously with each other is described as Machine-to-Machine (M2M) communications. M2M communications are characterized by very low power, low cost and low or minimal human intervention. IoT can be seen as a generalization of M2M. It refers to everyday objects that are readable, recognizable, locatable, addressable, and/or controllable via the Internet.

M2M and IoT will make extensive use of wireless communication in both licensed and unlicensed spectrum bands, but most of them will connect with short-range technologies like Wi-Fi. It is estimated that currently around five billion M2M devices are connected to mobile networks; some estimates increase that count to 50 billion by the end of the decade. For example, M2M applications like on board diagnostic systems (OBD) are able to measure several parameters related to the car’s condition and autonomously relay the information to service centres where machines with intelligent algorithms can analyse the need for preventive maintenance without human intervention.

All systems are becoming instrumented and interconnected, which given the explosion of data is putting a significant strain on IT infrastructure and operations. A general problem of modern distributed computing systems is that the complexity and difficulties of their management are becoming a significant limiting factor for further development and utilization. Cloud computing tries to alleviate these problems by enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Sensor networks and cloud computing can be applied in several traffic-related application areas. One of the ways in which cloud computing could assist is better vehicle engineering; another is traffic monitoring and control. For example, the EU project Instant Mobility is attempting to study the conditions needed for dynamic traffic management & integrated urban space, and how to use future internet technologies such as cloud data storage, cloud computing virtualization or services-in-the-cloud.

2.3 ITS based services

In the new update of the ITS strategy of the Ministry of Transport and Communications, the following key services are mentioned as the cornerstones of building a Finnish Intelligent Transport System:

1. ITS architecture
2. Traffic Situation Awareness and Proactive Traffic Management
3. Interoperability and integration of public transport
4. Intelligent automated enforcement
5. Driver support systems
6. Multi-Service Model
7. Smart Logistics
8. Sustainable Mobility.

In Finland, one of the key motivators for ITS systems is building Traffic/Transport System Situation Awareness; it is also second on the list above. Situation awareness is seen as a basic element for both individuals and society for handling choice making in a complex situation. A thesis on situation awareness describes it as following: ”A person that is situationally aware can answer questions about what happens around her, what is going to happen next and what possibilities she has to act. There is no unambiguous definition of situational picture but it can be seen as a support for decision making.” Road, railway, and sea traffic is increasing, which emphasizes the importance of fluent and effective traffic management. To manage traffic, operational actors need a good situation awareness and situational picture and proactive measures. Also, individuals have to make their own safe choices in traffic. (Koistinen 2011)

In the MonArkki project (see Section 4.3.4) the following LBS services were listed as “authority services”:

- Access control - Verification of access rights of persons or vehicles to restricted areas
- Electronic vehicle identity, electronic number plate, electronic license plate
- eCall, automated emergency call with location information
- EETS, European Electronic Tolling Service: one contract, one bill in all European road tolling systems

- Hazardous Goods Tracking
- Livestock Transport Tracking
- (Digital Tachograph)
- “Commercial services” including the following:
- Intelligent Truck Parking
- Fleet management
- Cargo monitoring
- Navigation including access restricted navigation
- Traffic information
- Parking information
- Safety Information
- Payment services
- Pay-as-You Drive Insurance
- Intelligent Speed Advice/Adaptation
- Cooperative systems V2V, V2I etc.
- Infotainment.

The most promising ITS services are those that either fulfil a wide range of needs or are tailored to the specific need of an important customer with a good price. The mobility-related “killer application” for everyone has yet to emerge. As stated earlier, situational awareness is currently a very “hot topic”. It is very important for any traveller or driver to be able to make good choices of mode, route or parking space. Also hauliers need to be aware of different possibilities regarding route, road maintenance, collection/delivery time slots and secure parking for their transport. Road and street traffic and maintenance managers and authorities need to be aware of everything concerning road networks. To get a “holistic big picture” for all requires combining data from different monitoring systems and services.

With intelligent transport systems and services, the “big shift” (Piarc 2004) is changing the traditional road operation and traffic management systems procured and run by authorities to a variety of public and private organised services, which together make the intelligent transport network. The transport network authorities (road and rail operators) are looking for new ways to acquire and maintain telematics systems, searching for ways to use mobile sensors and new data sources and also for new ways to inform and interact with drivers.

The “big shift” motives are cost-effectiveness, better incident handling procedures, improving customer services by acquiring services rather than systems, creating multiple stakeholder networks, and partnerships with the commercial sector through new funding and financing tools. Also data and information management is an issue that road operators have listed as an important tool for intelligent transport network (Piarc 2004). The Finnish Ministry of Transport and Communications published a report of available transport data sources in 2013 (Liikenne- ja viestintäministeriö 2013).

Authorities like cities and states organise traffic management, public transport and road network-related safety infrastructure and services for the public. One outcome from the MonArkki project was that authorities aspire "to plant" their own services (e.g. eCall, pay-as-you-go charging, information services) onto multi-service platforms. These “official” services must meet the requirements of the authorities, who will seek to support and enable commercial sector efforts to develop an effective Multi-Service Model. Building the procurement procedures to support the value network production, focusing on international cooperation and standardization, and preparing the pilot are the first preparations needed for PPP co-operation. (Salovaara et al. 2012.)

Another notable trend related to traffic services is increasing utilization of Location based services (LBS). As such, LBS have very good spreading potential, now that almost everyone has a smartphone with geographical positioning and the ability to detect POIs, which abound on digital maps. People are using mobile devices to transmit their location to central systems as well as each other, and to receive location-based information. LBS and location-based data are crammed full with opportunities emerging from the new local-mobile paradigm, especially from location-enabled mobile ads and feature-boosting mobile apps.

Local data also have the potential to connect hundreds of thousands of small and medium-sized businesses to the mobile economy (Business Insider 2013). Location-enabled mobile ads have generated excitement for their effectiveness and the impressive prices they command. The simple fact of a user being physically close to a business is seen as an opportunity. The problems seen here are mainly technical, relating to the quality and correctness of location data.

Location-based features have helped both big and small players (Nokia, Apple, TomTom, Facebook, Google etc.) in creating popular applications in their ecosystems. The challenge now is trying to find even more imaginative apps with location-based notifications and location-aware services. The most interesting feature is perhaps direct contact – that is, local data based search enabling the user to connect directly to services. Google highlights that one third of "local mobile consumers’" mobile searches have local intent, and that 94% of smartphone users have searched for local information. In a Google survey, 76% of the respondents said they would like the location or operating hours of businesses, and 61% said they would like to be able to "Click to call" the business. (Business Insider 2013.)

3. Multi-Service Model for ITS

3.1 Description

The Multi-Service Model aims to provide a variety of transport services to the user easily, at best with a single contract and invoice. Services may take advantage of the same basic elements, such as a user identification and tracking, data transfer, payment processing, etc. The consumer can choose the device and with it purchase the services or subscribe to them, just like purchasing a mobile phone with a network provider agreement. Multi-service replaces many separate devices and systems.

From the perspective of the service provider (government and commercial entities), a Multi-Service Model aims at development of services and synergies between stakeholders and the creation of new business opportunities. The target is to form a common public and private services ecosystem, which would function as a network. It is hoped that the relevant authorities will enable multi-service development e.g. by offering open data.

Multi-Service Model thus refers to a service eco-system formed from a variety of intelligent transport services, which can be combined and offered to consumers and business customers in a platform-independent manner.

In this document we aim at specifying the means, a high-level architecture and required functional components, to realize the model. From this point on, we call these means as **Multi-Service Facilitation (MSF)**, in the heart of which lies ICT enabled functional components that enable the realization and supplying of the mobility services as intended in the Multi-Service Model.

3.2 Core value chain

Figure 1. Core value chain in the Multi-Service Facilitation context.

The core value chain for traffic related services (Figure 1) is adapted from the value chain descriptions for traffic information services presented by Kochs and Ansorge (2006) and Herrala (2007). Both sources depict essentially the same value chain but name the actors in a different way. These two examples reflect the confusion and instability in the terminology, and in naming the roles in value chains also on a larger scale. Especially the overloaded term *Service Provider* is used for different kinds of actors depending on the source; we revise this terminology in order to clarify their roles in the value chain.

In this document, we use **Service Provider** as a general term for any party providing any kind of business-to-business (B2B) or business-to-customer (B2C) services, and allocate specific names to the main service provider types. We have added cross-cutting element Multi-Service Facilitation in the value chain (orange block in Figure 1) to emphasize fact that the enablers supporting Multi-Service Model are aimed to support the value chain in its every part. The core value chain in the MSF context comprises (above the End-User) three main kinds of service providers, which together offer a service to the end-user:

Data/Content Provider (or just **Content Provider**) is a B2B service provider that collects and provides traffic-related content for others to utilize in their services. Examples of public and private content providers include (see, e.g. Kochs & Ansorge 2006):

- Traffic information or management centres (TIC) (dynamic traffic data)
- National, regional or local road authorities (dynamic traffic data, parking data, static road network data)
- Commercial traffic data and traffic information provider (like Trafficmaster, DDG etc.)
- Toll system operators
- Parking facilities operators

- Public transport operators (static and dynamic public transport data)
- Automobile clubs (traffic messages)
- Private road operators (dynamic traffic data, static road network data)
- Maintenance operators (road maintenance information, road works etc.)
- Private address and POI data provider (address data and POI data)
- Weather services (weather data and road weather)
- Map agencies (map data for background map, address data)
- Commercial map enterprises (road network data, road maps).

Service aggregator is B2B service provider that aggregates, processes and refines the data coming from data/content providers as well as from other B2B service providers, in order to provide mobility services that can be used as a basis for end-user services. The service aggregator may base its service solely on its own data or content, thus combining the roles at the beginning of the core value chain.

Service supplier is a B2C service provider that packages and distributes the services provided by the service aggregator to targeted end-users. In many cases, the service aggregator may also take the service supplier role, thus being effectively the same actor. Usually the service supplier has also taken care of billing, customer administration or marketing, but here MSF aims to offer support to alleviate the service supplier from at least some of this burden.

End user is the consumer of the service provided by the service supplier. End-user here can mean:

- **Private end users** using the services for their private needs
- **Organizational end users** where employees of the organization (company, authority etc.) use the service.

In the Multi-Service Model this core value chain is supported by **MSF**, which offers functionality for easier and more effective service realization throughout the core value chain. This facilitation may be operated by a single **multi-service operator** or by several co-operating parties taking different sub-roles, thus operating different parts of the facilitation services.

The core value chain cannot realize its services without other enabling roles. Such roles include:

- **Device manufacturers** that provide all kinds of devices ranging from sensors and roadside devices to computer hardware and end-user terminals
- **Software manufacturers** that provide a range of general purpose and dedicated software for data management, computing, visualization etc.
- **Telecom operators** that provide the essential infrastructure and communication capacity for service realization and distribution

- **Vehicle manufacturers** that provide access to vehicle data and integrate the mobility service enabler with the vehicles they manufacture
- **Financial service providers** that facilitate electronic fund transfers on a global scale, enabling, e.g. services by payment and clearing operators.

In addition to technical realization, the core value chain is dependent on other stakeholders that regulate, standardize or otherwise directly or indirectly impact the service realization environment.

3.3 Expected benefits

From the outset, the main motive for multi-service has been to bring different public and commercial services onto the same platform cost-effectively. Public services (road tolling, eCall, insurance etc.) on an in-vehicle platform could enable the wide implementation and penetration of services. If some service or system is mandated to be mounted in vehicles, the market in Finland is about three million units. This is tempting for in-vehicle device manufacturers. Public authorities expect to get these mandatory services for a more reasonable price when they are connected to other services, although numerous definition and certification steps are needed from their side.

The main beneficiaries of multi-service are the end users (they get more services), public authorities (they get their duties, and more, provided by the commercial sector), enterprises and companies (they get new services as B2B but are also creating new services and new business). All beneficiaries will get more by combining both services and cooperation.

Values perceived by suppliers:

- Multi-service brings new business possibilities and assisting services for B2B customers such as companies with mobile work/services (e.g. security services, outdoor ads, and maintenance services), leasing and car rental companies, car parks, insurance companies etc.
- New service business possibilities out of existing technology, new production processes and “sub-processes” if the value-chain is open – possibilities for creating new value-added services in the supply network.
- Multi-service can be developed for export and services made interoperable in different countries, e.g. smart corridors and cities.

More specifically, the value offered to end users (road users) compared to existing service provisioning are:

- Better accessibility to offered services, because the service supplier collects the services for the end user and serves them with a single window (single sign on, centralized accounting etc.) with payment schemes, as the user considers most suitable.

- Single trusted point of managing sensitive end user information (by operator of the Multi-service facilitation) enables easier control for users' own information, and thus enhances users' privacy and security.
- Cost efficiency for service creation for both road users and for enterprise or government customers; as the number of the service users and services provided increase, the supply grows, which usually drops prices. Different pricing schemes can still exist (premium services) but the basic services will be more affordable for all.
- Multi-Service improves everyday mobility and makes trips more sustainable, with better information for route and parking place optimisation etc. With multi-service the end user can optimise driving behaviour and save on petrol and even insurance costs in the future. Multi-service can create automatic driver logs for travel expense reports etc.
- For hauliers and mobile work companies, multi-service can make logistics more effective; the service combines separate transport services and logistics products into a more useful common service offering. The platform can contain both commercial (fleet management, routing, lifecycle, maintenance etc.) and regulated services (e.g. digital tachograph, HGV charges).

For society (governments and cities):

- Multi-service may also provide a platform for regulated services for authorities. It can collect anonymous data for traffic management (if agreed by the end user) and mediate traffic information to road users. It can also be used e.g. in different automated inspections (emissions etc.) and in any new user-charging scheme. With multi-service, no separate systems need to be built (only certification, enforcement and service purchasing remain for the authorities).

In this era of numerous new data sources, the Multi-Service Model could be the solution for an integration platform. Currently no such system exists anywhere in the world, barring some research and studies. For Finland, multi-service innovation for situational awareness (also outside the traffic and transport sector) could be a good opportunity for export.

3.4 Criteria for success

In order to gain solid user base and to realize the benefits exemplified above, the realization of the Multi-Service Model in practice must fulfil certain criteria:

- **Quality, performance and reliability** of the services and the multi-service environment must meet the expectations of the service users.

- **Safety** in the application and service use in traffic is of paramount importance (especially in in-vehicle context). For that reason standards and validation of the services and applications with respect to traffic safety are needed.
- **Common architecture, standards and open interfaces** are essential to allow developers to for service networks easily and create services that can spread easily to the market.
- **Privacy and security** concerns are especially high in the location-based mobility services where users' whereabouts are utilized and tracked. Control of sensitive data, rules governing data security, use and sharing as well as clear agreements between end users and service provider parties are needed to increase trust and user acceptance.

3.5 Business opportunities in the multi-service approach

Business and business networks can be examined based on value chains and the value network mechanism. A business model describes how a company can create, deliver and capture value (Osterwalder & Pigneur 2009). Key functions of business modelling are to identify potential customers and a company's offerings that propose value to customers and in exchange derive profit (value capturing) for the company. To create value, a company has resources and capabilities to offer and it tries to find its competitive advantages. The company's main function is to maximize value creation and return profit by providing product or services that enable customers' ability to create value for them. Since the customer could use several complementary sources to create their own value, co-operation with other companies is an efficient way to enhance customers' value creation and, in turn, willingness to pay for the received value.

In Porter's value chain (Porter 1985), each sequentially linked operator adds value to the product or service to the next operator in the chain. The goal of the value chain is to provide value to the product or service end user, e.g. to a customer. In a value network the value is created also for other participants of the network and the goal is to generate value both to stakeholders of the network and to customers. Each business player has their own agenda (a strategic choice) and positioning for capturing and creating value in a value network.

Since the release of James F. Moore's McKinsey Award-winning article in the Harvard Business Review in 1993 (Moore 1993), it has been trendy to talk about business ecosystems. The fast-evolving information technology field has eagerly adopted this ecological metaphor. Although the term "ecosystem" has become somewhat of a buzzword, the approach is helpful when seeking new perspectives to the study of relationships between companies and business networks.

The old model of competing individual companies in markets is becoming less appropriate to describe the relationships between stakeholders, due to the rapid change of technologies and globalization. The current economy is a complex, interconnected net of stakeholders where each company needs to define its relationships with customers, partners and competitors. According to Iansiti and Levien (2004), the focus is shifting towards management and influence of assets that are not owned and controlled by a single company. As with the case of intelligent traffic services (ITS) where the whole business is complex, networked and evolving fast, this ecosystem perspective makes sense.

Ecosystems emphasis relationships and permit companies to create values that no company could achieve alone. Moore (2005) emphasis aligned visions in such a way that “development investments are mutually supportive” and “capital investments and operating processes are synergistic”. Ecosystem can be seen as a collection of business concepts with a common core that consists of platforms, technologies, processes, standards or knowledge.

The Multi-Service Model proposed by Vilkman et al. (2010) could be interpreted as an ecosystem type business concept with a common goal (e.g. provisioning of wide range traffic related services more accessible and economically) by utilizing sophisticated agreements, certification procedures and procurement processes between public authorities, companies and sub-contractors in different countries and communities. In this concept the multiservice facilitation (MSF, “a multiservice platform”) forms an electronic marketplace, where suppliers can offer content and services via well-defined interfaces directly to end-users (B2C) or other companies (B2B) that can assemble and refine new services from existing ones.

As stated in Section 3.2 the participants in the multi-service ecosystem are data and/or content provider, content aggregator, multi-service operator, service provider and end users. Other related roles are device manufacturers, map providers, mobile network operators and all different transport network infrastructure and maintenance providers and service operators (like public transport operators).

Multi-service operator has the key role (“a keystone species”) for maintaining the multiservice ecosystem. The service operator maintains and offers the whole service platform, creates the agreement models, customer registers, and mediates 3rd party services to customers. The multi-service operator is the activator for growth and expansion of service delivery making the services interoperable and more useful for the customers.

Multi-service consortiums might evolve “naturally”. Clear competitors form a cluster of their own and there are a limited number of partners in one consortium – but it is good to promote cooperation between organisations and multi-services (which can be done with standards and a common B2B market).

The open common platform approach, together with a user community, is one of the reasons behind the success of today’s smartphones. This also true in the navigator business and in vehicle industry ecosystems. The current business model is to spread the business from device manufacturing to service provisioning. The motivation is to keep the customer interested beyond the one-time purchase of a device like a smartphone, a navigator and/or a car. Having a permanent provider-customer relationship throughout the lifespan of the device provides more payments for the service delivery.

A multi-service platform needs one or two key companies as activators. The natural role for the activator would be the multi-service operator. Key companies must also already have customers with whom the first services can be developed. All the chain partners need to have strong professional products or competences of their own already.

Also needed is a company or partner in the consortium that has know-how of the relevant domain of the target multi-service (e.g. in transport: transport systems, fleet management, public transport and/or telematics etc.). This helps toward meeting the necessary requirements and maintaining dialogue with the related authorities and customers. Fee distribution and revenue sharing constitute the main challenge to multi-service creation. Services for end users should also be beneficial in terms of costs: new services introduced to end-users should be very affordable and there should be even free-of-cost trials. Service providers need their costs to be covered and profit to be made for their business and development. In between there is the multi-service operator or multi-service broker, who is creating the service portfolio combining 3rd party services and own services with marketing efforts, back-office and call-centre services.

3.5.1 Example: TomTom ecosystem

Figure 2 shows the TomTom value network. The TomTom navigation device is one type of multi-service platform. TomTom sells navigation devices and related services (e.g. navigation software, maps, POI information and real-time information) to its customers. In return, TomTom collects sensor data from the users (through devices with a SIM card), refines the raw data, and sells it as floating car data (brand “TomTom HD”) to road authorities. The data also helps TomTom update its digital maps and route databases (to maintain good quality) and to provide dynamic information to the end user. In this model, TomTom has also an agreement with Mobile Operator which can provide also anonymised data for the TomTom back-end system.

TomTom’s main incomes come from:

- sales of the devices,
- customer support and service updates, and
- selling the HD data for public sector and other 3. parties.

The main costs result from:

- producing and supplying the devices,
- buying data from other sources, and
- buying subcontracting services (marketing, accounting, banking etc.).

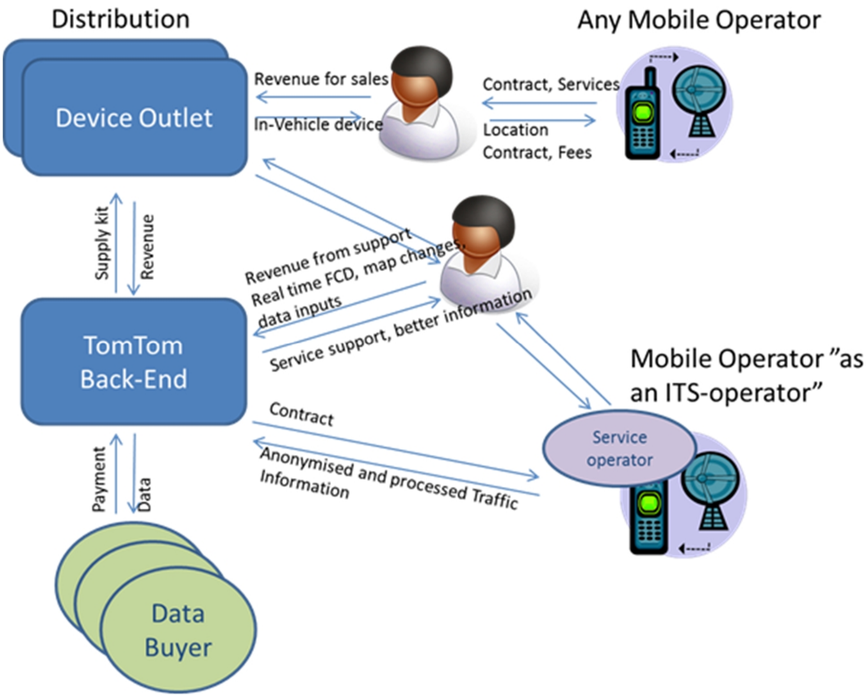

Figure 2. TomTom value web (Stevens 2012).

Other related cost/revenue flows are e.g. following:

- fees for communication from user to mobile operator, and
- possibility for the mobile operator to gather location data from the TomTom SIM card for an own anonymous Traffic data base (which can be sold etc.).

As a new service, TomTom launched a Pay-As-You-Go insurance service in UK in the beginning of 2012. Insurance companies are eager to offer new products but they need a lot of user data to create these products which are based on totally new insurance calculations. Pay-As-You-Go payments are usually based on the driver behaviour and driven kilometers, which are analysed afterwards. The insurance companies want good safety impacts with new products; the challenge is to create a balanced post-payment which does not drop too much the total of insurance payments which nowadays base on prepayments.

TomTom has created also alliances with vehicle industry (e.g. brand Tomtom Blue&ME with Fiat). TomTom owns the other main commercial digital map provider, Teleatlas.

In the model above, TomTom also has an agreement with the mobile operator, which can provide also anonymised data for the TomTom back-end system.

3.5.2 Finnish examples

The “multi-service” business case of Nokia is almost similar to that of TomTom. Smartphones create a bigger ecosystem than PNDs, but the services and customers are basically the same. The main incomes are from selling devices, but currently Nokia has put a lot of effort into selling mobility and location related multi-services for end users and traffic related value-added situational data for road authorities. Nokia has also launched public transport services similar to those provided by Google. Mobility-related apps are relatively cheap for smartphones but are sold in large numbers. As with TomTom, Nokia is active in creating alliances with the vehicle industry: the Car Connectivity Consortium (Mirror Link brand) was a Nokia initiative. As always, the main backbone of these services is digital maps; Nokia owns the other leading commercial digital map provider, Navteq. The new name for Nokia Maps is HERE, and it can be used as free web service.

Some other examples from Finland include Helpten. The main product is in-vehicle device providing a driver logbook with additional features like driving behaviour, navigation, calendar etc. The service products are based on a monthly fee, with basic logbook and vehicle monitoring costing under €30 per month. The device and installation costs are included in the monthly fee. The logbook with navigation and real-time communication costs about €47 per month.

The Finnish state procuration enterprise Hansel has signed a 4-year contract (from January 2013) with E-Bros, which provides in-vehicle devices for state owned vehicles. The system contains service contracts, in-vehicle devices, installation, mobile communications, web-service, customer registration and care, updating services, reporting, and user training. Hansel was especially pleased with the clarity of the contract, which explains what the service contains and what it costs per vehicle per month. The services are similar to Helpten: logbooks, driving behaviour (e.g. fuel consumption), vehicle-booking calendars, etc., depending on what the vehicle owner decides. Pricing is based on vehicle fleet size. For example, for a fleet of 20 vehicles prices can be under €30 per vehicle per month for basic services (logbook and vehicle monitoring). If the navigator is included, the fee is €44 per vehicle per month.

3.5.3 Data market places

In Europe a number of real-time market data information projects have been launched to collect data and processing services. For example, the Swedish Info24 (Info24 2013) advertises itself as an easy, safe channel to provide timely information, such as alarms, statistics, places, applications, etc., especially for businesses. Info24 is a diversified commercial information market. The idea is to convey everything from machine to machine automatically. Market data provides data transmission in addition to device management, financial reports, payment services, interfaces, secure, and flexible third-party applications for Internet and mobile phones. The website has featured a lot of different transport-related services and competitive tendering for new ones. It is used also by the City of Stockholm and Swedish transport agency for getting data for new services.

In the Netherlands, the national traffic information market (National Warehouse for Traffic Information) was guided by the national authorities and funded by development projects. It collects real-time traffic information from the Dutch road network and distributes it to different parties for further processing. Also road works, congestion data, accidents and other incident information is gathered and available. (NDW 2013.) In Germany, the Mobility Data Marketplace (MDM 2013), has been under development since 2007 and should be operational in 2013. MDM specifically examines business models, barriers and challenges to creating the platform, as well as data quality. Data is collected from the authorities and from the private sector (MDM 2013). These big projects under the direction of the national government have made very slow progress.

Data market places where marketplace operators collect data from different sources in order to act as a broker of aggregated data realize a model that is used also in wider context. For example global business brokers Ariba (2013) and Alibaba (2013) act as a broker between sellers and buyers of any products and services – not just traffic data.

Essential requirements for B2B information market places are such that those should be marketed widely among relevant stakeholders (i.e. it should be found easily) and should have a register of included services and contact information of sources. The register should contain a clear description of what data is available and a search function for data providers including data on boundary conditions (location, type etc.). A ready-made fast or automated contract template for a negotiation convergence service is also desirable. Easy documentation tools and secured data transmissions would be needed. Additional services like an active way of finding further innovations and new customers, and evaluating what contents, data or services are missing, and foresight as to where to go next could be supported.

3.5.4 Future trends

Richard Cornish from Vodafone (ITS international 2013) has confirmed the mobile operator's current interest in obtaining a share of growing revenues in the ITS scene. He lists key issues related to end users (service buyers) now in intelligent services:

- People desire to bring their personal, cloud-based social interactions, data and applications into their car and mobility
- Car ownership will diminish and other services (car-sharing, rental, car-clubbing etc.) will emerge
- With the above, also the costs of fuel and insurance will affect car ownership
- Environmental awareness and urbanization will increase

- The future is for smarter, multimodal travel patterns – which means combining different modes: car, public transport, bike, plane, train.

As a result, also services must be focused in a new way to serve the whole chain of travel and personalised needs. Vodafone is tackling the privacy issue with a strategy of “bonding the trust”, which means encouraging consumers to consent to being tracked in return for their journeys being made safer and their vehicle more secure (e.g. with stolen vehicle tracking). Vodafone understands the vital importance of public services like eCall, road tolling and usage-based insurance, but also services connected to safety and security, vehicle diagnostics, infotainment, mobile working, driver assistance and navigation and real-time traffic information. (ITS international 2013.)

In the flourishing multi-service market, individual data or content providers have many opportunities to sell their products and data to multi-service ecosystems or service aggregators in various formats (as raw data or in more processed form).

The service aggregator can create new contents combining datasets from different providers depending on customer demand. This content-creation can and should be made in cooperation with data providers and data users.

From the perspective of the service provider (government and commercial entities), a Multi-Service Model aims at developing services and synergies between stakeholders and creating new business opportunities. The target is a common ecosystem encompassing public and private services in a functioning network. It is hoped that the relevant authorities will enable multi-service development, for example by offering open data.

4. Related work

Chapter 4 looks at some interesting examples relating to the Multi-Service Model. The 511 service in the USA is a good example of a service that initially needed funding and activation from the authorities, but is now developing in many ways and with many stakeholders as a truly multi-modal service. In Japan, the most important transport systems and services are publicly activated, funded and run. In Europe, ITS architectures have been developed mainly in relation to public services and for traffic management by road operators. The Frame architecture covers a lot more, which is one of its drawbacks because it has become so massive that only parts of it are in use. Datex II is a successful standard for data delivery between different road operators. Instant Mobility, In-Time, and Mobinet are EU-funded research projects that have developed interesting architectures. In Finland, TelemArk, LihArk and TosiArkki were Finnish telematics architectures, and MonArkki was a first attempt of the Finnish Transport Agency to study multi-service architecture from the public-private partnership point of view.

Automotive industry has also developed value-added service offerings for traffic safety and comfort in the form of In-Vehicle Infotainment (IVI) systems that are directly competing with “standardized ITS services” and can be seen as a sort of multiservice approach. These systems are not examined in further detail here but interested readers could take look to the Appendix D for commercial multi-service offerings, where a brief introduction to IVI systems is given.

4.1 International efforts

4.1.1 The 511 service

511 is the US Federal Communications Commission’s (FCC) nationwide three-digit telephone number for traveller information that was initially designated for road weather information. Before the advent of 511 there were over 300 travel information telephone numbers for real-time traffic data. In 1999 the US Department of Transportation petitioned the FCC for a three-digit dialling code to make it easier for consumers to access these travel information services.

511, now also provided in Canada in addition to USA, is a successful example of a public-private partnership arrangement. States have the lead role in coordinating 511 deployments and national leadership is provided by the 511 Deployment Coalition. The Deployment Coalition includes more than 30 public agencies, industry groups, industry associations, and private companies and is led by the American Association of State Highway and Transportation Officials (AASHTO). Other leading member organizations include the American Public Transportation Association (APTA), the Intelligent Transportation Society of America (ITS America), and the US Department of Transportation.

The most advanced 511 deployment is currently under development in San Francisco. The Metropolitan Transportation Commission (MTC) is responsible for this project and funds the 511 telephone number, the 511.org Web page, and all other elements of the project. Caltrans is another major partner in the 511 project and provides information from its embedded detector system and by allowing MTC to install FasTrak readers along Bay Area freeways. Additional radar detector data is purchased from a private company to complete the traffic congestion information.

FasTrak is the electronic toll collection system used in California (state-wide). The system developed by MTC uses small antennas over freeways to read FasTrak toll tags usually installed on car windshields. When a car with a FasTrak toll tag drives by an antenna, the system detects the presence of the tag.

SFO 511 offers web and mobile accessible services in the Greater San Francisco Bay Area. SFO 511 Traffic services provide real-time information to help drivers plan their trips in the Bay Area, help commuters avoid traffic jams and incidents on the roads, and keep traffic moving to reduce congestion in the region. The web-based Traffic Page offers:

- An interactive map that displays real-time traffic congestion information, traffic incidents, construction projects, events, and live-streaming traffic cameras
- Real-time and predicted traffic information that updates automatically
- Current driving times and driving times for a specific time and day of the week based upon historical data
- Information about breaking traffic news and construction work that may affect the commute that displays up-to-date traffic incident, construction and event information
- Personalization, via a Recent Trips box that stores and displays recently requested trips, and customizable map preferences
- Information on tolls, express lanes, high-occupancy vehicle (HOV) lanes, park & ride lots
- FasTrak information where customers can find more information about FasTrak, access their FasTrak account, or order a FasTrak toll tag.

Mobile accessible versions of these services are under construction and can be accessed in the future by using an m.511.org web site. The current 511 Traffic page is not optimized for mobile devices or smartphones, but depending on the device many features may be available to customers. The Traffic map is currently available on the 511 Mobile site, m.511.org.

Driving time predictions and other traffic information provided by the 511 system are extremely accurate. According to user feedback, the 511 Traffic page is more reliable than other online tools and the system undergoes regular performance testing in order to identify any possible problems.

MTC and Caltrans both guarantee that FasTrak users will remain anonymous. The 511 system scrambles the toll tag's identification number to protect personal privacy. Encryption software encrypts each FasTrak toll tag ID number before any other processing occurs, and these IDs are discarded every day; the encryption code is changed every day and a historical database of encrypted IDs is not maintained.

The SFO 511 system does not have a publicly defined architecture but provides several open APIs and data feeds that can be used as a base for advanced services free of charge.

The free 511 Traffic Data Feed provides incidents, speed, and travel time data for individual segments on the road network in the San Francisco Bay Area. Registered ISPs will be able to receive XML-formatted data via the Java Message Service (JMS) over the Internet.

The free 511 Transit Static Data Feed provides schedules, stops, routes, fares, and other information for over 35 transit agencies/providers serving the Bay Area. The Transit Data Feed is provided in either CSV (Comma-Separated Values) or GTFS (General Transit Feed Specification) format.

The 511 RideMatch Service is an interactive, on-demand system that helps people find carpools, vanpools, or bicycle partners. The service can be customized to match the look and feel of a company’s website. This allows employees to access RideMatch with the added confidence of knowing that their employer supports ridesharing. Registrants have the option to limit their match results to company-only employees, or to expand their search to all regional matches.

The 511 Driving Times API provides 511's current and typical driving times between a starting and end point, including incidents along the route. The API consists of a collection of individual services implemented using simple web pages that accept query-string parameters and return plain XML documents.

The 511 Real-time Transit Departures API provides real-time departure predictions for several regional transit agencies. The API consists of a collection of individual services implemented using simple web pages that accept query-string parameters and return plain XML documents.

4.1.2 Japanese multi-service framework

Japan, having a remarkable population density compared to other developed countries, has an abiding interest in developing ITS systems to resolve its traffic related problems. The aims of Japanese ITS efforts are equivalent to European goals e.g.:

- To reduce traffic congestion time losses in Japan (5 billion hours annually),
- To reduce traffic accidents (730,000 accidents/year causing over 4,800 fatalities)
- To diminish environmental degradation (20% of all CO_2 emissions are due to traffic).

The practical coordination of these efforts is handled by several ministries, but the leading role is devoted to the Ministry of Land, Infrastructure, Transport and Tourism (MLIT). MLIT has released a comprehensive plan that includes several special development areas including:

- Advanced navigation systems
- Electronic toll collection systems
- Assistance for safe driving
- Increasing efficiency in road management
- Support for public transport.

To collect and transmit real-time traffic information, Japanese authorities launched a research project in 1996 and created the world's first vehicle information communications system (VICS), which has been available nationwide since 2003. Today over 39 million vehicles use the electronic tolling system (8% of the vehicle base) and 35 million vehicles use VICS. The latter was further developed in 2011 to offer new services including dynamic route guidance, support for driving safety and an electronic toll collecting system (ETC). Thanks to these systems, annual CO_2 emissions have been cut by over 2 million tons and almost all toll-gate congestion on expressways (30% of all traffic jams) has been eliminated.

The current Japanese system is based on public-private partnerships where the basic services (dynamic route info, safety and ETC) are provided by the road administration and additional services (currently local sightseeing information but others are planned) are offered by private companies. JARTIC (Japan Road Traffic Center) collects, organizes, and analyses constantly changing road traffic information in real time, online with each management organization and provides road traffic information to road users through radio, TV, cellular phone, car navigation systems and other media.

Japan's total road length is about 1.26 million km and information concerning it is acquired and provided through cooperation between road managers and the private sector. For creation of a real-time traffic situation snapshot, data is assembled using a fixed ITS Spot infrastructure, and for quality comparison purposes also from private sources. The co-operative ITS Spot system (see Figure 3) based on 5.8 GHz dedicated short-range communication (DSRC) has been in operation since 2009. Sixteen device manufacturers, including automobile and specialized navigation system and on-board unit manufacturers, have announced Spot- compatible on-board units that are able to communicate roadside infrastructure. The Spot infrastructure contains over 1,600 fixed probe stations installed at 10 to 15 km intervals along inter-city expressways and 4 km intervals along inner-city expressways. It is estimated that over 10 million Spot on-board units (OBUs) will be sold in the next 5 years (starting in 2009) as a car navigation unit.

The data collected by OBU includes basic travel data (time, location and speed) as well as behavioural data like accelerations when a force of 0.25 G is exceeded. An OBU is able to record events over an 80 km segment if those are not transmitted earlier to a probe server and further to the road administration via intranet. The information collected could be used in addition to basic services (dynamic routing, danger warning and ETC) to identify sudden braking points and take preventive action to improve the traffic environment. These actions include improved road marking and treating the roadside environment to improve visibility. Probe data is privacy protected; neither vehicles nor drivers can be identified and drivers voluntarily transmit the probe data (i.e. can opt out of sending probe data). The purpose of use is clearly described in the OBU manuals and on the MLIT web sites.

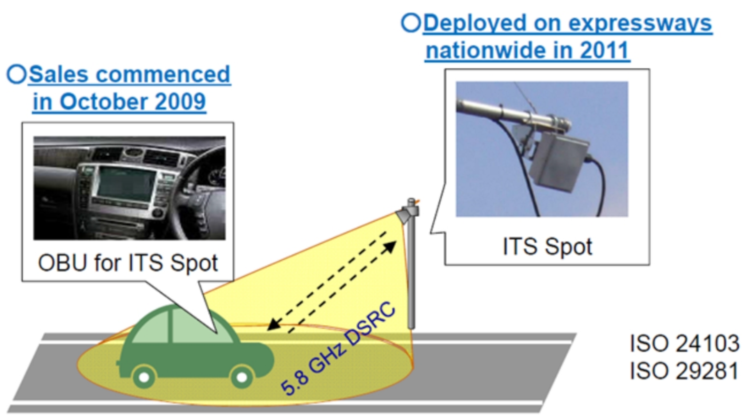

Figure 3. Japan's ITS Spot system (MLIT 2012).

The ITS Spot based system is currently (since 2011) able to:

- Select the optimal route based on the dynamic traffic situation, thus the whole road network can be used effectively. Part of the routing service provides information on passable roads during/after natural disasters.
- Alert drivers in advance when unexpected events occur like congestion, on-road obstacles, snow coverage on the road, and emergency information concerning natural hazards like heavy rains, earthquakes and tsunamis
- Provide electronic toll collection (ETC) services.

New applications for the Spot infrastructure are under development, including support for logistics, car-navigation-system based service provisioning, and drive-through payment (test case a purchase at McDonalds) etc. The main aims for the future are to share large quantities of data by effectively using both private and public data and to create end-user value and new ITS services. As a result of these Japanese experiments, the multi-service framework is manifesting concretely as co-operation between the authorities and the private sector. The embodiment of the system is a “Spot infrastructure compatible” car navigator unit (OBU) offering a traffic-context-compatible user interface for advanced traffic services.

4.2 European efforts

4.2.1 FRAME

The European Intelligent Transport System (ITS) Framework Architecture, often now known as The FRAME Architecture (FRAME 2013). It was created to provide a minimum stable framework necessary for the deployment of integrated and inter-operable ITS within the European Union. FRAME was originally created by the EC funded project KAREN (1998–2000) and first published in October 2000 and then further developed to its current form (by revising the original framework and adding new elements to it) in several other EC funded projects.

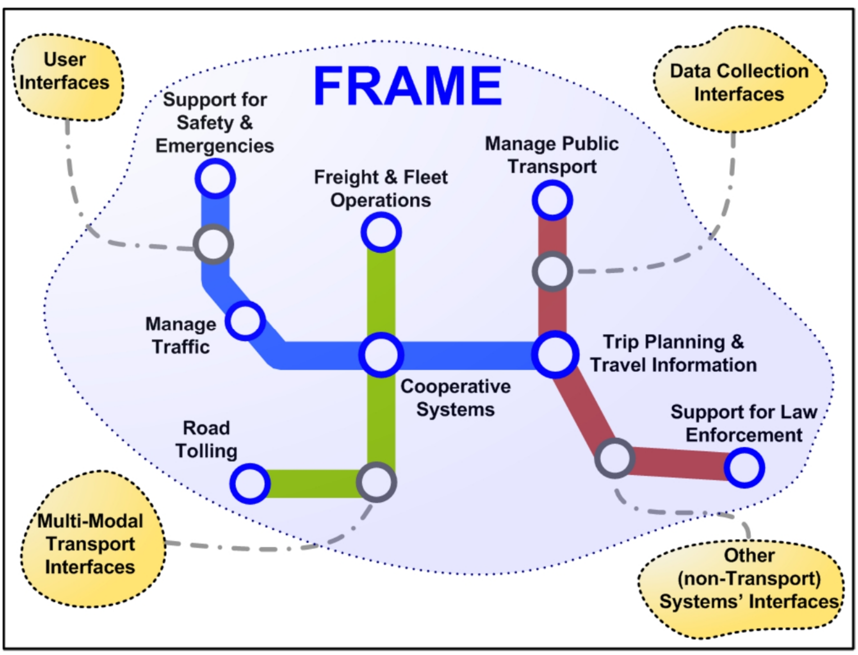

Figure 4. Functional areas in FRAME (FRAME 2013).

FRAME is not really an architecture but architecture development framework that comprises the top level requirements and functionality (use cases) for almost all the ITS applications and services that have been considered for implementation somewhere in the European Union (see Figure 4). It is indented to provide a reference for all ITS architects in ITS architecture realization and guarantee compliance at the interfaces between different European ITS systems and services so that seamless services can be provided to cross-border travellers, and an open European market of compatible components can be established.

The FRAME Architecture was created to provide a common approach, or "language", for use throughout the EU so that the implementation of integrated and inter-operable ITS can be planned. It does not mandate any physical or organisational structures on its users. Hence the FRAME Architecture makes no assumptions about the way that things are done. FRAME also doesn't require the implementation of the whole functionality but allows the creation of logically consistent sub-sets, which can then be used on their own. The methodology is supported by computer-based tools, and begins with the wishes, or aspirations, of the various stakeholders for ITS applications and services. These are identified within the FRAME Architecture and a sub-set is selected. The sub-set is then customised to fit the region in which they are to be deployed (see Figure 5). (FRAME 2013.)

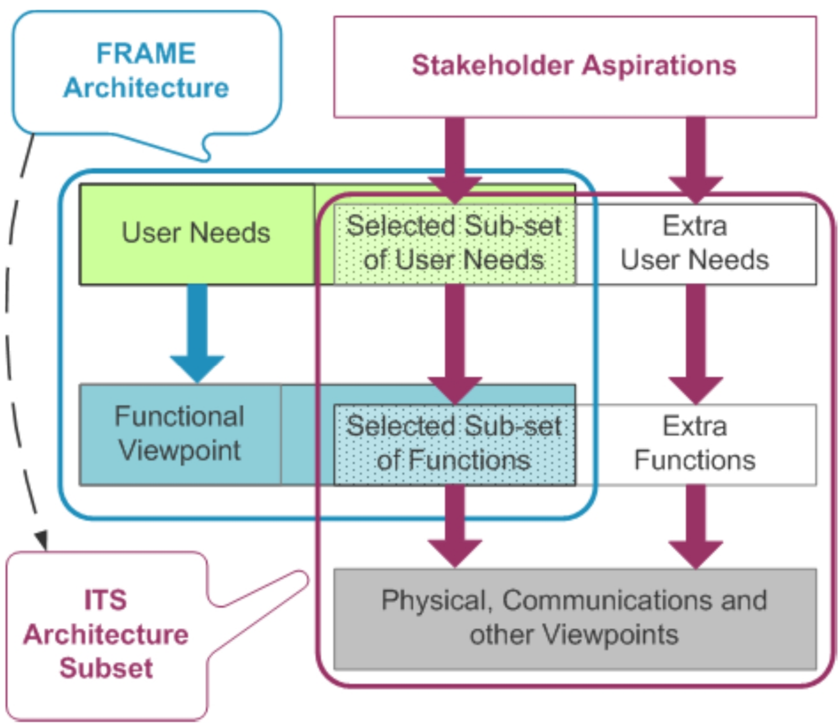

Figure 5. The process of creating an ITS Architecture Sub-set with FRAME (FRAME 2013).

4.2.2 Datex II

The DATEX II related standard CEN TS 16157 (published in 2011) was developed for transport information exchange. DATEX II aimed to provide a standardized way of communicating and exchanging traffic information between traffic centres, service providers, traffic operators, and media partners. The specification provides for a harmonized way of exchanging data across boundaries, at system level, to enable better management of the European road network.

DATEX II will play a strong role in the implementation of integrated ITS in Europe. It is supported by the European Commission; the initiative behind it was to cover European trunk roads, but it has gradually been extended to the urban and logistics domains. The contents of the standard are level of services of the network (in what type of situations information messages should be sent), travel times, all types of incidents and accidents, road works, road infrastructure status, closures, obstructions, road weather and the current setting of variable message signs.

Datex II uses the OMG Model Driven Architecture (Object Management Group 2013a) and the modelling approach is based on the Unified Modelling Language (Object Management Group 2013b), which provides an ideal environment to capture the DATEX II domain model. The current implementation platform for messaging is the W3C standard for XML schema definition (The World Wide Web Consortium 2013). The mapping is defined in the specifications and has been implemented in a tool that users can download (together with the model itself, the whole specification and further supporting material) from www.datex2.eu. (Datex 2013.)

4.2.3 Instant Mobility

The “Multimodality for People and Goods: the Instant Mobility” project is part of the FI-PPP initiative, co-funded by the European Commission’s 7th Framework Programme. The project was led by Thales and the participants included Volvo, CRF, ATAC, CEA DHL, DL etc. and several tele-operators including Orange and Telefonica. The budget was €8M, of which the European Commission provided €5M. The project started in April 2011 and lasted for 2 years (Instant Mobility 2011).

The aim of the project was to develop sustainable transportation practices with a dedicated focus on sharing modalities of all kinds of vehicles and providing new ways to optimize urban traffic. The project developed and explored a concept for transforming the mobility of persons and goods in the future through application of advanced Internet technologies. It introduced five scenarios that utilize next-generation Internet capabilities to deliver online mobility services including:

- The Multi-Modal Travel Made Easy service offers a wide range of travel and transport options, based on user preferences, for all stages of a trip that may use various modes including public transport, car and non-motorized means.

- Sustainable Car offers travellers the best route service with the least delay, least CO_2, shortest travel time, lowest cost etc. via a number of interactive online information services based on future Internet technologies.
- Collective Transport 2.0 expresses a vision where transport operators of the future will use Internet to detect the presence of passengers at stops and to register their destination, offer innovative online services flexibly matching vehicles, timetables and routes to actual demand.
- Trucks and the City Service shows how Internet can be used to manage commercial vehicle deliveries and routing, organize drivers' shifts and synchronize vehicle movements and goods pickup and reception.
- Online traffic and infrastructure management puts traffic management and control online, offering greater flexibility and performance of the current functions of today's traffic management.

These conceptual services re-group previous services into three use case scenarios: personal travel companion, smart city logistics operations, and transport infrastructure as a service that are further divided into several more approachable use cases.

The Instant Mobility project used Service-Orientation (SO) as paradigm to define the architecture that supports these scenarios and use cases. The main principle of the Service Oriented Architecture (SOA) is to enable service composition, e.g. services that can be re-used by other services that can create a hierarchical services structure that gains in abstraction by enriching a service with less abstract services.

The architecture document (Beckman et al. 2011) represents both functional and technical architectures for the Instant mobility platform. The functional architecture in Figure 6 depicts the functional composition of the system.

The architecture contains following elements:

- Service layer, which is responsible for domain-specific atomic services (floating car data, tracking of goods etc.) and core services like data storage & computing
- Business process layer, which includes Instant Mobility applications
- Presentation layer, which offers access to the services from mobile devices, web interfaces, etc.
- Auxiliary services like administration are also needed for maintaining the SOA based system (middleware).

The technical architecture has pretty much the same layers and the services are tied together with a service bus that is a software architecture model used for designing and implementing the interaction and communication between mutually interacting software applications in the SOA.

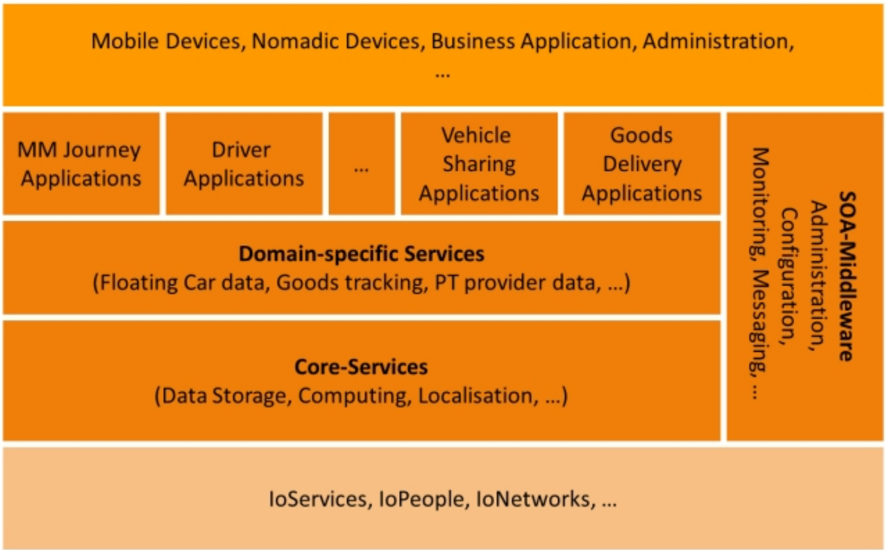

Figure 6. Functional architecture produced by the Instant Mobility project (Instant Mobility 2013).

4.2.4 In-Time

The In-Time project (In-Time 2013, Böhm & Lamprecht 2012) was based on the observation that increased traffic in Europe's cities has resulted in chronic congestion, causing delays and pollution and a yearly loss of nearly €100B (1% of the EU's GDP). Urban traffic is responsible for 40% of CO_2 emissions and 70% of emissions of other pollutants arise from road transport The main aim of the In-Time project was to provide Real Time Traffic and Travel Information (RTTI) services to drivers and travellers in Europe with the goal of drastically reducing energy consumption and emissions in urban transport.

The project, led by Austria Tech, started on 1st April 2009 with duration of 36 months and a budget of €4.58M. The contribution of the European Commission was €2.29M. In-Time was a continuation of a series of earlier projects, most notably eMotion (which ended in 2007), which influenced the In-Time architecture and infrastructure specifications.

The main aids offered in the In-Time environment were Pre-Trip and On-Trip Information services. Web-based interoperable and intermodal pre-trip and on-trip information would offer optimized journeys for the traveller, who would be informed about the different transport possibilities – including travelling modes. Other services would offer complete travel information (public and individual traffic) combined and interpreted at one location – the Interoperable and Multimodal In-Time Regional Data/Service Server (RDSS) operated by a regional operator. The service domains covered within In-Time were:

- Road traffic
- Parking
- Points of Interest
- Public Transport
- Weather
- Multimodal Journey Planning.

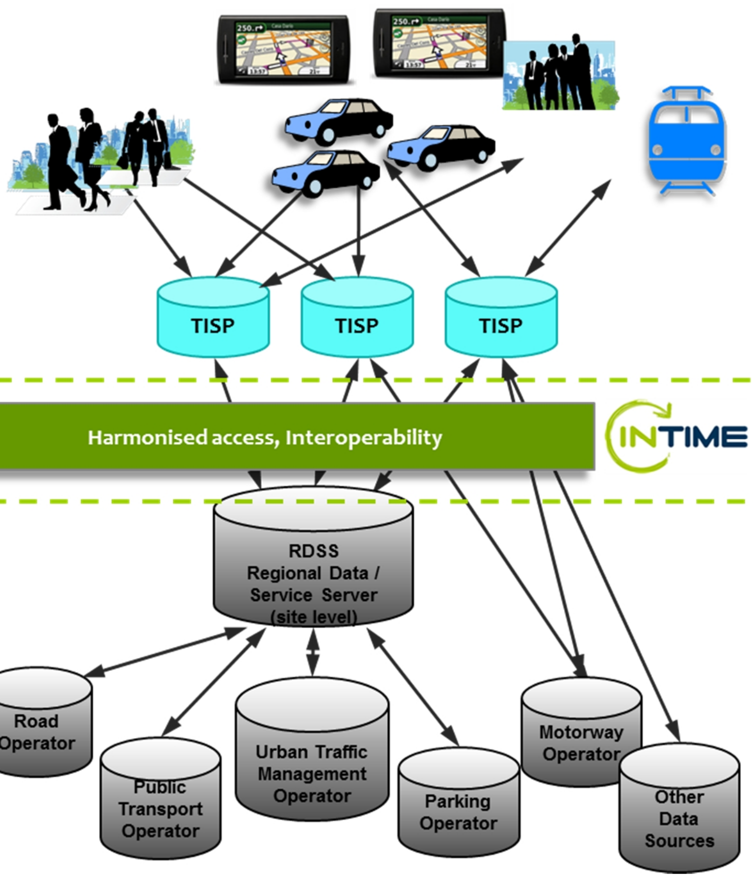

Figure 7. In-Time system architecture (In-Time 2013).

The In-Time architecture is represented in Figure 7. The architectural goals of the In-Time platform enabling In-Time services were to ensure harmonized access to data and services, use of EU ITS standards, bundling all transport info concerning one city, and enabling interoperability of end-user applications and local data/service resources (In-Time 2010).

These goals were achieved using a Commonly Agreed Business to Business (B2B) Interface (CAI) between local systems/services and the Traffic Information

Service Providers (TISP) system with a common data/service model. Data adapters were used to convert individual data sources into the common data model. The In-Time data model was based on several international and European standards like DATEX 2, TPEG, IFOPT, SIRI, JourneyWeb, OpenLS, ISO 19000 / OGC standards and ISO 19100 Geographic Information Standards.

The In-Time Service Model follows W3C (World Wide Web Consortium) and OGC (Open Geospatial Consortium) guidelines. Data Access Service is in accordance with Web Feature Service (WFS) definitions and map services conform to the Web Map Service (WMS). The model is also compliant with EU SDI initiatives, i.e. INSPIRE GML, ISO 19136 and others wherever possible.

The In-Time value chain contains four main roles: source data providers, integrators (Regional Data/Service Service e.g. RDSS providers) that use source data as a basis for value-added services, ITS operators like TISP that distribute the traffic information service to end users (ITS operators' customers). In that sense, the In-Time architecture resembles the Multi-Service Model by enabling multiple ITS-related source data providers and multiple ITS operators (TISP).

4.2.5 MOBiNET

MOBiNET is a new EU project launched to develop an 'Internet of mobility' and simplify the deployment of intelligent transport services in Europe. Over the next three and a half years, the €15.6 million MOBiNET project (MOBiNET 2013a) plans to capitalise on the widespread growth in smartphones, mobile data services, and cloud-based computing to launch a new generation of travel apps for European citizens, and transport services for businesses and local authorities.

The technology vision of the project sees a European-wide distributed, virtual platform supporting a vast network of mobility and transport services. At its heart is a (real, not virtual) multi-vendor E-Marketplace featuring a directory describing the products offered by and to all providers of mobility-related services, a mechanism for matching service requests and offers, and a group of utilities for the discovery, creation, orchestration and operation of services. (MOBiNET 2013b.)

The MOBiNET project targets the following problem areas (MOBiNET 2013b):

- Drivers and travellers need significant effort to discover, use and pay for services, then must subscribe separately to each one they need;
- There are very few Europe-wide travel assistance services for end-users, while locally-based services are inconsistent and of variable quality;
- Europe does not have a unified system for electronic ticketing and mobile payment for mobility services, causing extra delays for transport operations and inconvenience for occasional travellers;
- For cooperative systems, roadside and back-office application interfaces are not standardised, holding back deployment;

- Proprietary connected vehicle services are channelled through closed hardware and application software, making them expensive to deploy and difficult to extend and enhance;
- Business-to-business contracting requires one-by-one negotiation, making it hard to extend services beyond the local area.

The MOBiNET concept is based on the three main pillars (MOBiNET 2013b):

- A global multi-vendor B2B marketplace and service directory where the different kind of service providers can publish and exchange their products in order to compose new mobility services.
- A uniform middleware environment making each MOBiNET-enabled user device accessible for any service provider while enabling the use of MOBiNET app store and other features.
- A service factory that provides development kits and examples for the MOBiNET service developers.

The MOBiNET ideas resemble closely the ones in the Multi-Service Model developed in the PASTORI and SUNTIO projects (see Figure 8). The emphasis of the MOBiNET project lies on the solutions supporting the creation and operation of the B2B service providers' ecosystem. This largely immature topic area was found to be one of the main future research topics also in the SUNTIO2 project. Therefore, the participation of VTT as a full partner in the MOBiNET project and many other SUNTIO-partners as associate partners gives a great opportunity to continue to build on the already achieved results.

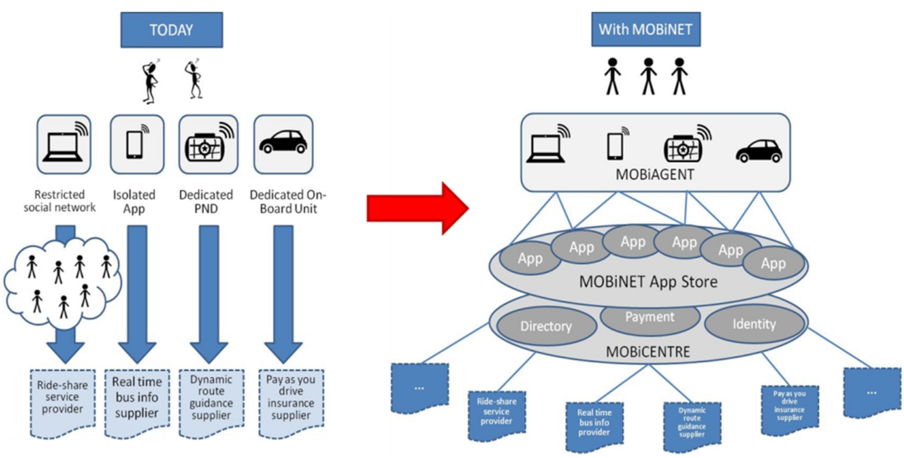

Figure 8. The core idea of the MOBiNET project (adapted from MOBiNET 2013b).

4.3 Finnish efforts

4.3.1 TelemArk

The Finnish national ITS project TelemArk (1998–2000) produced two high-level architectures split into conceptual and logical level architectures. The conceptual architecture lists existing systems, stakeholders, process descriptions including process components and data flows, and a refined list of architectural demands. The processes identified in the TelemArk architecture (Mäkinen et al. 2000) are comparable with other ITS architecture and compatible with the European Union's Karen/FRAME architecture, although the modelling perspective is more commercially oriented. The logical level architecture describes a structure that fulfils all the requirements defined and modelled in the conceptual architecture, including the functions and information content (datasets).

At the conceptual level, TelemArk identifies processes classified in accordance with the division given by the ISO project "Steering of the Development of Transport Telematics". These are:

- Traffic information services: alternative transportation, traffic flow, incident and road works, weather, parking and public transportation information
- Customer demand management: park & ride, demand-responsive public transportation, payment systems etc.
- Traffic management: signal controlling, local warnings etc.
- Vehicle and transport management: hazardous goods, road maintenance, incident management etc.
- Driver assistance systems: dynamic speed control, route guidance, emergency services
- Enforcement systems: speed limits, transport weight monitoring, lane enforcement etc.

The TelemArk architecture is very high level and not intended to be comprehensive, thus more detailed domain-specific architectures have been defined in separate projects. Two of these are LihArk (Tiehallinto 2005) and TosiArkki (Siponen et al. 2005) where the former is related to traffic management and latter is more oriented to the provision of real-time transport information.

4.3.2 LihArk and TosiArkki

LihArk's (Traffic Management Architecture) objectives were to describe the present and future state of traffic management in Finland, and to identify development needs in order to achieve the target traffic management status in 2007. The system architecture presents the functionality and connections between various traffic management sub-systems like traffic monitoring, receiving and registration of incident data, variable message signs and data delivery. The traffic management architecture is divided into seven subsystem elements:

- Traffic flow monitoring
- Weather and road condition monitoring
- Provision of information on traffic disturbances
- Traffic information management
- Traffic announcements
- Traffic information distribution
- Traffic management.

For each of these subsystems, more precise layered system architectures are described with high-level functional interface definitions. The general structure of the traffic flow monitoring and weather data condition monitoring subsystems is common to both cases and illustrated in Figure 9.

In Lihark's architecture report (Tiehallinto 2005), interfaces between layers have been explicitly described where possible. The report concludes that the important future goals of traffic flow monitoring are to produce information about travel times and automatic traffic disturbance detection. In these areas, Floating Car Data (FCD) data might have a focal role in the future when enough high quality FCD (or cellular FCD) becomes available. The first practical steps in this direction have already been taken. In autumn 2012, the Finnish Road Administration announced a public competitive bid for producing travel time information based on cellular FCD on Finnish road networks as part of the DigiTraffic service.

In the Finnish Lihark architecture, the weather subsystem has the same layers (probing and data assembling) as the traffic flow monitoring subsystem architecture. Fixed weather stations use XML-based formats over various transfer protocols (e.g. HTTP, FTP and SMTP) to convey measurements between probing and data assembling layers. When purchasing weather data from various sources, this simple XML-based sensor data format is required to be used, and applies also to extended FCD when used to pass on weather information.

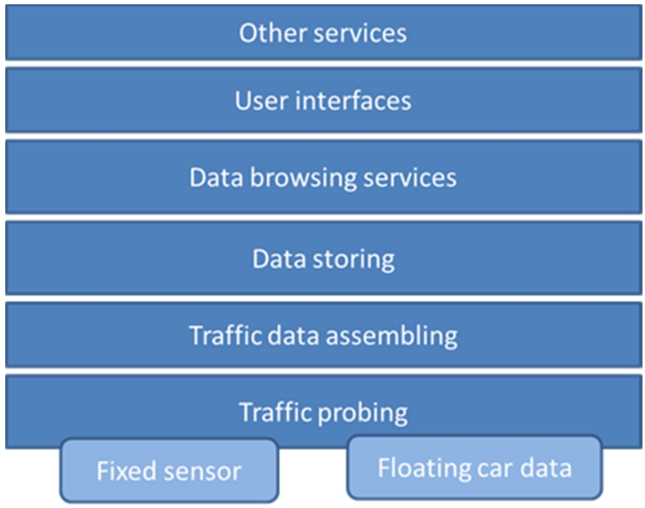

Figure 9. Traffic management subsystem for traffic flow and weather data monitoring.

The TosiArkki project (Siponen et al. 2005) aimed at clarifying roles and responsibilities in real-time transport information by describing possible value chains for real-time traffic information services, and to present the effects of realizing those services. The architecture describes the service transport mode-independent processes for transport and public transport networks. One of the future development ideas that emerged in the project was “distributed traffic information collection” using FCD in such a way that multiple parties would produce a traffic information system for their own repositories, where data could be delivered to a potential user via a standardized interface.

4.3.3 PASTORI and SUNTIO

The PASTORI and SUNTIO projects (Vilkman et al. 2010, Vilkman et al. 2011, Pilli-Sihvola et al. 2011a, Pilli-Sihvola et al. 2011b) funded by Tekes in 2009–2010, studied services and business models that exploit advanced location-based technologies. The motivation for these projects was that the greatest challenges of ITS service development are the lack of working business models, the ambiguity and complexity of value networks, fixed standard practices through regulation or long history, and a lack of risk-allowing approaches that should be facilitated. The aim of these projects was to take the first step towards meeting these challenges.

As part of the work, PASTORI envisioned a model of a multi-service platform as depicted in Figure 10. The key idea of this concept is to acquire location information as to the whereabouts of end users and collect this information into a location database via Internet. This information is then available to service providers through contracts that they make with the end users who subscribe to offered services. The service providers can utilize the information and create useful and valuable content and deliver it to the users' terminal of choice (e.g. a vehicle's information system, a personal navigation device or a smartphone) (Vilkman et al. 2010).

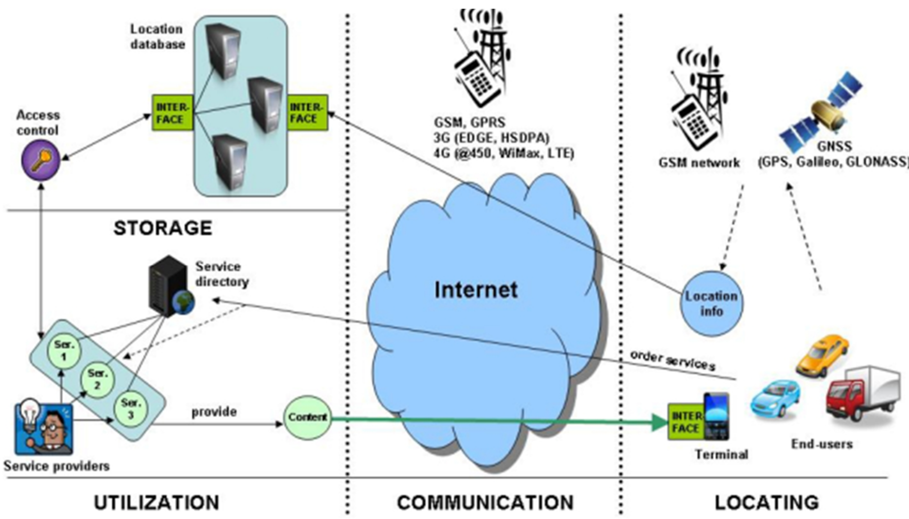

Figure 10. Envisioned PASTORI multiservice platform (Vilkman et al. 2011).

The key issues in this architecture are the interfaces for delivering location information to the location database and accessing the database. The interface for delivering content on users' terminals should use open and standardised ways of representing information in order to create a level playing field for different application and service providers. Access control to location information must be implemented in order to address privacy issues. Standardisation of data formats and interfaces is essential for the creation of such a platform.

According to Vilkman et al. (2010), this Multi-Service Model developed by the Pastori and Suntio projects is based on a mature value network and a common business model having specified roles for different stakeholders (large businesses, SMEs, authorities). The Multi-Service Model aims at more effective service production and a radical change in the service market through large-volume consumer services and well-refined support services for transport and travelling.

4.3.4 MonArkki

The Ministry of Transport, Traffic Agency and Traffic Safety Agency funded MonArkki project (Salovaara et al. 2012). It was supposed to produce a functional architecture for multi-service but the budget and timeline for such an effort was too modest. A good result from the MonArkki work was that it clarified the demands and needs of authorities, and created a framework of what the multi-service could be for authorities. The MonArkki report (Salovaara et al. 2012) contains the state-of-the-art, premises, objectives and implementation for multi-service architecture, architecture requirements and principles, authority requirements, multi-service players and their role description (see Figure 11), list of the key interfaces to be harmonised, lists of possible contents, list of key concepts with description of the multi-service model, and a development plan for the future.

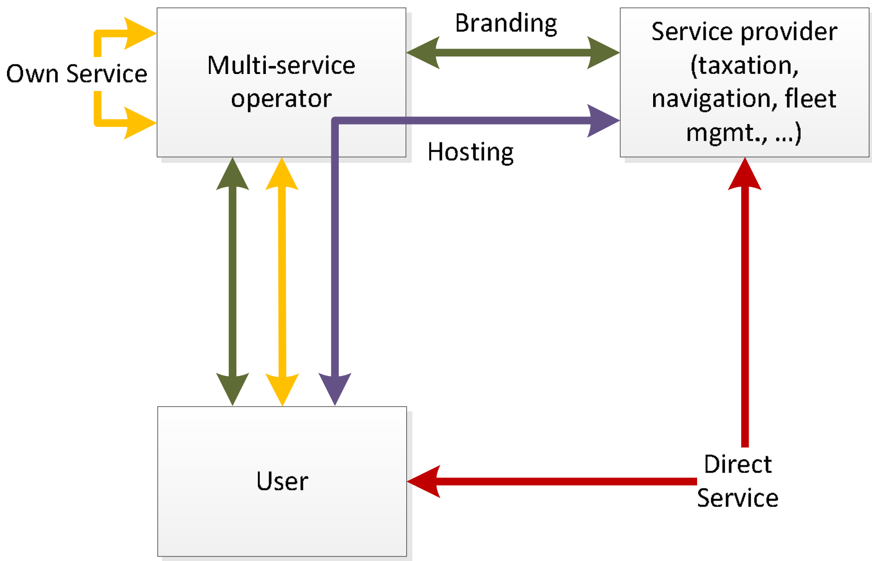

Figure 11. Role of the multi-service operator (Salovaara et al. 2012).

5. Requirements for the architecture supporting Multi-Service Model

5.1 Collection of requirements

MSF aims to support the core value chain described in Section 3.2 in its all parts. This is reflected in a large variety of stakeholders and potential cases as sources of the requirements for MSF. The purpose of the requirements collection was to determine the main functionality and other central features needed to build and provide services in a way that it fulfils the objectives and delivers the expected benefits of the Multi-Service Model (see Section 3.3).

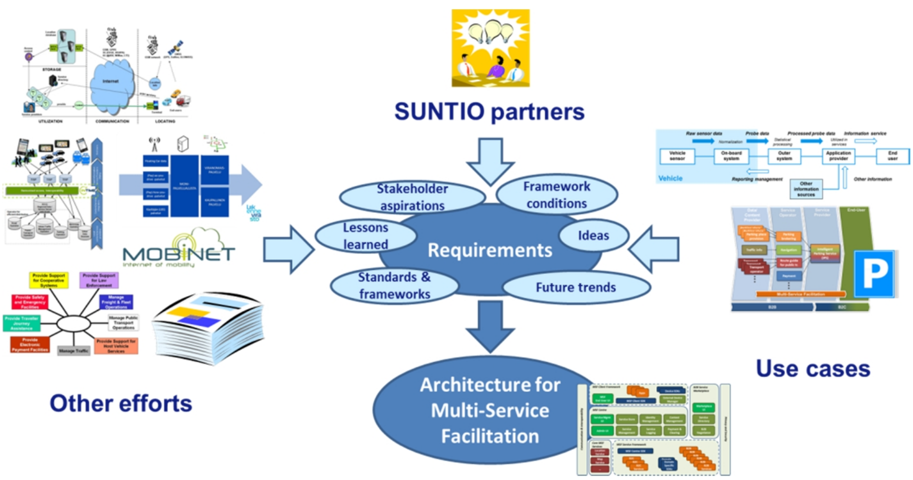

Figure 12. Requirements collection for the functional architecture of the MSF system.

Information for the requirements analysis was collected from several sources and in several forms (see Figure 12). First of all, the objectives and the expected benefits of the Multi-Service Model provided the initial set of high-level aspirations that were already produced in the earlier projects like PASTORI and SUNTIO (see Section 4.3.3). The SUNTIO2 project, in the context of which this report is produced, involved a tight group of commercial stakeholders already familiar from these earlier efforts, which enabled sharpening of these results to a more concrete level. SUNTIO2 partners formed a group that is able to populate roles in all service-provider roles in the core value chain (see Section 3.2) as well as many of the other roles enabling the core value chain (see Section Core value chain). Workshops, topic interviews and informal discussions with SUNTIO2 partners allowed us to form a solid view of stakeholder aspirations with respect to MSF.

The stakeholder aspirations collected from the SUNTIO2 partners were complemented with information coming from other related parallel efforts, in which SUNTIO2 team were participating. The MonArkki project (see Section 4.3.4) studying the applicability of the Multi-Service Model to the objectives of the Finnish Transport Agency (Salovaara et al. 2012) gave an insight into the authority perspective and public-private partnership. The work towards the development plan for ITS in the City of Helsinki (Helsingin kaupunki 2013) in turn concretized the needs for intelligent transportation-related services in cities in the near future. In addition, the SUNTIO2 work benefited from the early work of the MOBiNET project, which started in early 2013 (see Section 4.2.5). MOBiNET actually provides an important opportunity to continue and concretize the work with the Multi-Service Model in a European-wide setting, with emphasis on mobility service realization in the B2B ecosystem.

In order to gain background knowledge in the requirement analysis we also studied several other European ITS projects (e.g. Instant Mobility, In-Time), European ITS Framework Architecture (FRAME), existing ITS systems from Japan and the USA, and existing commercial efforts. See Chapter 3 and Appendix D for further information.

Finally we conducted an analysis of two use cases in order to help concretize the requirements for functional elements required in the architecture. The use cases were selected to represent both ends of the core value chain (see Section 3.2). The FCD collection case (summarized in Section 5.3.1 and thoroughly discussed in Appendix A) aims at finding requirements for MSF from the data and content production and service aggregation (i.e. from the start of the core value chain). The Intelligent Parking Service, in turn, starts from the end-user experience of the anticipated future service and analyses the case further for impacts on service supply and aggregation. Figure 13 illustrates the emphasis of the selected use cases with respect to the core value chain.

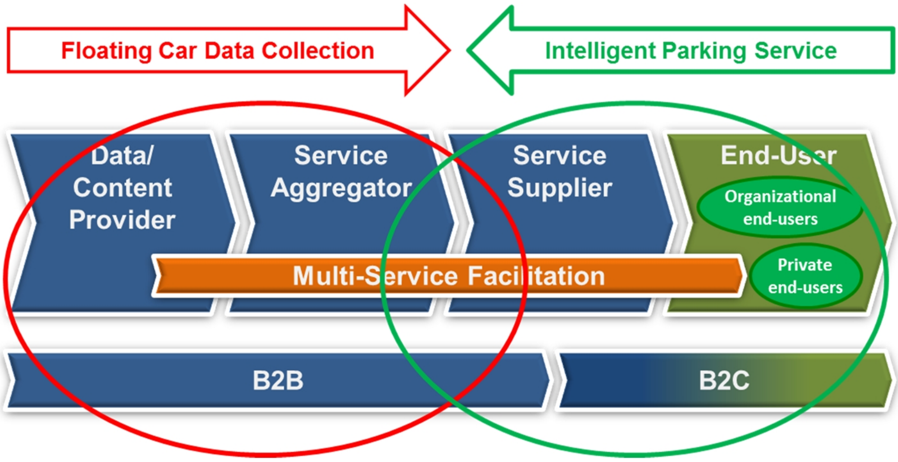

Figure 13. Emphasis of the selected SUNTIO2 use cases in the core value chain.

From the collected information summarized above, we have synthesized high-level functional and non-functional requirements for the architecture (see Section 5.4). Chapter 5 continues from this and describes a functional architecture candidate for MSF.

5.2 Stakeholder aspirations

In the following we summarize some of the central stakeholder aspirations that have come to light during SUNTIO2 discussions and from other sources where the Multi-Service Model has been discussed (e.g. the MonArkki and MOBiNET projects).

- **Standardized data.** Fleet owners and fleet management service providers are willing to sell floating car/cellular data (FCD), but to do so they need a standardized dataset definition for easy aggregation of traffic data from several sources. For their part, authorities and cities have stressed their need for data that can be used to form real-time views of traffic status. FCD is considered the most promising candidate for this, as a fixed roadside infrastructure providing a sufficient amount of traffic data would be too expensive to realize. It is easy to see that this need relates not only to FCD but also to many other traffic and mobility related data and content. DATEX II may provide a solution in some cases but in other cases may prove too complex.
- **B2B service market place.** Content and service providers would like to see MSF provide a common marketplace where they can offer their data and services to other service providers. On the other hand, service aggregators need the marketplace to easily search for suitable services on which they can build their own aggregated services.

- **Public-private partnership for public services.** Authorities have expressed their wish to have at least part of their services hosted or operated by commercial stakeholders. However, such an arrangement may require that systems that are vital to the authorities can reside securely outside the system providing or hosting the public service. This would include e.g. certain registers or sealed and certified devices (e.g. for road tolling).
- **Public services as a driver.** Commercial stakeholders that are potential operators of MSF (e.g. a platform) would like to see public services strongly drive the development of multi-service infrastructure. Examples of such drivers are eCall and road tolling. This desire is shared, e.g. by device manufacturers.
- **Open data exploitation.** The European Commission, along with national initiatives, strongly promotes the opening up of public data for eService creation. This desire is also shared by service developers and the public. MSF should, therefore, support the effective utilization of open data in the realization of services.
- **Multi-service platform scaling and interoperability**. Many stakeholders are already developing or planning to develop their own platforms that could serve in MSF. As no one expects their platform be the only choice in the future, they aspire to have a common general architecture, standardized APIs and protocols for interoperability between platforms. The need for interoperability between platforms has also come up in several other contexts, e.g. for scaling the multi-service platform geographically and for different stakeholders to be able to focus on different services and target groups.
- **Integration of existing (legacy) systems.** Service providers and many other stakeholders (e.g. authorities) have emphasized the importance of architecture that enables easy and secure integration of existing systems (legacy systems).
- **Varying roles and tasks in MSF.** Stakeholders that have expressed their wish to operate MSF have different target groups to whom they want to direct their services (e.g. B2B vs. B2C) and different sets of activities they are ready support (e.g. separation of technical platform operation vs. operation of multi-service provision).
- **Payment and clearing as a revenue source.** Some stakeholders already have payment and clearing functions in their operation and systems. They understand that using this existing functionality and integrating it into MSF (e.g. platform) is cost effective and would provide one needed source of revenue in multi-service operation (e.g. in the form of provisions).
- **Multi-service environment as an open innovation ecosystem.** The Multi-Service Model is seen as a promising candidate for an open innovation

ecosystem in which new and experimental services could be developed by a wide range of stakeholders. The system supporting this kind of environment should provide tools for easy development of such services.

- **Multi-service environment for secure and certified service provision**. Companies willing to use services conveyed by MSF wish to see that they do not pose any kind of threat to their mission-critical systems (to which they may connect the services). Thus the Multi-Service Model should, in accordance with their wishes, be realized in a way as to demonstrate the quality and security they require.

5.3 Use case studies

The starting point for the selection of these use cases is the simplified value chain depicted in Figure 1, which shows the elaboration of traffic-related content to services consumed by end users. The first case, dealing with FCD, concentrates on the start of the value chain and pinpoints the needs arising from the realization of traffic-related services. The second case, dealing with Intelligent Parking Service, gives an example of end use of the services. In this way we aim at pinpointing the central requirements to fulfil when developing architecture candidates for the system enabling MSF in the core traffic service value chain. In the following we present short summaries of the cases. The complete case descriptions are given in Appendix A and Appendix B.

5.3.1 Floating Car Data Collection

Traditionally, the collection of traffic information data is based on stationary sensors that are either embedded into roads or set up along roadsides. The main problems of a fixed sensor infrastructure are limited coverage and high establishment and maintenance costs. Other minor drawbacks are impaired sensing abilities in certain weather conditions. Advances in traffic sensing technologies have enabled new methods that are commonly referred to as floating car data (FCD) or floating vehicle data (FVD). FCD refers to technology that collects traffic status information from a set of individually recognizable vehicles that “float in traffic”. At least three different sources for FCD data probing can be identified: the global navigation satellite system (GNSS), short-range radio communication, and the cellular network system.

In the FCD-based approach a collection of uniquely identified location samples are collected and map matched to identify the correct road segment on which the vehicle is travelling and to determine the vehicle location on that segment. The data collection and refinement process is shown in Figure 14.

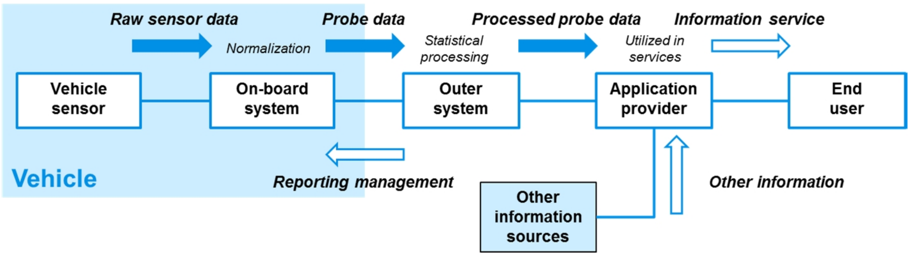

Figure 14. Floating car data collection process (Burkert et al. 2010).

When enough map-matched FCD data has been collected, the information can be used or processing multiple traffic services including:

- Road network performance analysis
- Monitoring of the complete network
- Before/after analysis, bottleneck analysis
- Route choice in case of events/detours
- Forecasting/estimation of traffic growth
- Origin-destination (OD) relations & matrices
- Route choice analysis for traffic simulation
- Emission modelling (air & noise pollution)
- Road maintenance and safety analysis
- Fixed sensors maintenance analysis.

Extended Floating car data (xFCD) uses new specific devices and sensors endowed in modern vehicles by using the vehicle's on-board diagnostic (OBD) systems. xFCD provides even more alternatives for traffic-related services including:

- Weather information
- Traffic incident data (collisions, accidents etc.)
- Road condition information for maintenance
- Vehicle-related applications (distance related insurance & leasing, car finder & stolen vehicle etc.).

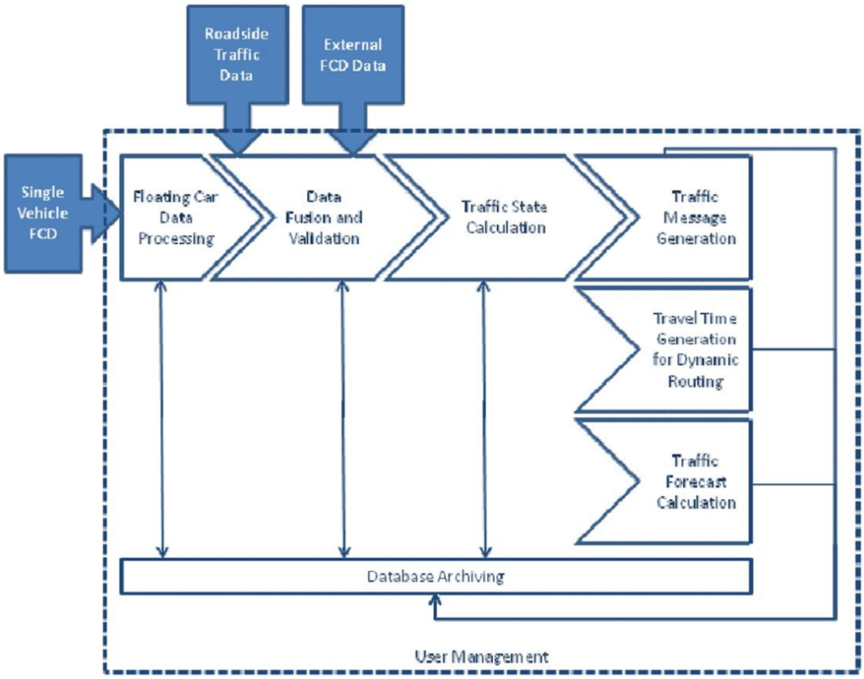

Figure 15. Multiple data source and data fusion used in traffic information provisioning (courtesy of Viaejo project).

One of the main aims of the FCD use case was to study the possibility of forming a real-time traffic snapshot based on xFCD. Studies have indicated that producing reliable travel time information requires a floating car penetration of around 2–5% of the total vehicle count. For detecting disturbances within minutes a far greater penetration, probably in the range of 5–10%, would be needed.

To achieve the required reliability and coverage for traffic applications, a combination of fixed and multiple FCD data sources need to be applied. This requires multi-sensor data fusion, which can be defined as “the theory, techniques and tools which are used for combining sensor data, or data derived from sensory data, into a common representational format. In performing sensor fusion our aim is to improve the quality of the information so that it is, in some sense, better than would be possible if the data sources were used individually.” The need for a common representational format led to further study of “a minimum dataset” as presented in Appendix C.

Current location systems can track users automatically and can generate an enormous amount of potentially sensitive information. Sensor information provided by a floating car could be used to generate vehicle profiles or even driver specific profiles unless the data is protected by an intelligent security and privacy system. The xFCD platform (or more generally ITS platform) needs user identification, authentication, and authorization as well as communication security. If very rigid security and privacy features are required, this affects the system architecture because the data collection and storage elements should be separate entities that use strong public key cryptography methods and infrastructure (PKI) for user authentication and privacy.

5.3.2 Intelligent parking service

The case study on Intelligent Parking Service (IPS) aims at providing an end-user perspective to the Multi-Service Model by describing a future parking service that combines several parking-related services into one aggregated service that supports the whole parking process. It shows how parking involves several related activities by the traveller in different roles: finding a parking space, navigating to it, parking and detection of parking, further navigation to the final destination from the parking space and back (possibly using public transportation e.g. park & ride), payment for parking, end of parking. The whole process involves several stakeholders, including services dedicated to supporting a certain part of the parking process (e.g. navigation or payment) or external systems (e.g. access control infrastructure or on-board units).

The IPS case clearly shows the validity of the multi-service approach and clarifies the requirements for the environment needed to support provision of such a service. The end-user perspective in the case study naturally produces requirements relating to the service provision in the first place. However, these requirements induce further requirements down the value chain, where the service is realized by aggregating it from several constituent services and systems.

Bundling several services into one aggregated service brings benefits to the user, as he does not need to discover and start a service for each parking-related task separately. Integrated payment, fluid transitions from service to service depending on the need, context-sensitivity, and proactivity with alerts and notifications are all features that make the service appealing and useful to the user. In addition, to further ease the parking process, the traveller’s mobile terminal or on-board unit should ideally speak with the infrastructure (in the IPS case with the parking infrastructure or with the OBD II unit in the car). All these features can be supported either by the service itself or they can be provided by the environment (e.g. service platform) used to realize the service. The main difference is, though, that in the latter case, building new services with these central mobility features would be much easier and cost-effective.

Bundling several services together by a service aggregator, in turn, requires means to exploit several services as building blocks. This requires a service marketplace for B2B service discovery, service level agreements and service descriptions. In the realization of aggregated services, common data models and standardized APIs and protocols should be provided when possible. In many cases there is also a need to dynamically change the B2B services exploited in the service aggregation (e.g. different parking place providers in different regions and at different times). The single sign-on and payment features propagate requirements for MSF such as user management, payment and clearing functionality.

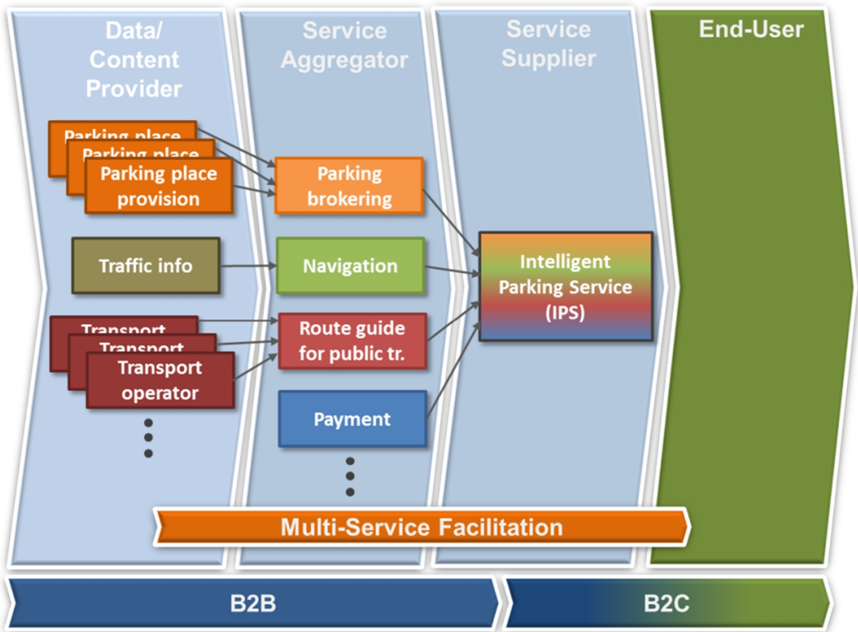

Figure 16. Example of service aggregation in the Intelligent Parking service case.

5.4 Summary of requirements

The analysis of stakeholder aspirations and the case studies above place a number of requirements at the core of the Multi-Service Model. The following summary aims not to exhaustively list these requirements, but rather to pick up the most relevant ones for the general architecture design.

5.4.1 Requirements for the service provision to end users

One-stop shop for end users. One of the key ideas of the Multi-Service Model is to collect traffic-related services in the same place in order to ease service discovery and use. For this purpose, MSF should provide means to arrange the service offering in a way that allows service users to easily find, compare and deploy the services. Furthermore, the service discovery should be (at least optionally) contextual so that services offered to the end users are relevant to the situation they are in (e.g. the service is available in a given geographical position).

<u>Requirement:</u> *To ease service discovery and use, MSF should provide means to arrange the service offering in such a way that service users can easily find, compare and deploy the services that are relevant to their context.*

Sources: Stakeholder aspirations, IPS use case

Single sign-on. Many mobility services (like the Intelligent Parking Service use-case shows) involve the use of several services from different service suppliers. Different services may require setting up user accounts defining user identities in order to use them – at least if advanced features comes into play (e.g. personalisation). Setting up and maintaining multiple service accounts, however, requires a lot of work from the end user. Furthermore, signing in separately to each service when using them worsens the user experience. Single sign-on, in turn, necessitates centralized user management by the system supporting Multi-Service Facilitation. It should be noted that separate traffic services may still maintain their own user management in parallel and therefore linking of the existing identities in the traffic services and the identities created in the Multi-Service context should be possible.

<u>Requirement:</u> *In the multi-service context, creation of user identities for each service should be replaced with a single user identity to allow single sign on to all traffic related services.*

Sources of the requirement: Stakeholder aspirations, IPS use case

Single payment. When multiple services are integrated to produce an aggregated service like in the IPS case, some of those services may impose charges, like the parking itself. Even though other payments than parking are not apparent in the IPS use case it is easy to imagine that the same case could be expanded to include e.g. road tolling when driving to a city centre. Navigation may also have an extra service that finds the best route to the parking place at a particular time and gives traffic information by exploiting real-time traffic data, which in turn might add some extra costs to the free basic navigation. Separate payment contracts, possibly varying payment methods and excessive interaction caused by the payments for each service separately causes inconvenience for the end user. Collecting all the payment transactions under the one common payment scheme allows for easy monitoring of the costs, easy payment with a selected single payment method for all services, and less user interaction with the payment transaction. Single payment implicates the inclusion of payment functionality either within the MSF or close cooperation with a 3rd party payment operator.

<u>Requirement:</u> *To avoid multiple payment contracts and/or payment transactions to several different service suppliers, the system providing IPS should allow all payments as a single transaction for the end user.*

Sources: Stakeholder aspirations, IPS use case

Contextualization. Traffic-related services like IPS are strongly bound to the geospatial context, which can be static (or slowly changing, e.g. geography, traffic network) or dynamic (e.g. traffic situation). The services, service suppliers and underlying service providers are selected based on this context and on the user's varying needs and preferences. For instance, in the case of IPS, parking areas of interest lie near the driver's destination (or along the public transportation route leading to the destination) and nowhere else. Therefore, traffic related services like IPS should be adapted to the context. This, in turn, induces requirements for MSF. It should provide means to acquire context information (location being the most important context parameter) from the user (and his/her devices and vehicles).

Requirement: To provide the right service in the right situation, the system should provide support for contextualizing services.

Sources: Stakeholder aspirations, IPS use case

Personalization. Users consuming services in a multi-service environment wish to customize their use experience to various degrees. The need for personalization can be directed either to the multi-service environment, the services, or both. Personalization may be based on profiles and settings explicitly provided by the user or learnt automatically from user behaviour.

Requirement: To provide a personalized service experience, the system should offer personalization functionality for the multi-service user experience and for end user services.

Source: IPS use case

Proactivity. Many of the services can be based on the request/response pattern. However, in traffic it is useful to have also a proactive mode for some services, like those that alert the driver to traffic incidents requiring immediate attention. Location-based advertising is another service that falls well into this category. Proactive features are also invaluable assets when designing safe and effective mobile or in-vehicle user interaction, as they reduce the need for user interaction (see requirement for *safe and effective user interfaces for mobile users* above).

Requirement: To initiate an alert or notification to service users, the system should allow proactive service functionality.

Sources: Stakeholder aspirations, IPS use case

Safe and effective user interfaces for mobile users. For example, In the IPS case the traveller uses the service mostly while on the move (by car, as a pedestrian or on public transportation). This requires a special design for user interaction, as using the service should not lower safety in traffic. In addition, mobile or in-car terminals with restricted UI related resources (compared to e.g. a desktop computer) may require exploitation of new innovative UI paradigms in service interaction to benefit the end user most effectively. Natural user interfaces (speech, gestures, etc.) may be needed to complement more traditional user interaction. In addition, automatic detection of the situation of use (including context) and adaptation to it, as well as proactive functionality, can be used to reduce the need for user interaction.

Requirement: *To provide safe and effective use of services while mobile, the system should allow construction of innovative user interfaces that do not distract the user's attention from traffic and provides effective use with mobile or on-board terminals.*

Sources: Stakeholder aspirations, IPS use case

Support for heterogeneous end-user terminals. Users of traffic services come with a variety of heterogeneous mobile and in-car terminals. In addition, part of the use takes place on desktop computers with superior user interaction capabilities compared to mobile terminals. For service providers, suppliers and builders this heterogeneity is a big source of costs or just restricting their offering to certain terminal platforms. Here, MSF should provide a common service and application development environment in order to realize services for everyone independent of the user terminal.

It should be noted that heterogeneity is not related only to user terminals but covers whole ecosystems (compare e.g. Apple or Google ecosystems) that may restrict exploitation of certain type of solutions.

Requirement: *To significantly reduce development costs of the service provider, suppliers and builders, the system should provide platform-independent development kits, libraries and APIs for building and providing services and applications for heterogeneous terminal environments.*

Sources: Stakeholder aspirations, IPS use case, FCD

5.4.2 Requirements for B2B service provision and service aggregation

B2B service marketplace. Many services exploit other services as constituent building blocks. For instance, FCD providers alone often cannot provide sufficient coverage of traffic data, but together with other providers and an FCD data aggregator this is possible. For that reason, MSF should not only support service provision to end users but also B2B type service provision for service builders. The key feature needed for this is a service directory that allows service providers to add their service descriptions for others to search and inspect. Ideally this service directory would have support for negotiation of terms of B2B service use, clearing, service level agreements, etc. Thus MSF could be used as a tool to build a B2B service ecosystem and help all players bring their offering to the marketplace.

Requirement: *To support the B2B service ecosystem and service aggregation, MSF should provide a B2B service marketplace with a service directory for service providers.*

Sources: Stakeholder aspirations, IPS use case, FCD

Support for service aggregation. Aggregated services use other B2B services as their building blocks. In addition to the B2B service marketplace, service builders need support for service aggregation to ease their work. MSF should provide service development kits, common APIs, and supporting functionality to aggregate and integrate services. Here it should be noted that service integration and aggregation may take place both in backend (back office integration) or in frontend (e.g. mobile terminal environment) depending on the architecture or type of service.

Requirement: *To easily aggregate functionality from several different content and service providers, the system should provide easy yet powerful aggregation mechanisms for service builders and service providers to realize new services that exploit other B2B services.*

Sources: IPS use case, FCD, stakeholder aspirations

Dynamic configuration of aggregated services. In the case of aggregated services the constituent underlying B2B services (from which the service is built) may be different depending e.g. on the location or time. In the IPS case, for instance, the parking place providers or even parking brokers may vary depending on the city. Yet for end users the same application should work wherever they wish to use the service. For that, a dynamic rule-based configuration of B2B services as the basis of aggregated services is needed. This, in turn, would require a well-thought-out set of metadata to define each service (e.g. the area where the service is relevant).

Requirement: *To build an aggregated service that functions similarly independent of context, MSF should allow dynamic or automatic configuration of the underlying B2B service blocks depending on the context.*

Sources: IPS use case, FCD

Clearing between different service suppliers and service providers. In the case of single payment for end users, financial flows from end-user transactions should be possible to clear to different service suppliers based on the use of their services. In addition, the system should support B2B service provision for service aggregation and possibly manage the clearing based on B2B service use by aggregated services (e.g. use of traffic data from different data providers, such as FCD providers, to provide real-time traffic situation as a service).

Requirement: *To simplify the complexity of financial contracts and flows between multiple actors in service provision, the system should provide effective clearing mechanisms.*

Sources: Stakeholder aspirations, IPS use case, FCD

Common data formats. Traffic-related information comes from numerous sources and no source alone is sufficient. For example, in the IPS case parking place information (both static and dynamic) is collected from several stakeholders. This might take place by parking brokers that aggregate the offering of single parking place providers to cover, e.g., some certain local area. In the IPS case there can be several parking brokers involved when querying available parking space. Collecting, storing and querying this information would not be possible if there were no common data formats as a basis of the service. Similarly, collecting

FCD data from several FCD providers should take place using one common data model and format in order to easily aggregate small flows of FCD to form a holistic view of the real-time traffic situation.

<u>Requirement:</u> *To enable effective aggregation of services by service providers and application builders, the system should provide support for harmonizing central traffic related data with the help of common data formats, APIs and protocols.*

Sources: Stakeholder aspirations, IPS use case, FCD

5.4.3 Requirements for supporting multi-service facilitation functionality

User management. Single sign-on and single payment, among other things like personalization, necessitates user management by the system supporting MSF. Here it is important to understand that users come in different categories, e.g. private end user customers, company employees (with company contract connected to the identity), organizational roles (where the identity is not bound to a single user) etc. Furthermore, it should be noted that separate traffic services provided using the Multi-Service Model may still maintain their own user management in parallel. Thus linking existing identities in traffic services with identities created in the multi-service context should be possible.

<u>Requirement:</u> *To enable single sign-on, single payment, personalization and other functionality and features related to user identities, MSF should include functionality and data models for user management.*

Source: IPS use case

Service event logging. Logging of service use to some degree is necessary for many reasons. First of all, payments should be based on verifiable service use, and proving the grounds of payment must be provided whenever requested. Second, the service providers want to know about their service use (how much, when, by whom etc.) in order to further develop their services. Third, the clearing between different service providers should be based on verifiable facts that can be collected from the logs. Fourth, end users may want to follow their service use for different purposes (depending on the logging contents). In this case, it should be noted that users may be company employees and it is the company accounting department that wishes to supervise some of the service use. Fifth, if done while respecting users' privacy, logs can be used to provide invaluable information for service planning, advertisement etc.

<u>Requirement:</u> *To make events related to service use verifiable and useful for other purposes, MSF should include persistent logging of service-related events.*

Sources: Stakeholder aspirations, IPS use case

Location-based data collection. Collecting location data and tracking information from the users as a side product of location-based service use produces valuable information that can be used to enhance the service itself (e.g. learning service) or as raw material for other services (e.g. traffic data). Tracking itself may be a service. No doubt, each service can separately collect this information (with permission from the user) but all involved parties would get better data if all of the data were collected into the same place by the operator of the system realizing the Multi-Service Model.

Location and tracking data, however, are sensitive with respect to privacy and therefore this kind of data collection has special considerations including anonymization, user control of the data, privacy laws, data security etc. For that reason, collecting location information may not seem appealing to some stakeholders. On the other hand, suitable incentives may encourage users to share their location data if they trust the system.

Requirement: *To enhance services and gather traffic data for the construction of other services, the system should allow collection of location and tracking data in a way that fulfils all the privacy and security requirements.*

Source: Stakeholder aspirations

5.4.4 Non-functional requirements

Security and privacy. The use of traffic-related services reveals a lot of information about the behaviour patterns and whereabouts of service users. If this data is collected for further use, the risk of misuse or exposure of sensitive data grows even more. Therefore privacy and security issues are of paramount importance throughout the value chain, architecture and supporting components. Any flaws in this respect can easily bring down the whole multi-service operation.

The minimum requirement is that privacy laws (which differ from country to country) must be taken into account. Technical and other proven solutions must be used to secure sensitive information. Anonymization and other solutions to dissipate users' identity from the collected data must be used. However, no solution is good enough if the users do not trust the service or the system. Therefore, special attention must be paid to conveying a message of well-managed privacy and security to the users.

Requirement: *To comply with privacy laws, to secure the privacy of system users and to convey trust, the system should offer solutions to all privacy and security concerns.*

Sources: Stakeholder aspirations, IPS use case, FCD

Scalability. The number of users and use of services in the multi-service environment are likely to grow from initial early adopters to a nationwide or even international scale. Especially in the case of mandatory public services (e.g. planned road tolling), the use of services comprises service transactions from millions of users. Thus the architecture and solutions used should be scalable in several dimensions, such as number of users (administrative scalability), workload (load scalability), geographical distribution (geographic scalability), and new functionality (functional scalability).

Requirement: To ensure sustainability of the developed multi-service environment, the supporting system should be scalable.

Sources: Stakeholder aspirations, IPS use case

Real-time data support. Some central traffic-related services are based on real-time data. The data may be massive in the B2B service aggregation part of the core value chain, and latency requirements for this data may be strict in the service provision to end users. This raises the need to support MSF in a way that enhances real-time capabilities of the core value chain or at least does not significantly add load in that respect.

Requirement: To enable services requiring real-time data, MSF should be based on solutions that maximize throughput of real-time data from data source to service provision in the core value chain.

Sources: Stakeholder aspirations, IPS use case, FCD

Reliability and high availability. Some traffic-related applications are very sensitive to reliability and constant availability. Effective traffic management and control require reliable traffic detectors and communications. Co-operative ITS aiming to increase traffic safety needs low-latency and reliable real-time communications, dynamic traffic control, and air and rail traffic that place very high demands on the reliability and availability (always working) of their systems. Customers who pay for services as end-users or service aggregators have to be ensured that they will get the service in a way service-level agreements (SLAs) are fully complied with.

Requirement: To provide reliable and continuous service offerings, MSF should be based on solutions that ensure maximum reliability and availability to fulfil the various needs of different SLAs.

Sources: Stakeholder aspirations, IPS use case, FCD

6. Multi-Service Architecture

6.1 Architectural context

MSF can be seen as a collection of processes and data flows that realize the Multi-Service Model in practice. In the context diagram depicted in Figure 17 this collection is presented as a single system (or process) that interfaces directly with external stakeholders (human actors and organizations as well as external systems).

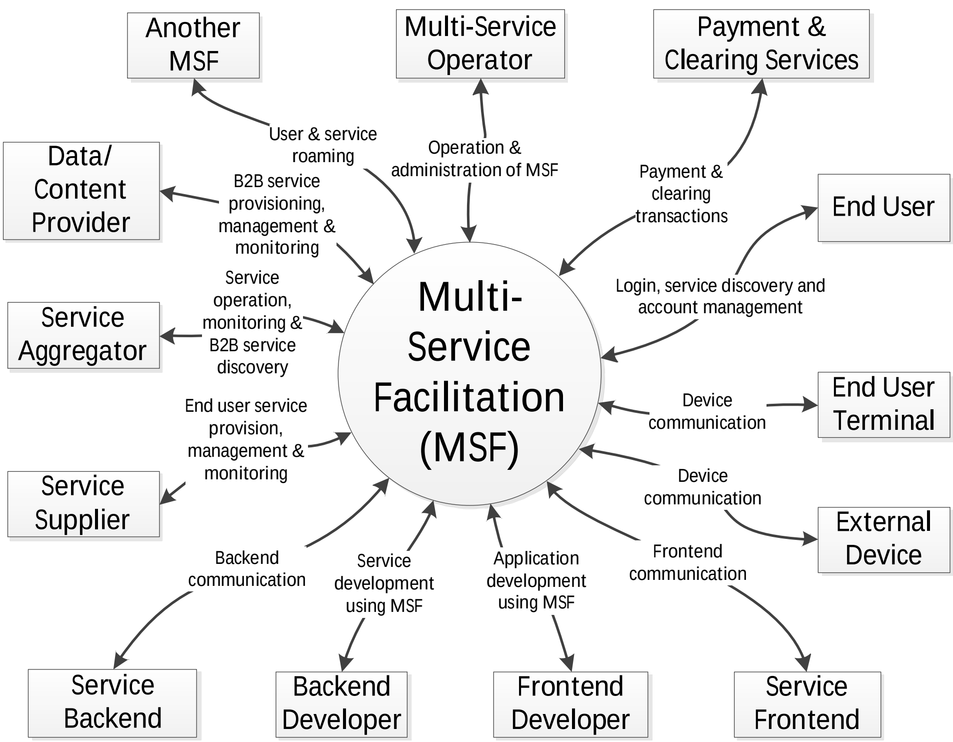

Figure 17. Context diagram for Multi-Service Facilitation.

Data/content Provider refers to stakeholders at the start of the core value chain that provide B2B services for service providers (integrators, aggregators, data processors, etc.) to build on. They are often content/data providers producing raw content for further elaboration.

Data/Content Provider use MSF e.g.

- to market their services for service aggregators,
- when providing their services in the MSF context by building or adapting their services to comply with the MSF framework and exploiting the MSF system with the help of service developers,
- to manage their service offering (e.g. updating their service description and API definitions),
- to monitor the use of their services with the help of MSF logging and monitoring, and
- to inspect clearing and revenues with respect to their services.

The **Service Aggregator** integrates other services with its own service offering, and aggregates and processes content received from data/content providers (possibly combining it with its own content), in order to provide services that service providers can use as a basis for their end user services.

Service Aggregators use MSF for same type of tasks as the data/content provider but from a different perspective. For example, in monitoring they are interested not only in the use of the services they produce, but also in the B2B services they build upon. In addition, they use MSF in their own service development to search for suitable building blocks from the B2B service offering.

The **Service Supplier** packages and offers the services produced by the service aggregator to the targeted end-users. In many cases, the service supplier may also act as a service aggregator (by creating further added value to the service).

Service suppliers use MSF for similar tasks to those of data/content providers and service providers, but added to these is the perspective of end-user service distribution, e.g.

- marketing and visibility to end users,
- management of end-user applications and their versions (if separate) and service-user interfaces (when, e.g. web-based services),
- management of the service provider's own end-user customers,
- customer service and other end-user oriented communication, etc.

For all types of service providers, the above tasks requiring interaction with MSF implies that there should be some kind of user interface that they can use for these tasks.

Service provided in the multi-service environment is either **service backend** or **service frontend**. In this simplified model, the service backend processes the service requests and manages the related data, while the service frontend provides the service interface to the end user. This division corresponds to the traditional client-server architecture. Mobile apps or web interfaces are good examples of frontends that communicate service requests over the network to the service provider's backend systems. It should be noted, however, that intelligence, data management, and service request processing can be balanced between backend and frontend in several ways, or the backend may in some cases lack altogether (e.g. true peer-to-peer systems).

In the MSF context, the service (both backend and frontend) may exploit the functionality and features offered by MSF (e.g. standardized APIs, user management, payment, starting of other services, logging events etc.). The communication between service components and MSF depends on the richness of the features offered by MSF, architectural choices etc. For example, in an extreme case all the service requests from the frontend may be communicated to the backend via MSF (see Sections 6.2 and 6.4 for further discussion of this topic).

Closely related to service backend and service frontend are the developers, who implement the software performing the services at both ends of the service realization. The **Backend Developer** ensures that the service backend works with, and exploits the services offered by, MSF. He might also integrate service backends together using MSF (e.g. by exploiting some standard data model and API offered by MSF).

The **Frontend Developer**, in turn, designs and implements end-user interfaces and applications that comply with the MSF framework, and exploits the features and functionality offered by MSF. Here MSF may e.g. offer a platform-independent framework for mobile devices, so that the application developed for one mobile phone model or platform also works with other phone models and brands.

External Device refers to any device that is used to provide data to services (e.g. sensors, OBD II), and any device whose functions are controlled by these services (e.g. actuators, access control, traffic signalling). It is not clear what kind of role MSF will play with respect to external devices or whether service providers always use their own systems to manage their devices. As in some realization alternatives MSF could have M2M support, we have explicitly added external device as a possible external actor in the MSF context.

End-User Terminal is a special type of external device among the other external devices that is used to provide interactively the service to the end user by executing the service frontend. End-user terminals have their own operating systems, application development platforms and hardware capabilities that are provided for the applications in various ways. Some of the features, e.g. in mobile terminals are useful also in the MSF context. One of the most important is the capability of a mobile terminal to determine its position. This in turn helps the services, as well as MSF, to contextualize their services. In the near future, mobile payment and mobile wallet might be other central features that become significant in the MSF context.

Standard interfaces between MSF and the user terminal (especially when this is a smartphone or similar) facilitate frontend development for multiple heterogeneous terminal types and standardized use of their capabilities.

The **End User** is a stakeholder with two roles:

1. as a user of MSF with his/her own user interface (e.g. for service discovery), and
2. as a user of services conveyed with the help of MSF.

The explicit appearance of the end user is related to the first role, where he/she directly interacts with the MSF system, for instance to log into the system, manage the user account and discover services. This implies that there should be some user interface with MSF for end users.

In the second role, the end user is present only implicitly as a user of the service frontend, and thus not directly using MSF. However, depending on the way service frontends are constructed in the MSF context, this role division may be blurred. For example, if service frontends are constructed with MSF specific frontend libraries, the use of services may include direct interaction with MSF. (See Section 6.4.3 for the discussion on the alternative frontend approaches.)

Payment and Clearing Services can be integrated in the MSF system or can be 3rd party services used co-operatively by MSF. In the latter case, interfaces for communicating information dealing with transactions liable to charge, payments and clearing should be developed.

Another MSF refers to some other Multi-Service Facilitation system in the distributed environment. Standard protocols and interfaces are needed, e.g. for user roaming of account information and services between the systems operated in co-operation with several MSF operators (e.g. serving in a European-wide environment).

Finally, **Multi-Service Operator** is a stakeholder who operates and administrates the technical system providing MSF. For that, it needs various kinds of interfaces for administration.

6.2 Functional architecture

The functional architecture for MSF (MSF architecture) in Figure 18 reflects the functional requirements presented in Section 5.4. These requirements, in turn, are based on the characteristics of the Multi-Service Model and the typical value chain related to the construction and provision of mobility services (Section 3.2), stakeholder aspirations (Section 5.1) and use case studies (Section 5.3). Furthermore, it takes into account the required interfaces to external stakeholders (human users as well as external systems) interacting with MSF that are described in Section 6.1.

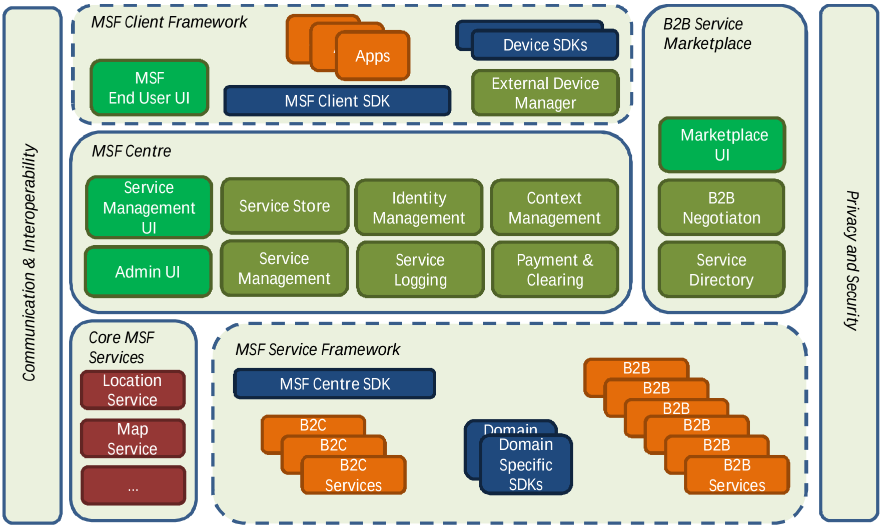

Figure 18. Functional architecture for Multi-Service Facilitation (MSF).

Even though the functional architecture depicted in Figure 18 does not directly represent the topology of the architecture, it clearly seems to indicate a client-server architecture where clients constructed in the MSF client framework communicate with a centralized platform (MSF Centre) and with the service backends using the MSF service network. This kind of interpretation, in fact, has been dominant in discussions with stakeholders involved in the SUNTIO2 project, as well as in ITS related literature and projects. For example, the presented architecture is in line with the generally accepted architectural frameworks used in the development of ITS systems. For example, the European ITS Communication Architecture (Bossom et al. 2010) splits (cooperative) the ITS system into four components: Personal Station, Central Station, Vehicle Station and Roadside Station (see Figure 19). All these components can be included also using MSF:

- The MSF Client Network with its MSF client SDK provides tools for constructing service frontends at the personal station,
- The device SDK and device manager create a way to connect to the vehicle station and roadside stations, and
- The MSF centre and MSF service network support the central station development.

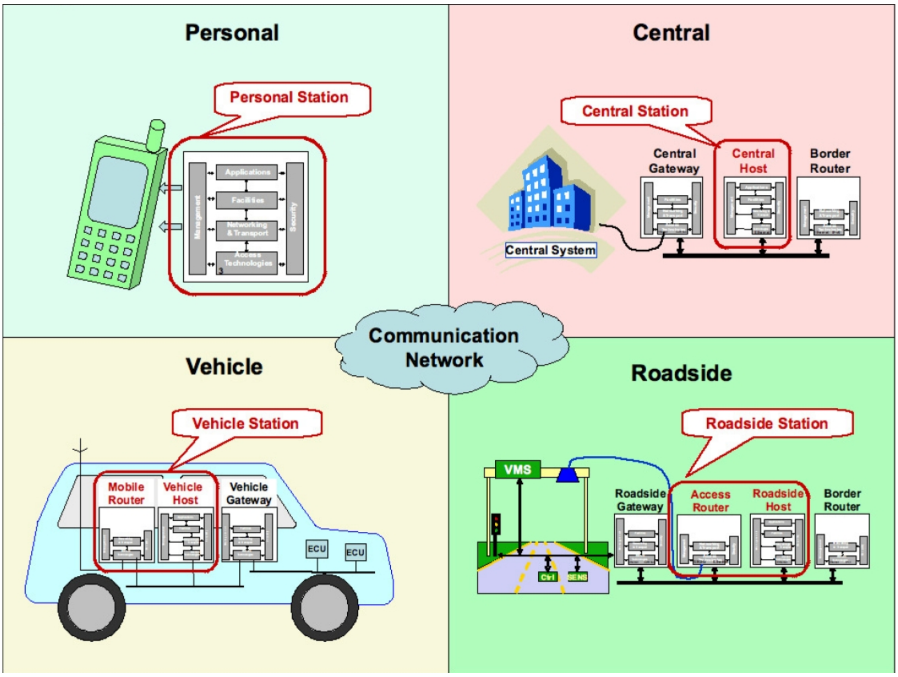

Figure 19. European ITS Communication Architecture (Bossom et al. 2009).

However, the MSF architecture presented here does not exclude other architectural solutions in some instances. E.g. the B2B marketplace in the MSF architecture may be based on a non-centralized peer-to-peer solution as described by Hoyer and Stanoevska-Slabeva (2009).

6.2.1 Components of the architecture

The MSF architecture depicted in Figure 18 is divided into functional blocks of main components. These include:

- **MSF Centre,** which includes the main components for operation of the MSF system. These components are related to service provision, management of users and other entities related to mobility services, management of contexts of the users and other entities, logging of service-related events and, finally, payment & clearing.
- **MSF Client Framework,** which offers the MSF environment for applications at the end-user terminal and from the end-user interface to the MSF centre. In addition, it offers development kits and interfaces for external device interaction.

- **MSF Service Framework**, which offers service development and an aggregation environment for service providers.
- **B2B Service Marketplace**, which offers service discovery and negotiation services for service providers and operators who exploit B2B services and open data in the construction of their own services.
- **Core MSF Services**, containing a set of B2B services that are central building blocks for most of the mobility services and thus an essential part of MSF.
- **Communication & Interoperability**, which refers to all communication management needed between the functional blocks in MSF and between distributed interoperable MSF systems.
- **Privacy and Security**, which refers to the solutions and principles throughout the MSF system that ensure the required privacy and security in its operation.

6.2.1.1 MSF Centre

The MSF centre is the hub of the multiservice operation through which service providers offer their services to their customers. Service providers can rely on the standard MSF functionality in the construction, operation and provision of their services, which lowers the threshold to get their services to the market and reduces the need for their own resources in these tasks. The functional blocks of the MSF centre are as follows.

Service Store manages all the information and required resources that are needed to market and provide services to the service (end) users. It can be compared to the app stores that allow end users to purchase mobile applications in mobile service ecosystems. The difference, however, is that frontends to convey provided services to end users are not necessarily manifested as mobile applications but can be e.g. web frontends. The service store contains e.g.:

- description of the service,
- pricing,
- terms of use,
- metadata for service discovery,
- resources for the service frontend (apps, web code & graphics etc.)
- a user guide and
- social media tools (rating, feedback, community tools etc.).

The functionality of the service store is closely related to user interfaces to the end users (MSF End User UI) in the MSF client network, as well as Service Management UI dedicated to service providers (discussed later).

Identity Management is used to manage all the information related to the entities related to the services. The most important entity is the service user whose user account, personal information, service subscriptions, preferences, service use, transaction history etc. must be managed securely. In addition, other entities like identifiable vehicles and end-user terminals and devices may be managed and related to their owners or other responsible parties by this component.

Identity management is responsible for providing a single sign-on environment where users may log on to any service in MSF context using a single user account. This does not mean that user accounts related to the existing separate services become invalid when the service is migrated to the MSF environment. MSF identity management allows the service user account to co-exist and be used to map the existing service user accounts to the MSF user account.

Context Management is used to store and manage contextual information related to the different entities managed by identity management (end users, terminals and devices, vehicles). The most important context information to manage is location information that can be used to contextualize mobility services or service discovery. Location information can also be used to construct proactive services that are triggered by context (e.g. time and given location) or to dynamically aggregate services to suit the context. Thus, this logical component is closely related to identity management, service store and service management.

Context information is sensitive information that has special requirements for privacy and security as well as the need for sufficient options for transparency and control for end users. Some stakeholders have expressed their concern over having functionality for storing users' locations and location history, while others see it as offering substantial potential for service enhancement and extra revenues.

It is possible to leave context management mainly to service providers and minimize this component in the architecture. Such a minimized component would obtain context information only when a user uses location-based service discovery and explicitly allows this information to be used. Other more complex location-based functionality and location tracking would then be the service providers' burden.

Service Management concentrates on the operation of the service in the MSF environment. It e.g.:

- stores all the configuration details related to the service (e.g. URIs of the service backends),
- stores metadata that allows, e.g. aggregated services to find automatically the most suitable underlying B2B service depending on the context of the service request (e.g. parking service covering a given area),
- knows the dependencies between services (what B2B services are needed to enable aggregated B2C service),
- possibly uses the context manager to dynamically direct service requests to different B2B services depending on the context, and
- keeps track of the service use and load.

The tasks of service management depend greatly on the role of the MSF environment in the service provision – is it e.g. actively conveying all the service requests through it or just providing identity information and keeping a log of a minimum required set of events related to the service interaction?

Service Logging logs all service-related events and requests that they are directed to MSF or conveyed through it to the event log. Depending on the information stored, this log can be used e.g. for:

- MSF system and service monitoring, and
- grounds for payments and clearing.

Payment & Clearing that can be integral part of the MSF system or realized in co-operation with some 3rd party, handles payments from the end user and deals with clearing of payments to respective service providers. It may also manage some MSF-related credit and bonus service for mobile wallets that can be used to pay for services alongside other payment methods.

In addition, the MSF centre has two main kinds of user interfaces:

Service Management UI gives service providers a user interface for managing their services. The managing tasks supported include:

- adding services and related material to the service store and managing their service offering in the service store,
- managing their customer base,
- monitoring their service use, and
- managing and monitoring the B2B service use in the aggregated services.

Admin UI offers the way for MSF operators to administrate and operate the whole MSF system.

6.2.1.2 MSF Client Framework

MSF Client Framework refers to the MSF developing and operating environment in end users' terminals. It offers an end-user interface to MSF services, as well as development kits and functionality to implement MSF compliant service frontends (e.g. applications) in end user terminals. Functionality in the MSF Client Network is described briefly below.

MSF End User UI offers a user interface to MSF services. It allows users to:

- create and manage user accounts and set user preferences,
- log in to the MSF system using their single sign-on credentials,
- discover services and inspect information related to them,
- subscribe to, launch and run services,
- use social media functionality that is related to services (ratings, reviews, discussions, community tools), and
- manage external devices connected to the user terminal.

MSF Client SDK offers enablers for the building of service frontends, called **apps** in the architecture (Figure 18). The enablers include platform independent APIs and libraries that e.g.:

- facilitate the exploitation of terminal capabilities,
- enable communication with the MSF centre, and
- offer powerful, high-level elements for service frontend UI creation.

In addition, this SDK provides guidelines, sample implementations and test environments for service frontend development.

It should be noted that the MSF architecture does not cover technical or implementation details, and therefore the nature of service frontends can vary. Section 6.4 discusses this topic further.

Device SDKs allow the development of external device connectivity for end-user terminals. They offer APIs and libraries, for example, to interact with on-board devices like OBD II or communication devices offering V2V or V2I connectivity. In this way, service frontends are able to utilize data from the users' terminal context. For example, large scale Extended Floating Car Data (XFCD) collection could be enabled by connecting the OBD II units in vehicles to smartphones with OBD II SDK, combining it to GPS data acquired from the end-user smartphone and sending it to the service provider offering real-time traffic data.

External Device Manager allows end users to manage and configure their connectivity to external devices. It allows a wireless end-user terminal to act as a communication hub that connects to devices and infrastructure in its immediate surroundings, thus extending its capabilities.

6.2.1.3 MSF Service Framework

While the MSF client framework allows efficient development of service frontends and connectivity to external devices, the MSF service frontend offers support at the service backend. It offers support for exploitation of MSF functionality and features as well as tools for service aggregation. In the following we introduce the elements in this functional block.

MSF Centre SDK provides APIs, libraries, protocol specifications, guidelines, sample implementations, and test environments for service developers to create services that are MSF compliant and gain maximum benefit from the functionality and features of MSF. This SDK is used to connect both **B2B services** and **B2C services** to the MSF environment. It also offers support for general mechanisms of service aggregation in the MSF environment.

Domain Specific SDKs concentrate on harmonizing the service development of a given service area. They define common data models, APIs, protocols, guidelines etc. especially for aggregating services. For example, FCD collection from multiple different sources requires a common data format, standards for data quality and map matching etc. Fusing FCD with other traffic data brings further requirements for compatibility. Appendix A discusses the issues related to this FCD example and proposes a minimum dataset as the basis for one potential Domain Specific SDK: FCD Collection SDK.

6.2.1.4 B2B Service Marketplace

B2B Service Marketplace is a special service for service providers who:

- offer their B2B services as building blocks for other service providers (service aggregation), or
- search for suitable B2B services to build on their own services.

It should be noted here that B2B Service Marketplace is not restricted to commercial service providers but also allows open data providers (e.g. authorities) to add their data sources and services for others.

Marketplace UI offers the frontend to all services offered by B2B Marketplace. In the architecture, these services are split into two main components as introduced briefly below.

Service Directory holds all the information related to services that could be used as building blocks by other services. The information includes, e.g.:

- a general description of the service,
- a detailed description of the contents of the service,
- coverage of the service (e.g. geographically),
- available resources / potential for service provision,
- references to APIs and other resources for service exploitation,
- target group of the service,
- terms of service use, and
- pricing or other ways for service use compensation.

The service directory has a common data model for structuring all the service-related information to make it commensurable and searchable. This information is also used in the B2B Negotiation discussed below.

B2B Negotiation tools help service providers reach agreement on their terms of cooperation for exploitation of services. According to Green (2004) the basis of this kind of negotiation has two sources:

1. The need of the consumer of the service (in the MSF context, the service provider who builds on other services)
2. Available resources of the provider (in the MSF context, the service B2B provider that offers its services for others to exploit).

Figure 20 depicts in a simplified way how convergence can be achieved in negotiations between the consumer and the provider of the service. B2B Negotiation offers tools for achieving this convergence, providing functionality to the structure, guide and partly automatize the negotiation process towards convergence and finally towards a binding Service Level Agreement (SLA) between the service provider parties.

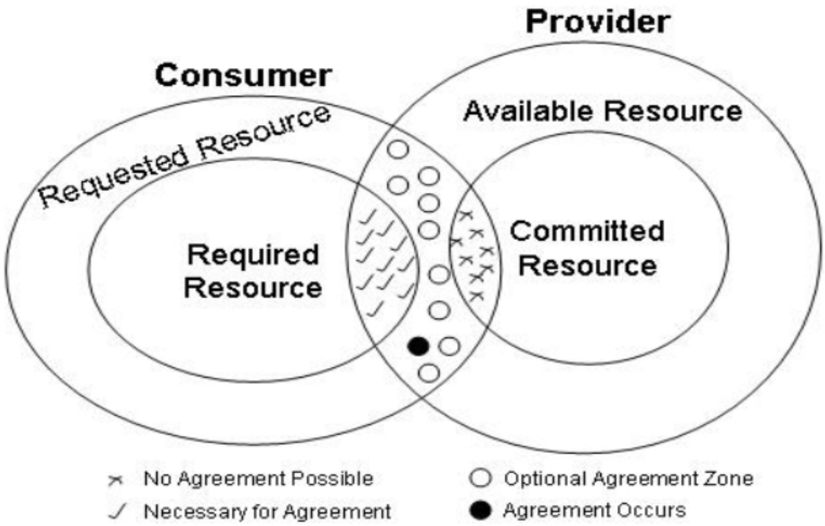

Figure 20. Negotiation convergence (Green 2004).

6.2.1.5 Core MSF Services

In addition to the functionality and features of MSF itself, there is a need for basic services in the realization of most mobility services. Such services, included in the functional block **Core MSF Services**, include e.g. **Location Service** and **Map Service**. These services could naturally be provided by 3rd party service providers in the MSF Service Framework, but certain services might be more useful when they are standardized as part of an MSF service offering. This enables harmonized service development resulting common user interaction, e.g., for navigation services throughout the whole range of services.

6.2.1.6 Communication & Interoperability

Communication & Interoperability is not a single functional component but refers to all communication management needed between the functional blocks in MSF as well as between distributed interoperable MSF systems. It includes communication protocols and APIs between the functional elements of the MSF system, management of the information about all (possibly distributed) communicating parts of the system, as well as the business logic needed to ensure the effectiveness and correctness of the communication.

The “interoperability” part of the name refers to the communication between distributed MSF systems required e.g. to enable roaming of users and services between MSF systems (e.g. covering different geographical locations).

6.2.1.7 Privacy and Security

Privacy and Security is another part of the architecture that is not covered by a single functional block. On the contrary, it consists of solutions in technical architecture, communication, encryption, privacy and security principles and practices etc. that are omnipresent in the MSF system.

6.3 Initial concept model

The functionality present in the functional architecture of MSF implicates the management of various data representing entities and processes related to mobility services in its context. In the following we introduce the initial concept model that represents the entities to be managed by MSF (see Figure 21). The model is not exhaustive, nor it is a data model that would go deeper into data structures and types. However, it can be used as a starting point for the construction of a logical data model and after that a physical data model related to data management.

The concept model in Figure 21 uses graphical notation that is familiar from UML class diagrams for relationships, but replaces the notation for classes with simple boxes containing the concept names. Next we briefly describe the entities of the model and refer to the architecture components that are central to the concepts.

Service represents the services to be provided in the MSF context. These can be B2B services offered for service aggregation in the *B2B Marketplace* in the architecture, or B2C services to be supplied in the *Service Store*.

Service Resource represents all the information and entities that are needed for service supply, service use or service exploitation in the service aggregation. Such information and entities include e.g. help files, marketing material, configuration files, APIs, libraries and guides for service exploitation etc. *Service Store* functionality is used to manage end-user oriented resources, and *Service Directory* in the *B2B Marketplace* manages the resources that are related to B2B exploitation.

Service Frontend is a special service resource that enables service users to interact with the service. It can be e.g. terminal application software or a web application. In cases where frontend software entities (e.g. applications) are managed by MSF, there is a need for configuration management for versions and releases for different platforms. In the case of web frontend, in turn, it is likely that the service provider manages the frontend elements in its own backend and only provides URIs pointing to the right frontend. Service Frontend is managed mainly by the *Service Store* component in the architecture.

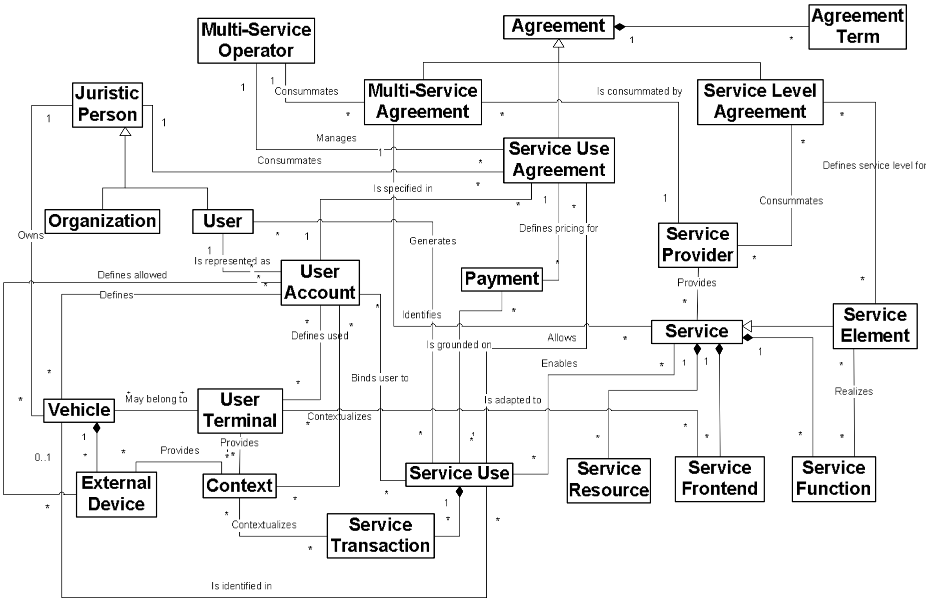

Figure 21. Core concepts and their relationships in MSF.

Service Function refers to the functional elements of the service that are realized by exploitation of other B2B services that are represented in this concept model by the **Service Element** entity (specialized from service entity). These concepts are referred to and managed in *B2B Marketplace*, *Service Management* as well as *MSF Service Framework* and its *MSF Centre SDK*.

Service Provider represents an identity that is providing or supplying services in the MSF context. It is naturally related to the service concept and to agreements made between the service provider and multi-service operator and other B2B service providers. Of these agreements the first one, **Multi-Service Agreement**, refers to the service provision or supply agreement in the context of MSF and the latter one to the **Service Level Agreement** that is consummated between the service aggregators and B2B service providers. Service providers themselves are one kind of (complex) identity managed by the *Identity Management* component in the architecture. Multi-service agreements, where the service provider agrees on the terms of its operation in the MSF context, relates also to identity management. Service level agreements (SLAs) are consummated with the help of the *B2B Negotiation* functionality in *B2B Marketplace* and the agreement is then managed by the *Identity Manager* for each agreement party.

The third kind of agreement in the core concept is **Service Use Agreement** that contains the agreement with the terms of service usage (including pricing) by the end user. In this model it is not consummated by the service provider entity but the **Multi-Service Operator** entity that represents the operator who is legally responsible for the MSF operation. This solution reflects the idea that the multi-service operator acts on behalf of the service provider in this agreement consummation. This, in turn, means that service providers in their agreement with the multi-service operator (Multi-Service Agreement) adapt their service terms to the standard way of expressing service terms for end users. *Service Use Agreement* is an entity that is closely related to *Service Store* and *Identity Management*, managing also most of the user-related information in the functional division in the architecture.

All three agreement types are specializations of the **Agreement** concept, which represents any kind of agreement to be managed in the MSF context. **Agreement Term**, in turn, refers to a collection of terms that are included in each agreement.

Service Use Agreement relates to the user of the service. However, the entity related to the agreement is the **Juristic Person**. In this way, agreements can be made both with organizations represented as the **Organization** entity, and with private end users denoted here by the **User** entity. User management performed by the *Identity Management* component in the architecture is thus more complex than management of single user accounts.

The User entity represents the actual end user interacting with the services. The end user is the person having a user account (**User Account**), which is an entity representing the user with all user-related static and partially also dynamic information for the system. Thus, the User Account entity is connected to:

- **User Terminal**, which identifies and defines the characteristics of the end user terminal
- **Vehicle**, which defines the vehicle(s) used by the end user
- **External Device**, which defines the external devices that the user has allowed to connect through the end-user terminal.

These user-related entities are managed by *Identity Management*, where the user has access through the *MSF End User UI* in the *MSF Client Framework.* In addition, with the same UI the end user may control the connectivity to external devices using the *External Device Manager.*

Context holds information about the surrounding environment of any identity (e.g. User, User Terminal, Vehicle and External Device). It can be determined directly or indirectly with the help of devices connected to the MSF Client Framework. The most typical context information is location, which is the core information for most mobility services run in the MSF context. Contextual information on identities is managed by the *Context Manager*.

Service Use is a compound element that connects users to the actual usage of the particular service. Usage of the service is allowed after the user has accepted the service use agreement (thus connection to the Service Use Agreement entity). The service consists of service-related events logged by the *Service Logging* component. These events are represented by the **Service Transaction** entity in the concept model. It should be noted that the use of MSF functionality is also included in this logging. Service Transaction may also be connected to the Context entity to contextualize the service use history. Service use also gives the grounds for payment (the **Payment** element).

6.4 Alternatives for realization

In the previous sections we introduced the functional architecture for MSF along with the core concepts that should be managed in this context. As pointed out earlier, this architecture does not commit itself to any particular technical solution. In addition, the functionality included in the MSF architecture can be realized only partially or in a lightweight manner. In the following we discuss some topics related to realization of the MSF architecture.

6.4.1 Architecture topologies

The MSF environment can be realized as a monolithic platform offering all the functionality and operated by a single actor (that we have called Multi-Service Operator in Section 6.1). However, the MSF environment can also be realized as a result of multiple co-operative parties that each provides some part of the MSF functionality. The latter is called a service-oriented solution. Figure 22 is a simplified illustration of the difference in architecture topologies between the two.

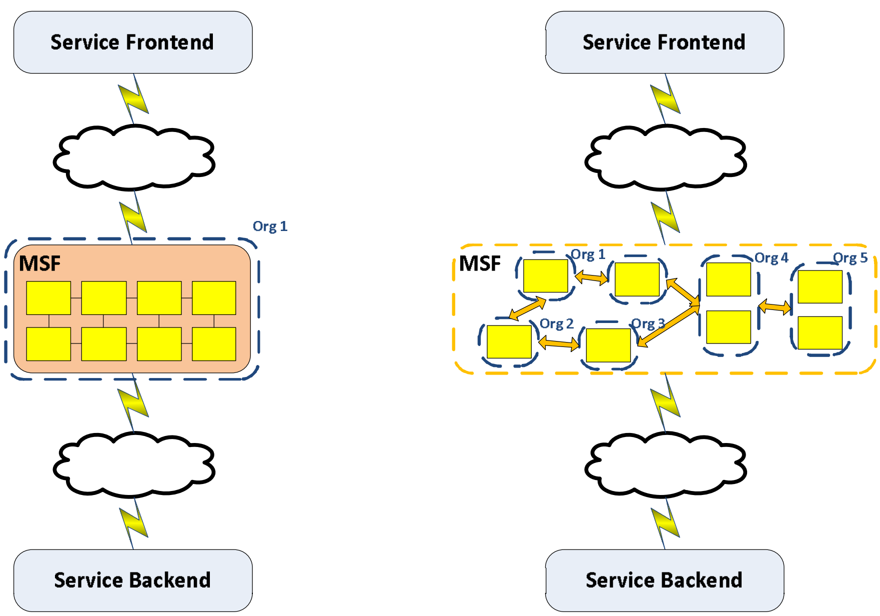

Figure 22. Monolithic vs. service-oriented approach for the realization of MSF environment.

The service-oriented approach offers several advantages over the monolithic system, including the following:

- **Loose coupling** of the functionality loses e.g. technological and organizational constraints in the system design, gives more freedom of implementation of the functionality as well as integrating legacy systems.
- **Distribution** allows different parts of the system to be operated by different organizations (and in different locations), and thus allows selection of the best choice for the task specific to a given functional part.
- **Location transparency** related to distribution, in turn, makes different parts of the system agnostic to the location of resources.
- **Reusability** that comes from a properly modularized and designed architecture allows the same service blocks to be used in the realization of multiple different systems and services.
- **High availability and scalability** is based on the flexibility of the distributed, loosely coupled system that eases e.g. clustering of individual service blocks, adding redundancy, providing load-balancing etc.

On the other hand, the monolithic system can usually be implemented to provide better performance, as a service-oriented design tends to include overhead caused by network communication, data marshalling and standard bulky message formats. Providing a holistic security solution for a service-oriented approach is another challenge that can be more easily tackled in the monolithic system. Because SOA involves distributed components in a heterogeneous environment and may require distributed transactions, achieving high reliability is also challenging.

In the case of MSF, it is probable that a balance between the monolithic and service-oriented approach is the way to go. Certain parts of the system might be better to operate as a monolithic system for performance and security reasons (for instance user management and context management), while certain services could be obtained from third parties (e.g. payment services using well proven, standard secure approaches).

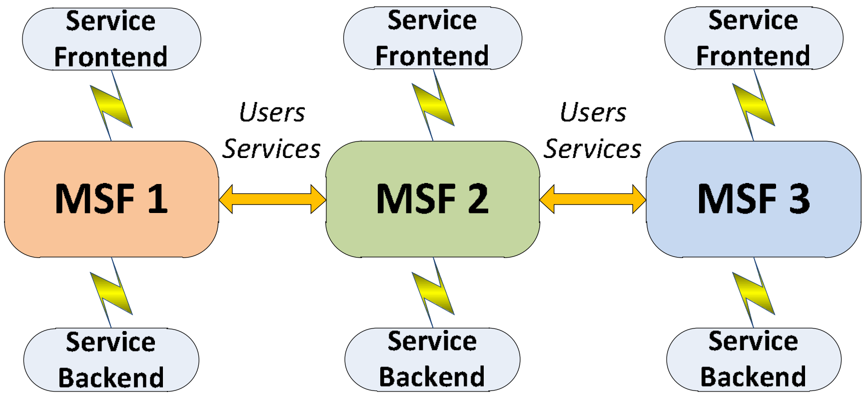

Figure 23. Distribution of the MSF environment.

Another dimension to the architecture topology is the distribution of MSF systems and roaming of core data between them (Figure 23). This topic is relevant for many reasons. First, it is desirable that MSF is operated by several players in order to enable a large variety of mobility services and competition on the market. Second, different multi-service operators can specialize in enabling certain kinds of services rather than acting as general purpose enablers, thus being able to serve their target groups optimally. Third, scaling MSF geographically (e.g. across national borders) to serve mobile users everywhere usually requires distribution of physical setups as well as adding new multi-service operators.

6.4.2 Tight vs. loose coupling of services to Multi-Service Facilitation

The degree of coupling depends on the richness of the support offered by MSF. Such desired central features like single sign-on or payment and clearing already requires that every service is indeed coupled with MSF at least in some degree. In this minimum case, MSF acts as an authentication server and payment service. Number of such solutions already exists and interested reader may consult a list of single sign-on implementations in Wikipedia (2013).

In the realization of the MSF environment one of the most important questions is: how tightly the mobility services should be coupled with the MSF system? Figure 24 illustrates two extremes of coupling. In the first case (left in the figure) service frontends communicate with the respective service backends solely through the MSF system. In the second case, only minimum control data is communicated with the MSF system, while practically all the service requests are communicated directly between service frontends and backends.

Functionally there are not very big differences between the tight or loosely coupled models. Clients should discover suitable services according to use context in any case by using some directory service and after that service related information is flown trough the MSF or directly between backend server and the service front end.

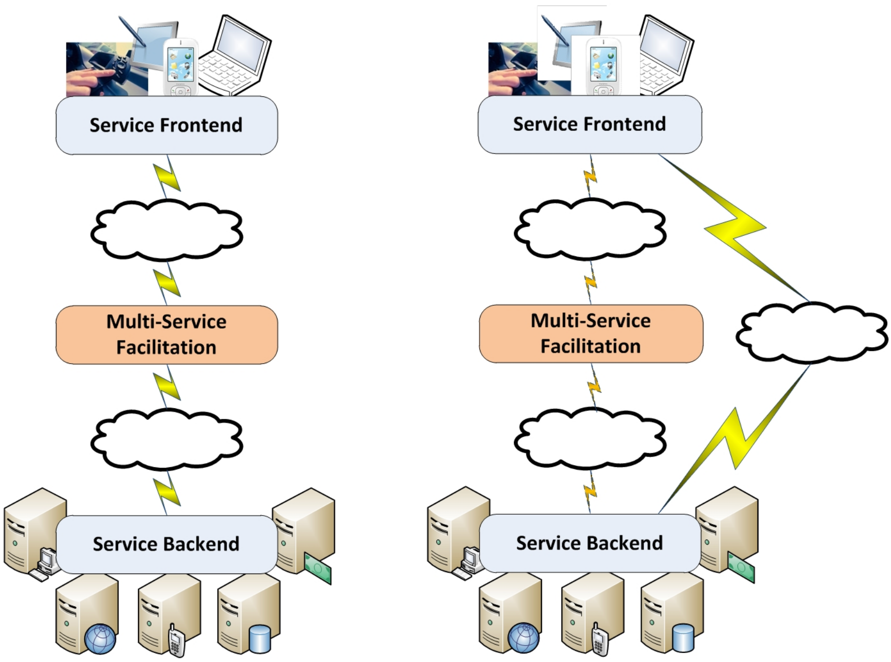

Figure 24. Tight vs. loose coupling of services with MSF.

When all traffic from Service Frontend to Service Backend flows through the MSF, MSF is working as a proxy server. A proxy server is a computer system or an application that acts as an intermediary for requests from clients seeking resources from other servers. Proxy servers can be used for several purposes:

- To keep machines behind the proxy server anonymous mainly for security reasons. Proxy can also anonymize request from Service Front end clients before those are send to the backend servers.
- To speed up access to resources by using caching. Caching proxies keep local copies of frequently requested resources significantly reducing bandwidth usage and costs and increasing performance by preventing downloading the same content multiple times from the backend service.
- To scan transmitted content for malware before delivery and to scan outbound content, e.g., for data loss prevention. Proxy can also filter and adapt contents according to the information the requester provides.
- To translate languages to localize contents for different markets. Traffic from global audiences is routed through the translation proxy to the source site.
- To log / audit usage of backend services.

From MSF point of view the logging and auditing of usage and the anonymization are the most interesting features that MSF as a proxy – functionality can provide. Logging and auditing feature makes identity management, single sign on and indisputable usage accounting for services (and in turn charging) more ease to realize and follow up. For context dependent services, the proxy functionality can be used to supply language translations for localized services.

When all traffic goes through the MSF, service operator might collect additional information of services usage in general. Many of traffic related service requests contain location information that can be collected by the proxy for various purposes. On the other hand, privacy concerns have been issued when discussing traffic related services. By using a trusted MSF proxy service generating surrogate ids, backend services might not need to know the real identity of service users and still service providers could trust that the accounting of usage of service is correct.

As pointed out MSF as a proxy server might have several advantages it is not without drawback though, for example it creates a single point of failure if not designed and distributed correctly.

6.4.3 Front-end development in Multi-Service Facilitation environment

Three different approaches are MSF front-end solutions are presented in Figure 25. Common key elements of the frontend solutions are indicated by using different colours. *Terminal Runtime Environment* marked as light blue is used here as a general term for computing environment that provides support for user interaction with services in MSF framework. It is a combination of hardware and software that is materialized in a form of mobile phone, a tablet or a laptop/PC executing an operating system like Apple IoS, Windows, Android etc. Terminal capabilities may vary considerably, e.g., with respect to processing power, screen size and input methods. In addition, the support for applications offered by the terminal operating system API's and libraries might also vary.

The green coloured *MSF End User App/UI/Web App* is a component provides an end user interface for communicating with MSF Centre, e.g., for service discovery, management of own services and user account management. MSF applications (named *MSF Apps*, *Terminal Apps* or *Web Apps* depending on the approach), in turn, are denoted with orange colour represent service specific frontends that connect users with dedicated mobility services like multimodal trip planners, parking services etc. These frontends are explicitly downloaded and installed by the user or, in the case of Web Apps, executed as a part of the fetched web page (e.g., as a combination of HTML and JavaScript).

Apps in turn use different Application Programmes Interface (API) libraries (dark blue and medium blue) in varying ways depending on the approach. *MSF API & Libraries* refers to the MSF specific API that offers MSF specific functionality and hides Terminal APIs & Libraries with common harmonized platform independent terminal API. *Ext. Device APIs* provides APIs to external devices that can be accessed via the end user terminal. *Terminal APIs & Libraries*, in turn, refers to the APIs and libraries offered by the native end user terminal platform.

Variations of the possible configurations in the realization of the frontends are vast. In this context, however, technical details are abstracted and only the main properties of these approaches are discussed.

The first alternative “Service front-ends as MSF specific apps” can be characterised as a “virtual environment” approach. MSF Client is a program unit that contains the MSF End User interface for seeking and installing MSF applications that are executed in a MSF specific execution environment. Differences between terminal environments and operating systems are hidden by this MSF execution environment (indicated as olive green) that is common for all MSF applications. This simplifies the application developer's work – theoretically, only one platform independent frontend is needed to be implemented for each service. This is done by exploiting platform independent MSF API & Libraries and Ext. Device APIs alone and the frontend application is runnable in all environments, to which MSF Execution Env. is ported and capable to run on. In reality, while the application logic can remain the same in different devices the solutions, e.g, for user interaction may require user interface adaptations due to variations of screen sizes and input methods. This approach resembles Sun/Oracle’s idea of Java: “write once, run everywhere” that didn’t never fully become materialized – in fact the fragmentation of different Java environments like Java Micro Edition (J2ME) targeted to mobile devices etc. is widely known. Another problem of this approach is the burden related to the MSF Execution Environment: substantial efforts for development and maintenance are mandatory to keep that up to date in the heterogeneous and rapidly changing terminal scene. On the other hand, in the successful case this burden would be lifted away from the App developers.

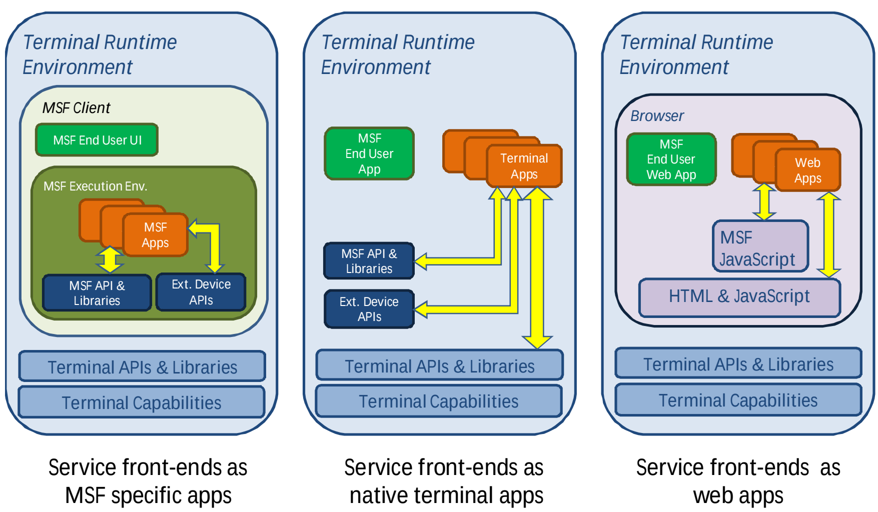

Figure 25. Approaches for front-end development.

The second alternative “Service front-ends as native terminal apps” represents a familiar approach from the current mobile ecosystems environments like iPhone, Android and Windows. Each service offered in the MSF framework has its own frontend implemented as an operating system dependent native application relaying on the specific operating systems (e.g. Apple iOS, Android or Windows Phone). These Terminal Apps (see the sub-figure in the middle of Figure 25) communicate with backend services (server) by using service related libraries (available in the backend; not presented in the Figure), utilize MSF API & Libraries as well as Ext. Device APIs adapted to each terminal platform, and rely heavily on the libraries provided by the particular operating system. In this case, MSF End User App providing the interface to MSF Centre is application among the other applications (and downloadable from applications stores like ITunes, Google Play or Windows Store). After installing the MSF End User App, MSF specific service frontends (implemented as native applications) may be installed from dedicated “MSF app store” (Service Store in the MSF Architecture) for different operating systems or directly from service provider’s web site. MSF End User App might not even be obligatory for the service discovery since the own operating system of the device may support application search and management functions like installations and automatic updates when a new version of an app is available. Unfortunately current generic app stores like Google Play or Apple App Store have so vast offerings (around 1 million apps) that MSF specific apps could be hard to find without a proper support. Also user management and logging on services with a single sign-on principle is difficult to implement without the MSF End User App.

The main advantage of this “native apps” approach is that it generally enables the best possible user experience for services (e.g. a platform related native user interface), best application performance, abilities to fully utilize terminals own resources and a certain degree of autonomous functionality (if application is correctly designed) when data communications are non-existent or not properly working. The main drawback is that several parallel (and costly to maintain) terminal specific versions of each frontend for different operating environments should be offered if maximal accessibility to services is needed to be guaranteed.

The “Service front-ends as web apps” approach utilize the fact that web browser environments are pretty standardised having only minor differences to tackle on. All the browser environments of the terminal devices are based on standardized HTML and nearly all of them are capable of executing JavaScript. For that reason, frontends could be implemented as a combination of JavaScript, HTML and Cascading Style Sheet (CSS) packages downloadable from the MSF server sites or service backends. Frontend interfacing could be eased by providing specific MSF JavaScript libraries and graphical interfaces could utilize generic JavaScript libraries like Jquery, Prototypes or MooTools etc. MSF End User Web App allowing user interaction with MSF Centre can be used to conduct user actions, e.g., for logging in, searching and selecting suitable applications for the use context. MSF services might not fully utilize terminal environments capabilities but usually access to location (e.g. GPS) information is provided and communicating with Bluetooth devices is enabled in some environments like Android.

The main advantage of this approach is that commonly available web designer skills are applicable for nearly all devices, browsers environments are today even more standardised due to acceptance of HTML5 and maintaining apps in user devices are more straight-forward that in other alternatives. Some drawbacks still exists. For instance, the web approach suffers from performance penalty and limited access to terminal resources (depending on a device and an operating system). In addition, user interaction design is limited in the case of special requirements. Despite of these limitations, amazingly smooth interfaces to services could be implemented.

It should be noted that all of these possible technical approaches could not be applied in practise in every environments due to commercial restrictions even though those could be technically feasible. For example, Apple's App store model doesn't allow applications that download apps and execute them inside the original application (in our case the MSF Execution Env.) due to the commercial (and security) reasons.

7. Conclusion and next steps

In the Multi-Service Model mobility related data and service providers are collected together to provide their services in a one stop shop for their users. The same model aims also to create ecosystem that enables enhanced co-operation in the service creation and provision by using technology solutions. The SUNTIO2 project has collected stakeholder aspirations with respect to the Multi-Service Model and surveyed experiences from other similar endeavours. In addition, it has analysed value chains and mobility service use cases. This work was used to deduce requirements for the realization the Multi-Service Model with the help of a special Multi-Service Platform. These requirements were elaborated to functional architecture and initial conceptual data model to guide the further work towards concrete realizations.

During the work in SUNTIO2, some key issues came up repeatedly and they are listed in the key findings below. In addition, the further steps in the realization of the Multi-Service Model are presented.

7.1 Key findings

The findings from the architectural perspective are the following:

- The end-user/consumer side is already quite well supported, e.g. in the smartphone app store model (but the apps in current smartphone ecosystems are not interoperable, which is a serious flaw from the end-user's viewpoint).
- There is a clear need for cooperative business development support like traffic data marketplaces and agreement tools between partners in the creation of value networks.
- The privacy and security solutions as well as ways of action increasing end users' trust are central for the success of the mobility services.
- Location-Based Services (LBS) are converging with ITS and mobile terminals like smartphones will increasingly serve as user terminals and – what is remarkable – as data sources in mobility services.

Conclusions related to data:

- Standardized data models, protocols and APIs for different types of traffic related data are needed to accelerate data markets. Common models are needed for raw data (e.g., FVD data collection), processed data (e.g., traffic situation) as well as roaming of the services in the distributed (possibly multi-operator) multi-service realizations.
- The data has at least two dimensions that should be considered in the development: a) business related chargeable data with revenue sharing models and strict rules for their use (can contain personalised data) and b) open data to be shared and used in development communities freely.

Conclusions related to the multi-service business and value network:

- A single company or small consortia cannot produce all the relevant services and applications. The multi-service value network must be a growing and developing community. It has to find and attract stakeholders and operators for the provision of large range of services as well as for the other tasks related to the operationalize the Multi-Service Model.
- Data gathering, processing and mediating and the growing value-network need a B2B service marketplace with relevant tools to support cooperation and business development. Open data and tools inspire newcomers and new developers to join the ecosystem, but the real effort will be in the creation of business.
- The system realizing the Multi-Service Model should support single sign-on and single payment models for end users as well as payment clearing and revenue sharing functionality for service and content providers.

The best way to develop and promote the Multi-Service Model is to realize the functional architecture like the one presented here – even if just partially with some core functionality – as well as to create, test and produce new services in a multi-service frame of reference. In this way, all the relevant challenges will be handled and innovations and best practices can emerge. Also participating in international projects advancing the objectives of the Multi-Service Models (e.g., the on-going MOBiNET project) and following the international development of the open platforms, multimodal services, data markets and ecosystems will help the creation of multi-service.

7.2 Next steps

The **roles** of the multi-service value network must be developed further with the support of business tools. We have to find stakeholders and partners to populate the key roles in multi-service provision. The multi-service operator is a central actor and an activator that draws different partners into cooperation.

Creation and promotion of **standards and common agreements** are required in many levels in order to make the service repertoire to grow and diversify (e.g., data roaming, service delivery, B2B cooperation, service-level and other agreements, data between buyers and sellers).

There is a need for a **B2B marketplace** where stakeholders (data and service providers, multi-service operators and other stakeholders) can contact each other, co-operate in the service creation, test their service offering and end up effortlessly to service level agreements.

Multi-service will grow gradually; examples in foreign countries show that eco-systems with the most potential are those that contain **both private and public services**. Thus all efforts related to creating public services such as road-tolling, emergency services, situational-awareness building by authorities, safe truck parking etc. should be related to commercial services (navigation, fleet management, Point-of-Interest and touristic services etc.). There are no grounds for separate devices and services in this setting.

Services that need tracking and tracing, location and communication to backend systems should be integrated into a common platform. This cannot be done without forgetting privacy and the need for secured transmission. Hence, **privacy and security** are of paramount importance to be considered in the further development steps.

Steps to building a multi-service:

1. Create innovative ideas for services and contents (“pile of services”/ “stack model”), preferably a combination of public/private services.
2. Seek out potential customers and markets (connected to the above; ideas should be beneficial and address real needs).
3. Define and agree on the roles and relationships in the value network.
4. Develop business models (especially revenue/cost sharing models and contracts between partners).
5. Define the connecting elements (architecture, interfaces, standardization of data, instructions and tools etc.).
6. Make Proof-of-Concepts and test the service processes.
7. Finally go for Service Delivery, Marketing and Sales.

References

Aakre, A., Appel, K., Kronborg, P., Wendelboe, J. T. & Yde, P. 2012. ITS Terminology, NVF-rapport (ITS-utskottet) 3/2012. http://nvfnorden.org/lisalib/getfile.aspx?itemid=5582.

Alibaba 2013. Alibaba.com. http://www.alibaba.com/. Accessed 15.10.2013.

Ariba 2013. Ariba – An SAP Company. http://www.ariba.com/. Accessed 15.10.2013.

Beckmann, D. Schlauch, T. & Thielen, D. 2011. WP4.1 – Global architecture definition & requirements, Instant Mobility project.

Bossom, R. et al. 2010. European ITS Communication Architecture – Overall Framework, Proof of Concept Implementation. COMeSafety Deliverable D31. Version 3.0, 03.02.2010.

Burkert, A., Milli, M., Denis, Y. et al. 2010. Detailed Design of the Integrated Open Platform, Deliverable D3.1, Version 1.0. VIAJEO – International Demonstrations of Platform for Transport Planning and Travel Information, 6.10.2010.

Business Insider 2013. How Location-Based Services Are Transforming The Mobile Industry. Business Insider, May 22, 2013, 11:31 AM. http://www.businessinsider.com/location-data-is-transforming-mobile-2013-4. Accessed 21.10.2013.

Böhm, M. & Lamprecht, G. 2012. In-Time–Intelligent and efficient travel management for European cities. http://www.in-time-project.eu/download/presentations/in_time_project_presentation_general.pdf. Accessed 10.10.2013.

Datex 2013. Datex II web-pages. http://www.datex2.eu. Accessed 14.6.2013.

Faezipour, M., Nourani, M., Saeed, A. & Addepalli, S. 2012. Progress and challenges in intelligent vehicle area networks. Communications of the ACM, 55(2), pp. 90–100.

FRAME 2013. The home of the FRAME architecture. Peter Jesty Consulting Limited. http://www.frame-online.net. Accessed 25.6.2013.

Green, L. 2004. Service level negotiation in a heterogeneous telecommunications environment. International Conference on Computing, Communications and Control Technologies (CCCT04). Austin, TX, USA, August 2004.

Helsingin kaupunki 2013. Älyliikenne Helsingissä. Helsingin kaupunkisuunnitteluviraston liikennesuunnitteluosaston selvityksiä 2013:3.

Herrala, M. 2007. The value of transport information. VTT Research Notes 2394. VTT, Espoo. http://www.vtt.fi/inf/pdf/tiedotteet/2007/T2394.pdf.

Hoyer, V. & Stanoevska-Slabeva, K. 2009. Business Models for Digital Business Ecosystems: The Case of the Open Negotiation Environment (ONE) Platform. 3rd IEEE International Conference on Digital Ecosystems and Technologies, 2009. DEST '09.

Iansiti, M. & Levien, R. 2004. The Keystone Advantage: What the New Dynamics of Business Ecosystems Mean for Strategy, Innovation, and Sustainability. Harvard Business Press. 255 p.

Info24 2013. Profit from taking physical products and resources online. http://info24.eu. Accessed 15.10.2013.

Instant Mobility 2013. Instant Mobility – Multimodality for people and goods in urban areas. http://www.instant-mobility.eu/. Accessed 15.10.2013.

In-Time. 2010. D3.1.1 – In-Time System Architecture, version 0.3, 12.5.2010. In-Time – Intelligent and Efficient Travel Management for European Cities. http://www.in-time-project.eu/download/publicdeliverables/d_3_1_1_intime_system_architecture_v03.pdf. Accessed 26.6.2013.

In-Time 2013. In-Time, Intelligent and efficient travel management for European cities, project home page. http://www.in-time-project.eu/. Accessed 21.10.2013.

ITS international 2013. The move towards shared telematics platforms. http://www.itsinternational.com/categories/networking-communication-systems/features/the-move-towards-shared-telematics-platforms/. Accessed 23.10.2013.

Kochs, A. & Ansorge, J. 2006. Requirements and Service Analysis. eMotion Deliverable D1, Final 1.0, October 2006.

Koistinen, M. 2011. Tilannetietoisuus ja tilannekuva operatiivisessa liikenteenhallinnassa. Diplomityö. Yhdyskunta- ja ympäristötekniikan laitos, Insinööritieteiden koulu, Aalto-yliopisto. http://civil.aalto.fi/fi/tutkimus/liikennetekniikka/opinnaytteet/diplomityo_koistinen_2011.pdf.

Liikenne- ja viestintäministeriö 2013. Liikenteen ja viestinnän avoin tieto (Open data in transport and communications). Publications of Ministry of Transport and Communicarions of Finland. 10/2013. http://www.lvm.fi/c/document_library/get_file?folderId=2497123&name=DLFE-19419.pdf&title=Julkaisuja%2010-2013. Accessed 26.6.2013.

MDM 2013. Mobilitäts Daten Marktplatz. http://www.mdm-portal.de/. Accessed 15.10.2013.

MLIT 2012. Vehicle-Infrastructure Cooperative System and Probe Data in Japan. Road Bureau (October 2012). Ministry of Land, Infrastructure, Transport and Tourism. http://www.mlit.go.jp/road/ITS/pdf/CooperativeSystemandProbeData.pdf.

MOBiNET 2013a. MOBiNET web site, http://www.mobinet.eu/.

MOBINET 2013b. MOBiNET, Europe-Wide Platform for Cooperative Mobility Services, Annex I – "Description of Work", 2013-05-15.

Moore, J.F. 1993. Predators and Prey: A New Ecology of Competition. Harvard Business Review, May–June 1993.

Moore, J.F. 2005. Business ecosystem and the view from the firm. The Antitrust Bulletin/Fall 2005. 58 p.

Mohammad, S.A., Rasheed, A. & Qayyum, A. 2011. VANET architectures and protocol stacks: a survey. Nets4Cars/Nets4Trains'11. Proceedings of the Third International Conference on Communication Technologies for Vehicles. Springer-Verlag Berlin, Heidelberg. Pp. 95–105.

Mäkinen, P., Ruoti, K., Lähesmaa, J., Lehtonen, M., Oinas, J., Ristola, T. & Appel, K. 2000. Liikennetelematiikan kansallinen järjestelmäarkkitehtuuri – TelemArk. Arkkitehtuurikuvaus. Liikenneministeriön mietintöjä ja muistioita B 5 /2000. 94 p. http://www.transportal.fi/Hankkeet/tetra/TETRA/H8/t8_15.pdf.

NDW 2013. National Datawarehouse for Traffic Information. http://www.ndw.nu/en/. Accessed 15.10.2013.

Object management Group 2013a. MDA – The Architecture of Choice for a Changing World. http://www.omg.org/mda/. Accessed 23.10.2013.

Object Management Group 2013b. UML® Resource Page. http://www.uml.org/. Accessed 23.10.2013.

Osterwalder, A. & Pigneur, Y. 2009. Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. Wiley. 288 p.

Piarc 2004. The Big shift. The Move of European road administrations towards network operations – version 2.0. August 2004. Piarc org.

Pilli-Sihvola, E., Auvinen, H., Tarkiainen, M. & Hautala, R. 2011a. Paikkasidonnaiset liikenteen palvelut – Palveluiden nykytila. VTT Working Papers 174. VTT, Espoo. http://www.vtt.fi/inf/pdf/workingpapers/2011/W174.pdf. Accessed 21.10.2013.

Pilli-Sihvola, E., Tarkiainen, M., Vilkman, A. & Hautala, R. 2011b. Paikkasidonnaiset liikenteen palvelut – Teknologia ja arkkitehtuurit. VTT Working Papers 173. VTT, Espoo. http://www.vtt.fi/inf/pdf/workingpapers/2011/W173.pdf. Accessed 21.10.2013.

Porter, M.E. 1985. The Competitive Advantage: Creating and Sustaining Superior Performance. The Free Press, New York.

Salovaara, A., Appel, K., Pihlajamaa, O. & Vilkman, A. 2012. Liikenteen monipalvelumalli. Liikenne- ja viestintäministeriön, Liikenneviraston ja Trafin teettämä selvitys.

Stevens, A. 2012. Road operator business models for cooperative ITS. Presentation in ITS Word congress Vienna 2012.

Siponen, A., Higgins, A., Lehtonen, M.J., Levo, J., Lähesmaa, J., Mäkinen, P. & Öörni, R. 2005. Ajantasaisen liikennetiedon arkkitehtuuri. Loppuraportti, VIKING MIP 200 5: Real-time Traffic Information Architecture Final Report. AINO-julkaisuja 20/2005. 117 p. In Finnish. http://www.durbit.se/Archives/ExternalPDF/MIP2005/Domain8/Deliverables/VIKING_MIP05_D8_1_S_96_Real_time_TI_architecture.pdf.

Tiehallinto 2005. Liikenteen hallinnan arkkitehtuuri – Tavoitteellinen järjestelmäarkkitehtuuri. Tiehallinnon sisäisiä julkaisuja 45/2005. 32 p. In Finnish. http://alk.tiehallinto.fi/julkaisut/pdf/4000481-v-liikhall_jarjarkkiteht.pdf.

Vilkman, A., Hautala, R., Pilli-Sihvola, E., Leviäkangas, P. & Intosalmi, J. 2010. PASTORI project defines the future ITS services portfolio. ITS World Congress 2010, Busan. http://www.vtt.fi/files/sites/pastori/Pastori_Busan.pdf. Accessed 21.10.2013.

Vilkman, A., Hautala, R. & Pilli-Sihvola, E. 2011. Paikkasidonnaisten liikenteen palveluiden liiketoimintamallit – Edellytykset, vaihtoehdot, haasteet ja mahdollisuudet. VTT Working Papers 175. VTT, Espoo. http://www.vtt.fi/inf/pdf/workingpapers/2011/W175.pdf. Accessed 21.10.2013.

Wikipedia 2013. List of single sign-on implementations, Wikipedia. http://en.wikipedia.org/wiki/List_of_single_sign-on_implementations. Accessed 25.6.2013

The World Wide Web Consortium 2013. XML Schema. http://www.w3.org/XML/Schema. Accessed 23.10.2013.

Appendix A: Floating Car Data Collection

A1. Introduction

The City of Helsinki and Finnish traffic authorities have expressed a need for real time traffic situation snapshots for various purposes, and several Finnish companies have voiced an interest in producing floating car data (FCD) from their fleets to provide a better real-time picture of traffic in the Helsinki capital area. Another motivation for providing such information is to boost emerging traffic data markets in Finland as a part of value-added Intelligent Traffic System (ITS) services. This appendix attempts to explain some technical foundations and challenges related to the FCD collection based on the cellular and global navigation satellite system (GNSS), and what kinds of quality factors should be taken into account when using this for traffic data acquisition purposes.

In 2005, a preliminary study called "Producing a real-time traffic state model using floating car data"[1] (Kosonen & Pahlman 2005) examined the feasibility of xFCD as a source of traffic flow measurement. The study was part of the AINO programme funded by the Ministry of Traffic and Communications. It introduced the FCD topic, sketched several possible technical solutions, made some economic calculations, and presented different traffic modelling methods. The report also briefly mentioned possible applications based on FCD and problems related to FCD technologies like initial costs & privacy issues. The report did not make any practical suggestions on how to proceed, nor did it define any detailed requirements for xFCD information.

In 2011, VTT Information Technology in co-operation with Nokia Siemens Networks and Sonera performed an FCD pilot in the Espoo area neighbouring Helsinki. The project aimed to acquire travel time information from data provided by mobile phones. The results were not as encouraging as hoped; the main conclusion was that for traffic management operations, monitoring of cell handovers of active 2G phone calls did not produce a high enough number of observations (Innamaa & Hätälä 2012). Another related study (Innamaa 2012) also indicated that Bluetooth-based monitoring showed that the quality of Bluetooth-based travel time data was acceptable. Surprisingly, no similar projects for evaluating the feasibility of a satellite-based traffic information collection system have been suggested or considered so far – despite the fact that the Finnish road toll metering system is considered to be GNSS (GPS/Galileo) based.

The flowing chapters introduce collection methods for basic road traffic information, then clarify the principles of cellular- and satellite-based FCD acquisition. Some FCD-based research projects and companies that provide FCD data are mentioned, and emerging future trends are discussed. The final section summarises our conclusions on FCD as a traffic data collection method.

---

[1] In Finnish: "Ajantasaisen liikennetilamallin tuottaminen floating car -datan avulla."

A2. Road Traffic surveillance technologies

One of the key elements of ITS is probing traffic flow by several means including sensors and visual observations based on traffic camera systems. Traditionally, traffic information data collection is based on stationary sensors that are either embedded into roads or set up along roadsides. These sensors are classified as in-roadway and over-roadway depending on the mounting system. In-roadway sensors are embedded in the pavement or subgrade or otherwise attached to the road surface. In-roadway sensors are sometimes considered intrusive because their installation and maintenance create disturbances to traffic. Examples of such devices include magnetometers, inductive loops, piezoelectric cables etc. (Mimbela & Klein 2007, Leduc 2008).

Over-roadway sensors are mounted above or alongside the roadway and are considered non-intrusive because their assembly and maintenance do not create prolonged disruption of road traffic (especially alongside units), and maintenance/replacement is easier than for in-roadway sensors. Machine vision systems, microwave radars, laser radars, ultrasonic and infrared sensors, passive acoustic arrays are typical examples of nonintrusive sensors (Mimbela & Klein 2007).

The general data collection process of these sensors starts with the transducer (a sensing element) detecting the passage or presence of a vehicle. A signal-processing unit converts the transducer's output into electrical signals, which in turn are converted by a data-processing device into traffic parameters like presence, count, speed, weight, travel time etc. A network interface sends the parameters to an appropriate destination for further processing and dissemination.

Although these traditional measuring technologies are well proven (Leduc 2008) and their merits are indisputable, they also have notable drawbacks. The main problems of a fixed sensor infrastructure are limited coverage and high establishment and maintenance costs. Other minor drawbacks are impaired sensing abilities in certain weather conditions like strong winds, cold weather, heavy rain, snow and fog. Especially over-roadway sensors are susceptible to these circumstances.

Advances in cellular communication technologies and satellite navigation systems over the past few decades have made it possible to use alternative technologies to collect traffic data. These methods are commonly called floating car data (FCD) or floating vehicle data (FVD). FCD refers to a technology that collects information on the state of traffic from a set of individually recognizable vehicles that float in traffic (like a cork in a river flow). At least three methods of FCD data probing exist: that based on GNSS, short-range radio communications, or the cellular network system.

With the GNSS-based method a set of vehicles is equipped with satellite receivers that periodically transmit the vehicle's current location (e.g. coordinates) to the data collection centre through on-board radio units or cellular networks like 3G. Typically satellite probe devices are mounted in cars driving in fleets, such as taxi or other professional services like couriers, transportation companies, etc. However, also ordinary satellite navigator (PND, Personal Navigation Device) users can participate in the data collection process.

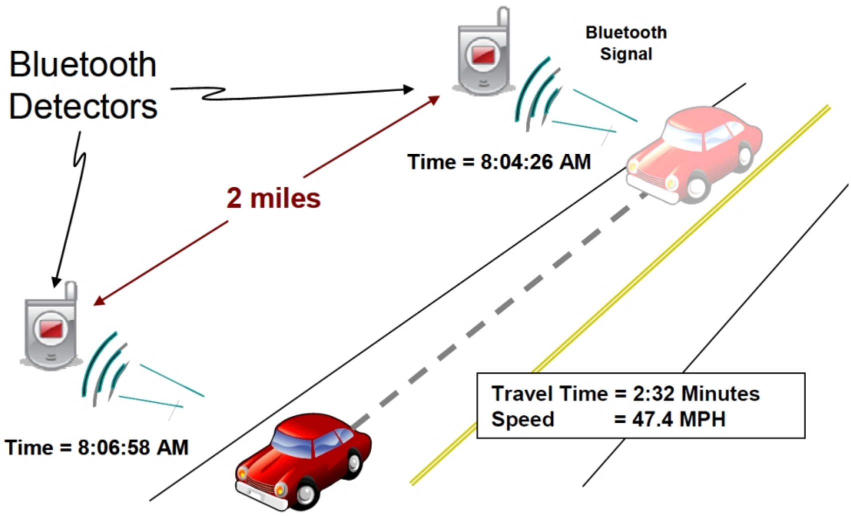

Figure A1. Bluetooth traffic monitoring operation concept (University of Maryland 2008).

Another set of FCD data collection methods utilizes short-range radio communication technologies like RDIF and Bluetooth (Puckett 2010). Bluetooth networking is widely used as an interfacing mean to connect peripheral devices like hands-free units to mobile phones. Each Bluetooth device has a unique electronic identifier known as a Media Access Control (MAC) address, and as a Bluetooth device passes a detection point both this address and the time of detection can be registered as a location sample. By determining the difference in detection time of certain observed MAC addresses, the travel time between sampling locations can be derived with a trivial calculation (Figure A1).

Basically, RDIF probing works similarly to Bluetooth detection. Longer-range active RFID tags used as electronic toll passes are uniquely identifiable transponders, which can be read at toll collection points and also at many desirable non-toll locations.

The third alternative – a cellular network based data collection method – has two sub-cases: network triangulation and hand-over location detection. The cellular network triangulation probe is less accurate than satellite navigation systems, requiring collection of numerous samples and the application of complex algorithms in order to extract high-quality data from cellular floating car (CFC) sample sets.

The hand-over location scheme is even more inaccurate. A handover refers to a mechanism by which an on-going call is transferred from one base station (BS) to another. By knowing the base station's location and its service area a mobile device's location can be roughly estimated. It should be noted, though, that the handover decision requires measuring several parameters like received signal strength (RSS), signal-to-interference ratio (SIR), and traffic load and mobile velocity – it is not only based on a simple distance between the mobile station and BS.

Even though these cellular based data collection methods are less precise, they are believed to have some real advantages over conventional sensor or satellite based methods, since no additional hardware or infrastructure needs to be built in vehicles or in the environment.

The basic FCD data can be enriched with additional information assembled from a vehicle's other sensors via an inbuilt communication network like CAN (Controller Area Network) or LIN (Local Interconnect Network). This extension data, commonly called extended FCD (xFDC), could include state of warning lights, wipers, outside temperature, torque, braking information, ABS/ESP and other diagnostic data that can be used to derive some parameters affecting traffic conditions.

FCD data collection methods are supposed to provide either supplementary information for better coverage of traffic data or to completely replace a fixed infrastructure that needs heavy investment and constant maintenance.

Several studies and practical FCD experiments have been performed internationally over the last 10 years. The results are promising but also slightly controversial; besides the technical problems associated with FCD technologies there are issues concerning privacy. The purpose of this work is to describe the FCD collection process and examine the technical challenges associated it, analyse the results of previous satellite (mostly GPS) based FCD systems and estimate data quality requirements for useful applications.

A3. Cellular network based FCD collection process

Tracking of mobility is one of the basic features of wireless cellular communication networks. Following of terminal movement is needed for location management and for handovers. With location management, the system tracks and locates a mobile terminal for a possible connection and the handover management maintains ongoing communications when the terminal roams from one cell to another.

Information available from 2G/3G network location and handover management can be utilized in several ways for user positioning. The simplest one is to use call data records (CDR) that the operators have collected for billing. The CDR format is not standardized but usually contains at least the cell and the time when a call was initiated and terminated. Some operators also maintain information about intermediate visited cells during the call (Janecek et al. 2012). From this information, terminal moving speed and traffic speed can be calculated pretty easily without costly additional investments for an infrastructure. Albeit the method is very coarse there is a good correlation of measurement compared to fixed loop detectors. This data collection method can be applied for example on highway segments with poor positioning accuracy.

Another non-triangulation based FCD data collection method is to acquire and analyse 2G/3G mobile network signalling information. Cellular networks contain Core Networks (CN) and Radio Access Networks (RAN). CN is divided into Circuit Switched (CS) and Packet Switched (PS) domains to support voice calls and data transmission respectively. Radio cells are the smallest spatial entities and those are grouped logically as Routing Ares (RA) and Location Areas (LA). A mobile terminal's change of LA and RA is always registered and monitoring these changes at CN/RAN level (even idle terminals) with suitable infrastructure positioning information for FCD purposes could be done. According to Janecek et al. (Janecek et al. 2012), two-level information (from active and also from idle terminals) originating from the mobile network for FCD purposes could be used to detect road incidents like congestion in a reliable and timely manner.

The third cellular based alternative for FCD collection is 2G/3G cellular network based triangulation. Time Difference of Arrival (TDOA) and Advanced Forward Link Triangulation (AFLT) are the most common methods of location determination. TDOA is the more complex method because the calculation is done inside the network.

Following the US E911 and European E-call initiatives, LTE networks (4G) now support a range of complementary positioning methods. E911 requires accuracy within 50–150 metres (95% reliability) when terminal based positioning is used, and 100–300 metres (90%) when network based positioning is used. LTE has several positioning alternatives to meet these requirements.

The basic method – Cell ID (CID) – exploits cellular system knowledge about the serving cell of a specific user. Enhanced Cell ID (E-CID) is another network-based method that utilizes CIDs, RF measurements from multiple cells, timing advance, and Angle of Arrival (AoA) measurements. The OTDOA (Observed Time Difference Of Arrival) method is based on reference signal time difference (RSTD) measurements conducted on downlink positioning reference signals received from multiple locations, where the user location is calculated by multi-lateration. OTDOA is expected to have an accuracy better than 100 metres.

Several other non-standard cellular positioning schemes could also be applied, like Radio Frequency (RF) fingerprinting, a method of finding a user position by mapping RF measurements obtained from the user equipment onto an RF map, where the map is typically based on detailed RF predictions or site surveying results. Another method is Adaptive Enhanced Cell Identity (AECID) that enhances the performance of RF fingerprinting by extending the number of radio properties used, where at least CIDs, timing advance, RSTD, and AoA may be used in addition to received signal strengths, and where the corresponding databases are automatically built up by collecting high-precision OTDOA and A-GNSS positions, tagged with measured radio properties (Ericsson 2011).

A4. GNSS based FCD collection process

The working principle of a satellite positioning system is trilateration, where a satellite transmits a signal containing its orbital data and a precise time (based on rubidium or caesium atom clocks), the receiver compares the time of the transmission with the time of reception measured by an internal clock, and calculates the distance to a satellite using the time difference. By using several (at least three, a fourth satellite is needed for the clock) satellite signals simultaneously, the receiver's position can be calculated quite accurately using the iterative Newton-Raphson method.

Currently, the US NAVSTAR Global Position System (usually better known as GPS) and the Russian Global'naya Navigatsionnaya Sputnikovaya Sistema (GloNASS) systems have worldwide coverage. The European Galileo will be operational in 2020 and the Chinese Beijou system is expected to expand from regional to a global use at the same time. Thanks to its wide coverage and first appearance on civilian markets, the GPS system has the most widespread use and commercial support. Availability of high sensitive integrated chipsets like CSR SIRF-IV enables cost effective GPS receivers to be embedded even into mobile phones. The technical challenges of GPS receivers are nowadays related mainly to mobile the GPS antennae of mobile devices (e.g. to eliminate interference with 3G and GSM radio frequencies). Also, the first multifunctional chips that enable simultaneous use of both GPS and GloNASS systems are now emerging (announced by Qualcomm, Broadcom, STM etc.) onto the markets and will improve the usability of GNSS in areas where receiving GPS signals is problematic (e.g. high latitudes north or south).

A satellite based FCD system contains a positioning sensor (a chipset, antenna and signal processing unit), a processing unit that converts the raw location data usable for road traffic monitoring, and communication unit that periodically sends data for further analysis and use.

The functional phases of the satellite FCD data collection process are:

- Sampling positioning data with timestamps and filtering out unwanted samples
- Combining the samples as trajectories
- Matching collected trajectories to the road network (map matching)
- The results of map matching can be exploited by traffic flow related services (estimated travel times, congestion detection etc.).

The first phase of the data collection process is to determine the validity of a location sample. Since often the origins of probe data are from commercial fleets like taxi or courier firms, it is important to recognize whether the vehicle has purposely stopped in order to wait for or drop off a customer, or deliver or collect a package, or whether it has stopped because of traffic congestion. Several statistical methods can be developed to filter out unwanted samples from a measurement set, or simple pruning heuristics can be applied based on other available sensor data (e.g. whether or not the taxi meter is on).

A GPS position sample itself is not flawless either. Several sources of positioning errors affect the measurement result, the most influential being (Olynik 2002):

- Earth's ionosphere distortion delays a signal depending on its frequency. By using multiple frequency bands (in the case of GPS, L1 and L2C frequency bands) of carrier waves, this distortion's effect can be calculated and partially corrected. Ionospheric correction data are also distributed in the USA using the Wide Area Augmentation System (WAAS), and in Europe with the European Geostationary Navigation Overlay Service (EGNOS). The estimated ionospheric error for the GPS signal is around 5 metres.

- The satellite's clocks and receiver clocks are not free of noise or clock drift errors. Although these errors are minimal, they affect accuracy by a couple of metres. Additionally, the satellite orbit has a slight instability, which affects accuracy accordingly.
- Multipath propagation, where radio signals reflect off surrounding terrain, causes interference (fading) and phase shifting of signals. The net effect of these on GPS accuracy is around a metre.

GPS accuracy is also dependant on a satellite constellation, i.e. the number of visible (line of sight) satellites and their geometric distribution in the sky related to an observation position. So-called urban canyons, where GPS signals might be blocked by tall buildings, reduce the accuracy and usability of GPS. These errors have an unknown distribution which makes them very difficult to model and correct. Additional signal sources like GloNASS satellites or dead reckoning (e.g. deducted reckoning, DR) sensors such as inertial navigation systems (INS) based on gyroscopes, odometers etc. can alleviate this urban canyon problem.

Since a GPS sample inherently contains inaccuracies, the FCD tracking data is collected as a set of samples into a database referred to as trajectories (Brakatsoulas et al. 2005). A trajectory is a path that a moving object follows through space as a function of time. The accuracy of the trajectory is also dependent on the sampling rate, i.e. how often a GPS position sample is acquired. To be useful, the trajectory data has to be aligned with the underlying road network by means of map matching algorithms. This map-matching task is probably computationally the most complex task related to FCD data processing.

A5. Map Matching

Map matching algorithms have been studied for nearly a hundred years (including the work of Maurice Frechet in the early 19th century), and new algorithms are still under development. Positioning accuracy has improved considerably due to new map matching algorithm approaches. Algorithms can be targeted for specific purposes or generally applicability, for real-time applications or for uses where post-processing is sufficient. Map-matching algorithms can also be characterized as either global or incremental. Global algorithms batch-process the entire input trajectory before generating a solution. Incremental algorithms use strategies that divide the input trajectory into smaller segments and process them sequentially. Most existing incremental (online) algorithms use localizing strategies such as a fixed sliding window and a fixed-depth recursive look-ahead (Quddus et al. 2007). The sliding window method simply divides the trajectory into fixed-sized input sequences and handles them independently. A larger window size leads to better accuracy but a longer output delay, and vice versa. Most ITS services require a map-matching algorithm that can be calculated in real-time, and most of the current map matching algorithms are developed for real time satellite navigation (SATNAV) purposes.

In general, map matching means establishing a correspondence relationship between two feature sets. The purpose of a map-matching algorithm in ITS application is to identify the correct road segment on which the vehicle is travelling and to determine the vehicle location on that segment. A good map-matching algorithm enables the physical location of the vehicle to be identified and also improves the positioning accuracy if high quality spatial road network data are available (Quddus et al. 2007). High quality map matching is essential e.g. for some critical ITS applications like road usage charging (RUC), where measuring the usage is considered to be satellite based (for example in Finland) (Grush 2008).

A digital map brings the FCD system another error component: A representation of the road network (finite network of roads) is always an approximate of true road paths. Studies by Brakatsoulas et al. (2005) and Quddus (2006, et al. 2007) indicate that the quality of spatial data has a pronounced impact on the performance of map-matching algorithms. Recently the modelling quality of map data has raised questions and caused user annoyance (Cooke 2012).

Usually straight road parts (segments) are represented as single lines and curvatures are represented as piecewise linear lines (for a gentle curve) or as polylines (for a sharp curve). This planar model, where roads are presented as arcs (road segments) and nodes (ends of straight lines, intersections) and shape points (nodes that describe curvatures), is a commonly accepted representation because of its high efficiency (ability to support finding minimum cost route to destination) and low complexity (see Figure A2). Essentially a road network is a (directed) graph of finite set of arcs and nodes (vertices). For map-matching purposes nodes have attributes that identify their spatial location in a coordinate system, for other (ITS) applications more attributes can be attached to arcs and nodes.

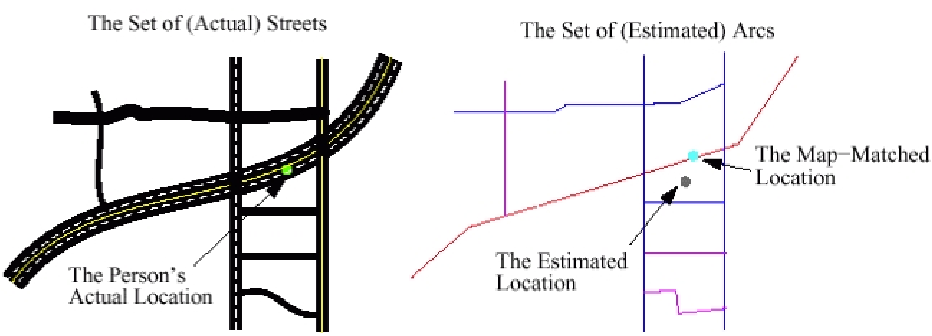

Figure A2. Map matching process with a modelled road network.

If the vehicle position is well estimated and the digital map is very accurate (i.e. precisely modelled), development of a map algorithm seems to be quite straightforward and simply snaps the positioning data to the nearest road segment. However, even the simplest snapping techniques may give incorrect results in urban areas where the density of roads is high. Also, many of the developed map-matching algorithms fail to meet the requirements of many ITS services. Quddus (2006) and Quddus et al. (2007) have indicated at least the following problems related to these:

- The algorithms are unreliable, especially at junctions (Y-junctions) and in the vicinity of parallel roads.
- The algorithms ignore the error sources associated with the navigation sensors and the digital maps.
- They are not suitable for urban road networks due to inaccurate results.
- There is no method to determine the accuracy offered by the algorithms.
- There is no discussion on the level of confidence (integrity) of the map-matched locations.
- No sensitivity analysis has been done to assess how the navigation sensors and the digital map quality affect the performance of map-matching algorithms.

Map matching is essentially a heuristic search problem: finding the best matching criteria for selecting the best candidate from the neighbouring segment (link) set. Several different approaches can be applied to solve this problem (Yang et al. 2010, Quddus et al. 2007).

A simple geometric analysis is based on the shape of the road network links (segments) only, and doesn't take into account how the links are interconnected. Earlier the most frequently used method for geometric map matching was a simple search where the position samples were matched to the closest nodes of the road segment. This "point to point" mapping is efficient to implement, but accuracy is sensitive to how the road data is acquired and modelled. In point-to-curve matching, the location sample is matched onto the closest curve in the network. The line segment that gives the smallest distance is considered the one on which the vehicle is apparently travelling. In curve-to-curve matching, the vehicle's trajectory is compared against known road segments. Unfortunately, when the traveling speed is low and the vehicle makes brief stops along the course, this approach will encounter problems like unpredictable GPS drift (Zhou & Colledge 2006). According to the studies of Quddus (2006) and Quddus et al. (2007) these methods in general are able to identify the correct link with a likelihood of 85–88%.

Topological map-matching algorithms use relations between mapping entities like connectivity of lines, adjacency and containment of polygons points. Using trajectories (e.g. historical measurement data), and combining these into possible road segment alternatives restricted by topology, increases the reliability compared to simple geometrical distance (or angle based) methods. Several algorithms have been developed (White 2000) that have the ability to identify a correct link with around 90% probability.

Probabilistic approaches use a confidence region around the location sample and superimpose it on the road network to identify where the vehicle is most likely travelling. Some of the probabilistic algorithms are very suitable for low speed travelling in urban areas with frequent stops. The correct link identification rate is around 98% and accuracy around 10 metres.

A set of even more advanced algorithm concepts using Kalman filtering, particle filtering, fuzzy logic, Bayesian inference and genetic algorithms have been developed over the last few years, the best of which are able to identify a correct segment of road with 99% probability (Work et al. 2008).

The practical applicability of these algorithm approaches also needs study of time-space complexities (denoted usually with a big O). These aspects related to performance are not often clearly expressed in research papers (and are also beyond the scope of this study), but they might affect the architecture of the FCD collection system. If a high accuracy and matching probability is needed for the intended ITS application and resource, hog algorithms ought to be applied, and calculations should be distributed. In this case map matching may more likely be performed in vehicle units instead of centralized data collection points (as done currently with popular cloud services). This thick vs. thin sampling client dilemma has another practical aspect from the viewpoint of the FCD system architecture, since data transmission is not free. Thin sampling at a high sampling rate means increased traffic compared to a more autonomous thick client. Also privacy concerns (not examined more closely here) have some effects on the selection of architecture.

Mapping accuracy of these algorithms is enough for real-world applications, and occasional errors caused by e.g. matching errors encountered with navigation systems can be resolved by users. Usually when matching algorithms are developed, it the location sampling rate is expected to be around 0.3 to 1Hz at least (e.g. a sample every 1–3 seconds) (Yuan et al. 2010). Incremental algorithms perform well and run fast when the sampling rate is high but accuracy degrades fast as sampling rates decrease. In practice, the sampling rate might be far less than 1–10 s, around minutes in the case of a fleet-based system where FCD collection is not the primary reason for vehicle positioning. Even the most global algorithms (for batch processing) have only 60% accuracy when the sampling rate exceeds 2 minutes (Yuan et al. 2010).

If the sampling rate of a vehicle's path is 30 s and the travelling speed is a modest 50 km/h the distance of the location samples is well over 400 metres. Pfoser & Jensen (1999) have shown that a sampling error of two points (P1, P2) based on maximum speed on the measurement interval (t1, t2) has a lens-shaped error probability distribution. When this probability error is calculated over trajectory (a set of continuous samples), the positioning error forms an ellipse. By using 25 km/h average speed and 50 km/h top speed this error ellipse has a major axis of about 420 metres and thickness 390 metres (Brakatsoulas et al. 2005). All road segments inside this ellipse should be considered as possible travelling segments. Of course, topology information can be used to reduce this space but plenty of alternatives still remain.

The low sampling/polling rate problem has generated interest in new algorithms especially suited to this type of FCD application domain. Several approaches to solve this problem have been suggested, like ST-Matching (Lou et al. 2009), Interactive Voting-based Map Matching (Yuan et al. 2010) or Weight-based Shortest Path (Zheng & Quddus 2011).

For example ST-Matching utilizes the spatial geometric and topological structure of the road network and temporal (speed) constraints of trajectories to create a best candidate graph of travelled road segments. ST-Matching is a global algorithm, i.e. utilizes an entire trajectory, but also a localized version using only a shorter sliding window of samples has been developed. Time complexity is logarithmic O(log(n)), and n is the count of road segments in the model. IVMM also exploits road networks topology but matching is based on weighted mutual influence of GPS sampling points. The accuracy of these algorithms is in the 60–70% probability range and may not be suitable for some real time ITS applications like estimating link travel time for traffic flow analysis (Zheng & Quddus 2011).

A6. Practical FCD collection experiments

The first ideas to utilize GPS as a source of FCD in practice were voiced shortly after the NAVSTAR system became fully operational (1994). Especially in Germany, the FCD collection principle gained great popularity and many early experiments were performed. Since then, FCD related projects have been launched all over the world including EU countries, the USA, Japan and China.

Table A1 summarises several reported experiments from FCD projects. Most of them used data gathered from commercial fleets. Collecting vehicle position data and processing it to produce comprehensive travel time information is often seen to benefit both private fleet companies and public transport authorities as a "win-win" business solution. Using taxis as a probe fleet seems to be particularly popular, since taxi dispatch systems in several cities are based on periodically sent location updates.

Figure A3. Freeway Service patrol truck used in USA FCD projects.

Feasibility studies on using several other probe vehicles have also been performed. In 2006, Freeway service patrol (FSP) trucks (Figure A3) were used in Washington State as FCD vehicles. The study showed that FSP truck speed was a poor measure and always underestimated (Moore et al. 2002).

A study in Japan (Liu et al. 2008) assessed whether bus probes provide good data about fluctuations in travel times due to their repeated travel along specified routes. Methods to detect halts at bus stops were developed and evaluated – however, separating the stopping delay times at bus stops from those at traffic signals remained an unresolved issue when a bus stop and a red traffic signal were in close proximity.

Table A1. Some FCD experiments.

Reference	**Country/city**	**Fleet/fleet size**	**Year/ remarks**
VERDI	Germany	850 vehicles	1997
DLR	Germany/Berlin, Austria/Vienna	300 taxis (Berlin) 400 taxis (Vienna)	2001–2002
ORNIKO	Germany/Nuremberg	500 taxis	2004–2008
CVIS, COOPERS, GST	EU-wide	?	2006–2010/ extended FCD
P-DRGS	Japan/Nagoya	1570 taxis	2002–2004
Mobile millennium	USA/San Francisco	UC Berkeley students and volunteers 100 rented vehicles /2000 users	2008/spatial sampling principle
Montreal FCD	Canada/Montreal	53 car-sharing vehicles	1999–2004, 2006–2008

Several commercial companies use FCD data collection principles. The most import providers are listed below.

Table A2. Some commercial traffic information providers using FCD.

Company	**Area/ user count**	**Fleet / fleet size**	**Remarks**
Tom-Tom	Europe /80 million users	5.5 million vehicles (navigator based data, like Tom-Tom HD, Tom-Tom Work etc.)	FCD data is sampled once a second and synchronized with the backend system once every 1–3 minutes
MediaMobile	France	100,000 vehicles in France 26 million mobile users in France for cellular	The FMD (floating mobile data) technology developed by Orange
Inrix	USA and 30 other countries	Nearly one million GPS-enabled probe vehicles	Over a billion usable GPS data samples per month are acquired, 500,000 new samples every 15 minutes
Navteq/ T-mobile	Germany	Over 50,000 vehicles with FCD technology in Germany	Navteq also utilizes floating cellular data provided by T-Mobile

A7. Future trends in traffic data collection

Traffic information based on social media and crowd sourcing are the latest trends for obtaining real-time traffic information from users. Social media and crowd sourcing rely on a large number of voluntary users and their benevolence in providing information for the common good. An example of these activities (supported by an appropriate service) is TomTom, which uses its customer base as an information source.

TomTom has used crowd sourcing in its development of Map Share, IQ Routes and Go Live services. Map Share allows drivers to make map corrections themselves to create more-up-to-date maps. IQ Routes’ anonymous GPS logging (with “Home”-uploading software) enables navigation routes to be calculated using actual average speed conditions. HD Traffic makes IQ Routes nearly real time by enabling data collection using the FCD principle. TomTom has more than 80 million customers and, according to the company, most users are willing to contribute to TomTom’s system development through the offered tools.

Another example is Waze, a social mobile application that offers free turn-by-turn navigation based on real-time traffic conditions. In July 2012, Waze announced that it had reached 20 million users worldwide. Compared to other GPS navigation schemes Waze is a community-driven application. Similarly to Tomtom’s IQRoute/HD Traffic, the system learns from users' driving times to provide routing and real-time traffic update. Waze also enables users to actively report on local conditions such as traffic jams, crashes, speed traps, blocked roads, and correct map information (Figure A4). Waze simultaneously sends anonymous information, including users' speed and location back to its database to improve the service as a whole. Waze is also refining maps using the same crowd-source principle as TomTom is doing with Map Share. The Waze application also incorporates gaming and other social elements, like earning points. Waze has received risk funding twice ($25M in 2010, $35M in 2011), and the company's business plan is to offer location-based advertising in the future.

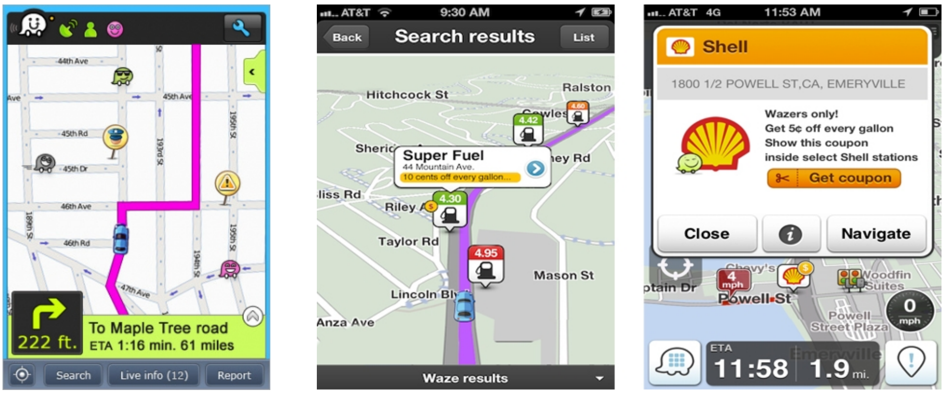

Figure A4. Waze, a social traffic application for mobile devices.

A8. Summary and requirements for real-time traffic situation information

Floating car based probe systems are considered to be the most cost-effective method of collecting data within a large area where other probe technologies are costly to implement and maintain. Commercial fleets with probe systems have some obvious advantages since the required infrastructure already exists (GPS and network connectivity), vehicles are on duty most of the day, and usage covers a wide part of the road network. Traffic surveillance systems based on FCD alone are still rare, but are supposed to be a viable alternative in developing countries where heavy investments and operational costs are not desirable.

The benefits of FCD can be summarized as follows:

- Coverage – capability to collect traffic information from a very wide area
- Cost effective – investments and maintenance costs are far less than with conventional fixed sensor technologies
- Very detailed information source – the only solution that can deliver very detailed origin, destination and route choice information
- Capable of supporting a vast array of applications.

Despite of the obvious advantages of FCD as a data collection method, there are also some restrictions that should be taken into account when considering its use. The main requirements related to the applicability of fleet based FCD are:

- The behaviour of a fleet should be representative of the total traffic
- The fleet size should be large enough to deliver useful information
- The sampling rate and measurement accuracy should be enough for the intended purposes
- Costs associated with data collection should be in balance to the information gained
- New ITS services should be created from “infrequent” FCD or extended FCD data.

As indicted by the experiments, taxi fleets have been a popular “test bed” for the FCD project, mostly due to the existing infrastructure for customer dispatching. The question of whether professional driver fleets represent the “normal” traffic flow is an essential issue and plenty of discussion about the topic can be found in research papers. For example, specialised fleets like taxis and buses can use specialized lanes. According to studies taxi drivers have a tendency to gather in high demand areas or at points where drivers can expect to find customers, thus the area coverage might not be as extensive as expected. Experienced and professional drivers will also avoid frequently congested areas, so some congested areas will be lacking data. Other professional fleets like logistics companies can also produce FCD data but the reporting period is often based on cluster transmissions like daily or weekly reporting, and is suitable only for post-analysis oriented applications.

Another issue concerning professional fleets as a main data source is spatial and time coverage of traffic data. The availability of active taxis for probing depends on the demand for taxis in the city, which varies with the time of day. The Stockholm study (Rahmani et al. 2010) indicated two daily taxi traffic peaks, one in the morning and the other in the evening of working days. The morning demand peak was sharper, while the afternoon peak was wider and not very high. Generally, taxi dispatch demand was less at weekends than on working days, except for Saturday evening. According to Japanese studies, the time distribution of taxi traffic is fairly equal to normal traffic flows.

FCD alone from special fleets like taxis and buses is not such a good source, since these fleets do not provide volumes, which are important for forecasts. Studies have indicated that in order to produce reliable travel time information, the penetration of floating cars should be around 2–5% of the total vehicle count. For detecting disturbances within minutes, far more probing vehicles are needed, probably in the range of 5–10% of the total vehicle count. To meet this requirement, probing vehicles and cellular data should be used in addition in order to provide enough data sources for the real time traffic snapshot.

As stated in the methodology part of this study, the sampling rate has a huge influence on the map matching reliability. When using only a fairly sparse sampling rate (minutes), even with the best known algorithms the map matching-accuracy is around 60%. With dense sampling (seconds), the correct matching probability is around 99%. Depending on the application, also the “freshness” (timing) of the data has obvious significance. For real-time traffic control and congestion estimation, it is generally thought that a sample older than 5 minutes should not be considered as a valid datum and it be dropped.

To date, the widespread applicability of FCD based services has been restricted for economic reasons – mostly due to high cost associated with wireless data transmission and prices of satellite positioning devices. The current tendency for flat-rate data transmission charging and the sharply falling prices of GPS chipset-equipped smartphones have levelled these obstacles. In the future, a bunch of Waze-type social media applications exploiting FCD collection principles for a wider audience are certain to be available. New application areas might be pedestrian and bicycle traffic data collection and related services.

A9. Data fusion architecture

It is unlikely that real-time traffic snapshots could be constituted solely based on FCD in the near future. Accurate GNSS based FCD needs participatory sensing, where the user actively engages in the data collection activity (i.e. the user manually determines how, when, what, and where to sample) by loading suitable software like Waze or Nokia’s HERE Drive+ into their mobile phones which periodically send collected data for further processing. Despite the increasing popularity of this kind of application, not enough coverage could be reached in the short run. As already discussed, even the use of existing fleets as a source of GNSS based traffic data has various restrictions. Opportunistic sensing, where the data collection stage is fully automated with no user involvement, is currently a more plausible alternative. Cellular FCD, fixed in-roadway and over-roadway sensors give better options for this kind of opportunistic traffic data collection.

To achieve the required reliability and coverage for traffic applications, a combination of fixed and xFCD data sources need to be applied. Multi-sensor data fusion can be defined as

> “the theory, techniques and tools which are used for combining sensor data, or data derived from sensory data, into a **common representational format**...in performing sensor fusion our aim is to improve the quality of the information, so that it is, in some sense, better than would be possible if the data sources were used individually” (Mitchell 2007).

Some hints related to this approach could be obtained from experiments in other countries like Taiwan. Chang et al. (Chang et al. 2012) have presented a cloud based data fusion architecture that utilizes multiple data sources like fixed probes, GPS probes and cellular FCD extracted from base station signalling to give extensive and accurate traffic information in near real-time. Data fusion from multiple sensor sources requires appropriate algorithms for further data handling like congestion detection etc. One possible method has been presented by Zhanquan et al. (2009). Road traffic data fusion methods are also studied at VTT as part of the EU-funded ASSET project (www.asset-project.com).

A10. References

Brakatsoulas, S., Pfoser, D., Salas, R. & Wenk, C. 2005. On Map-Matching Vehicle Tracking Data. Proceedings of the 31st VLDB Conference, Trondheim, Norway. Pp. 853–864.

Chang J., Chen S., Chia C. & Wang Y. 2012. The Methods of Multiple Vehicle Speed Detection and Data Fusion of the Road Network: A Case Study from Taiwan. Information Science and Digital Content Technology (ICIDT). Pp. 422–425.

Cooke, T. 2012. A letter from Time Cooke on Maps. Apple Computer Inc. http://www.apple.com/letter-from-tim-cook-on-maps/. Accessed 15.10.2013.

Ericsson 2011. Positioning With LTE. Ericsson white paper 284 23-3155. September 2011. 12 p.

Grush, B. 2008. The case against map-matching. European Journal of Navigation. Vol. 6. Number 3, November 2008, pp. 2–5.

Innamaa, S. 2012. Validation of blue-tooth based travel time data. VTT. (Internal memo). 4 p.

Innamaa, S. & Hätälä, E. 2012. Floating mobile data pilot in the Helsinki Metropolitan Area Validation of travel time data. VTT Technology 50. Espoo. 51 p. http://www.vtt.fi/inf/pdf/technology/2012/T50.pdf.

Janecek, A., Hummel, K. A., Valerio, D., Ricciato, F. & Hlavacs, H. 2012. Cellular data meet vehicular traffic theory: location area updates and cell transitions for travel time estimation. UbiComp '12 Proceedings of the 2012 ACM Conference on Ubiquitous Computing. Pp. 361–370.

Kosonen, I. & Pahlman, S. 2005. Ajantasaisen liikennetilamallin tuottaminen floating car -datan avulla. Esiselvitys. Aino-julkaisuja 12/2005. 57 p.

Leduc, G. 2008. Road Traffic Data: Collection Methods and Applications. Working Papers on Energy, Transport and Climate Change N.1.

Liu, K., Yamamoto, T. & Morikawa, T. 2008. Feasibilities and Challenges of Probe Technologies for Real-Time Traffic Data Collection. Plan, Build, and Manage Transportation Infrastructure in China. Pp. 328–340.

Lou, Y. Zhang, C., Zheng, Y., Xie, X., Wang, W. & Huang, Y. 2009. Map-Matching for Low-Sampling-Rate GPS Trajectories. Proceedings of the 17th ACM

SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '09). ACM, New York, NY, USA. Pp. 352–361.

Mimbela, L. & Klein, L.A. 2007. Summary of vehicle detection and surveillance technologies used in Intelligent Ttansporation Systems. The vehicle detector Clearinghouse, New Mexico State University, August 31, 2007. http://www.nmsu.edu/traffic/. Approx. 230 p.

Mitchell, H.B. 2007. Multi-Sensor Data Fusion, an Introduction. Springer, Heidelberg, Berlin. 277 p.

Moore, J., Cho, S. & Mezger, D.B. 2002. Feasibility of Using Freeway Service Patrol Trucks as Probe Vehicles. Journal of Transportation Engineering, Vol. 128, No. 6, November/December 2002. Pp. 528–536.

Olynik, M. 2002. Temporal Characteristics of GPS Error Sources and Their Impact on Relative Positioning. UCGE Reports Number 20162 Department of Geomatics Engineering. 140 p.

Pfoser, D. & Jensen, C. 1999. Capturing the uncertainty of moving-object representations. In Proceedings of the 6th International Symposium on Advances in Spatial Databases (SSD '99). Springer-Verlag, London, UK. Pp. 111–132.

Puckett, D. 2010. Bluetooth®-Based Travel Time/Speed Measuring Systems Development. Texas Transportation Institute Final Report, UTCM Project #09-00-17, June 2010. 53 p.

Quddus, M. 2006. High Integrity Map Matching Algorithms for Advance Transport Telematics Applications. Doctor of Philosophy Thesis, Imperial Collage London. 258 p.

Quddus, M.A., Ochieng, W.Y. & Noland, R.B. 2007. Current map-matching algorithms for transport applications: State-of-the art and future research directions. Transportation Research Part C 15. Pp. 312–328.

Rahmani, M., Koutsopoulos, H.N. & Ranganathan, A. 2010. Requirements and Potential of GPS-based Floating Car Data for Traffic Management: Stockholm Case Study. 2010 13th International IEEE Annual Conference on Intelligent Transportation Systems Madeira Island, Portugal, September 19–22, 2010. Pp. 730–735.

University of Maryland 2008. Center For Advanced Transportation Technology, 2008. Bluetooth traffic monitoring technology – concept of operation & deployment guidelines. September 17, 2008. 5 p.

White, C. 2000. Some map matching algorithms for personal navigation assistants. Transportation Research Part C: Emerging Technologies, Vol. 8, No. 1–6. (December 2000). Pp. 91–108.

Work, D., Tossavainen, O.-P., Blandin, S., Bayen, A.M., Iwuchukwu, T. & Tracton, K. 2008. An ensemble Kalman Filtering approach to highway traffic estimation using GPS enabled mobile devices. Proc. of the 47th IEEE Conference on Decision and Control, Cancun, Mexico, December 2008. Pp. 2141–2147.

Yang, S., Chon, K. & Kang, S. 2010. The map matching algorithm of GPS data with relatively long polling time intervals. Journal of the Eastern Asia Society for Transportation Studies, Vol. 6. Pp. 2561–2573.

Yuan, J., Zheng, Y., Zhang, C., Xie, X. & Sun, G. 2010. In Proceedings of the 2010 Eleventh International Conference on Mobile Data Management (MDM '10). IEEE Computer Society, Washington, DC, USA. Pp. 43–52.

Zhanquan, S., Mu, G., Wei, L., Jinqiao, F. & Jiaxing, H. 2009. Multisource traffic data fusion with entropy based method. 2009 International Conference on Artificial Intelligence and Computational Intelligence. Pp. 506–509.

Zheng, Y. & Quddus, M.A. 2011. Weight-based Shortest Path aided Map-Matching Algorithm for Low Frequency GPS data. TRB2011 annual meeting. 20 p.

Zhou, J. & Colledge, R. 2006. A Three-step General Map Matching Methods in GIS Environment: Travel/Transportation Study Perspective. University of California Transport Center, Faculty Paper: Fall 767.

Appendix B: Intelligent Parking Service

B1. Introduction

Passenger car traffic comprised 73.5% of all passenger kilometres performed in the EU-27 area in 2009. Adding 10.2% from other road transport modes gives the total road transport share of passenger traffic. In addition, road transport hogs the lion's share also in freight transport with 73.8% of all inland freight transport modes. (EU 2011.) For that reason, road traffic is one of the core issues in traffic services and management.

The dominance of passenger car traffic causes the biggest problems in cities (e.g. traffic congestion, reduced safety, harmful environmental and health impacts and decreased pleasantness of the city). Car traffic brings to mind moving cars and traffic flows through the traffic network. However, most of the time, cars are not moving and drivers are not driving – cars are parked for 95% of their lifespan (Ojala 2005, Litman 2013), often in several different places daily. Parking is, therefore, an essential part of the transportation system.

Parking can be defined as the act of stopping a vehicle and leaving it unoccupied for more than a brief time. Parking requires a parking space, and often (at least in urban areas) it is indicated by white-paint-on-tar and/or designated signs. A parking fee, in turn, is used to collect money in exchange for the right to park a vehicle in a particular place for a limited amount of time.

Parking can be classified according to:

- the destination of travel: daily business, work related, residential parking;
- the duration of parking: short, medium and long term parking;
- the responsible constructor: private, public parking;
- the parking place type: surface, underground parking;
- the location of parking: on-street parking, parking lots. (Koukkula 2011).

Parking is one of the main traffic management concerns for several reasons:

- Parking places form part of the traffic network and thus an important part of urban and traffic planning, added to land use and costly investments dedicated to traffic.
- People drive to parking places and away from them (between parking places), thus impacting the utilization rate of different parts of the road network and traffic safety.
- Parking availability impacts traffic flows and the use of alternative mobility modes.
- Non-optimal parking solutions cause extra traffic, more congestion, more environmental harm, less safety, and more human stress in traffic.

In urban areas these concerns culminate as a mass of people cramping themselves into an area with limited space and resources for traffic. For example, in Helsinki city centre surface parking space is very limited, which has led to very expensive underground parking solutions. The need for parking space in cities also has an impact on other functions in the cities. E.g. expensive parking space forces apartment builders to compromise in the quality and features of the apartments themselves, as the lion's share of costs goes to supplying the required parking space. More residential parking licences have been issued than there are spaces available, and parking places for guests are often non-existent.

The continuous increase of passenger car traffic has caused ever-increasing demand for parking places. Historically, this demand has been satisfied by offering more parking places. However, in the recent years parking management that aims at optimizing parking resource use has starting to gain ground. (Kodransky & Hermann 2011, Litman 2013) In addition, parking policies have been seen in the light of overall traffic management in the cities aiming at reducing people's automobile dependency (see Figure B1). In this, technological advancements provide efficient solutions with enhanced quality of service, e.g. in the form of:

Average costs of parking (for one passenger car):

- Street parking: €1–3K
- Garage parking: €10–15K
- Garage parking, underground: €15–30K

(Multamäki & Taskinen 2008)

- electronic parking guidance helping drivers to find parking places more efficiently,
- smart meters registering parking vehicles,
- pay-by-phone payment, and
- automatic number plate recognition for enforcement. (Kodransky & Hermann 2011).

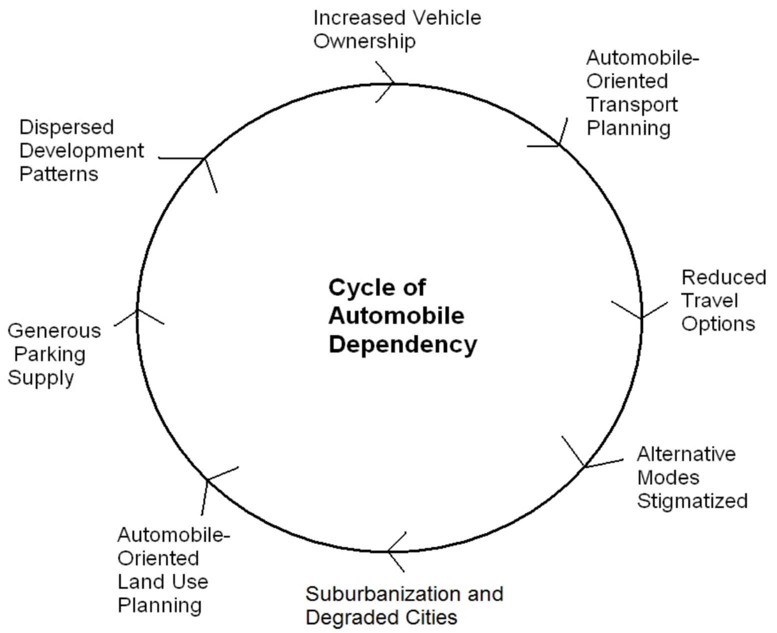

Figure B1. Cycle of Automobile Dependency (Litman 2012).

Despite policy changes and technological advancements, only some parking-related activities are (partly) supported by advanced services. For example, in Helsinki and many other cities, parking can be paid for with mobile payments, but 60% of payments are still made with coins (Helsingin kaupunki 2013). Some core activities related to parking are supported poorly or not at all. For example, in Helsinki, none of the service providers offer information covering all parking areas, let alone the availability of parking spaces.

Parking-related activities that are supported come as separate services altogether (e.g. navigation, payment, public transportation route planning), and the parking process as a whole is not supported. This probably stems from the fact that different parking-related activities belong to different stakeholders, which for several reasons do not co-operate well enough to create aggregated services covering the parking needs of drivers. To remedy this, one proposed action is to open parking-related data to service providers (Helsingin kaupunki 2013).

While parking is not well managed from the end-user point of view, it is not much better from the parking management perspective. No one has a clear view of the occupancy of parking areas as a whole, and thus from the city perspective it is difficult to manage parking well. The city is a mess of on-street parking, commercial parking areas and garages, plus parking provided by dwellings, offices, shops etc. Most of these parking areas are not used efficiently, at the same time that there is a chronic lack of parking spaces. For example, **Table B1** below lists the utilization rates of parking space at different times in different places.

Table B1. Utilization rates of parking space according to city norms at different times by urban function (Malmö stad 2010).

Utilization rate (%)				
	Working day 10–16	Friday 16–19	Saturday 10–13	Night
Dwellings				
Dwellers	55–75	55	50	80–90
Guests	30	70	40	50
Offices	60–80	20	10	10
Shops	40	80–90	100	–
Industry	60–80	10	5	10
Schools	90	10	5	–
Hotels	50	50	30	80
Restaurants	75	40	60	–

Shared parking aims at enhance the utilization rate of the parking places by using the parking areas for different user groups in different times. Different pricing schemes including dynamic pricing has also been used with success to optimize parking place usage. See, for example, Helsingin kaupunki (2013) for other actions for optimization of the parking. Many of these actions can utilize ITS based systems as invaluable enablers producing benefits for all stakeholders.

In the following we take a short look on the different stakeholders and then describe the main parking related activities. After that we take an end user look on the parking process as a whole that includes the parking related activities defined earlier as constituent sub-processes.

B2. Parking as a use case

Intelligent Parking Service (IPS) supports its users throughout the parking process in all its tasks. This kind of service is not existing yet anywhere in the world but is an example of the service whole that could be realized within the ideal realization of the Multi-Service Model. This imaginary future service is described as a system enabling identified use cases and stakeholders interacting with the system as shown in Figure B2. The system boundaries enclose the whole intelligent parking system, in the scope of which the use cases (representing parking activities) are performed without making any assumptions as to how the system is actually realized. The stakeholders outside the system are external entities that interact with the system and are thus involved in the use cases. Next we briefly describe the stakeholders and use cases.

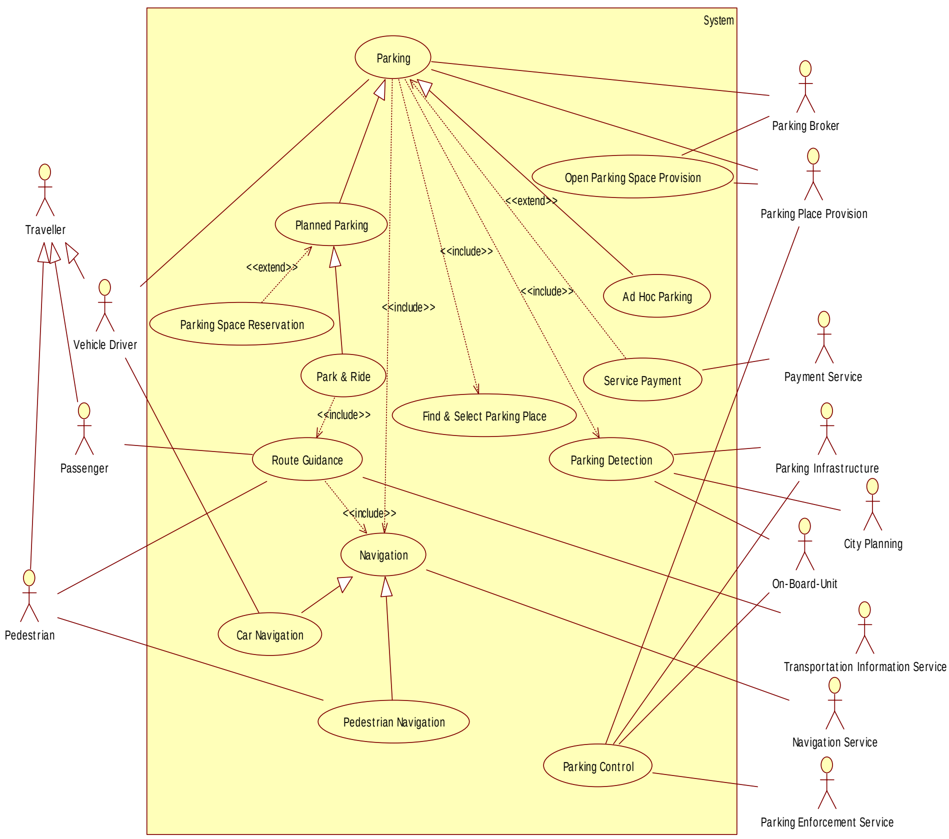

Figure B2. Use case diagram for parking activities.

B.2.1 Actors

Traveller *represents the person who uses their car and other transportation means to reach their destination. The traveller has the role of* ***Vehicle Driver*** *when driving a car or other vehicle needing a parking space. The role of vehicle driver changes to* ***Pedestrian*** *when the driver disembarks the vehicle to reach the final destination on foot, or to* ***Passenger*** *when using e.g. public transport for the same purpose.*

In the service provision we have identified several stakeholders that are involved in the parking process. **Parking Service** represents the stakeholder that actually provides the physical parking place, information related to it, and in many cases access to it (e.g. closed parking garages).

Parking Infrastructure is closely related to parking service, and consists of the system that detects parking, identifies vehicles and/or users, controls access, and provides information for parking enforcement etc. in the parking facilities. In this use case example, this stakeholder is deliberately distinguished from the parking service as its own entity, as in normal situations it often works automatically and autonomously without real-time control of the parking service.

Parking Broker, in turn, is a stakeholder that collects the (static and dynamic) parking place information from several parking services and conveys that information to the travellers. It may also handle e.g. payment, payment clearing and reservations on behalf of the parking services.

Payment Service in this use case refers to payment activities performed by the traveller. Possible back-office functionality, clearing and other financial functions are excluded here.

City Planning is included here as a background stakeholder, as it receives important information for its parking management plans and policies from the IPS.

On-Board Unit refers to devices in the vehicle that are the relevant enabler for automatic and intelligent parking activities, e.g. in the detection of parking or identification of the vehicle. In-car positioning and OBD II providing output for the IPS are examples of such on-board units in this example.

Transportation Information Service is responsible for providing information for public transportation (e.g. timetables, real-time information) in the Park & Ride case in this example.

Navigation Service provides positioning, map and routing services. Optionally it may provide a rich set of other services (e.g. route optimization based on real-time traffic information). In the multi-modal travel chain (Park & Ride) navigation is closely related to public transportation and route guidance.

Parking Enforcement Service uses the system to react when parking faults occur.

B.2.2 Parking activities

Parking in this example covers all the activities required, starting from when the traveller realizes their need for parking, and ending where the traveller has (paid and) left the parking place (or area). Parking can be planned in advance (**Planned Parking**) when the traveller plans the trip in advance, or the need for parking may arise while on the way (**Ad Hoc Parking**). In both cases, the traveller needs to find the parking place that suits their needs. This means finding and selecting an available parking space (**Find & Select Parking Place**). This is usually selected suitably near the planned destination, or along a public transportation route taking the traveller to the destination (**Park & Ride**). In ad hoc parking the usual case is to find a parking space as close as possible to the current location.

In the **Park & Ride** sub-use case, the traveller must continue from the parking place to the destination on public transport and on foot. Thus complete parking assistance should include **Route Guidance** supports the traveller in selecting optimal public transportation (and possibly other transportation) to the destination (and back to the parking place).

In planned parking, the traveller may also reserve a parking space in advance (**Parking Space Reservation**).

Navigation is one of the core activities of the parking process. It is needed to guide the driver to the parking place (**Car Navigation**) and the pedestrian to the final destination from the parking place (**Pedestrian Navigation**). Navigation is also needed as part of route guidance (e.g. during transit). Navigation may optionally include a rich set of services such as route optimization based on real-time traffic information. A detailed description of navigational aids is beyond the scope of this study.

Service Payment includes automatic payment activities that in this example are closely related to **Parking Detection**, where the vehicle or user is automatically detected in one of several possible ways. It should be noted that parking detection is not only used for payment, but also for real-time counting of available parking space and therefore providing the required information to drivers searching for parking place.

In Figure B2, we also included two separate cases that are not at the centre of the end-user parking process but are closely related to it. **Open Parking Place Provision** describes a service that allows anyone to lease out a free space (that they own) as a parking place, which increases parking space and optimizes resource use while creating new business. **Parking Enforcement Service**, in turn, is still needed to prevent errors. However, new intelligent technologies may substantially optimize this activity.

B.2.3 Future parking service – a use case definition

In the following we define a use case where the end user (Traveller) uses future Intelligent Parking Service to support all the parking related activities as a single service (instead of multiple separate services that is the current situation).

Use Case ID	Park		
Use case Name	Parking without advance reservation		
Created by	OJP	Last updated by	
Date created	24.10.2011	Date last updated	

Stakeholders	Traveller (as vehicle driver, passenger, pedestrian) IPS Parking broker Parking operator Parking infrastructure Navigation service, Route guidance service, Payment service, On-board unit

Description	This use case defines an example of parking activities supported by future IPS that aims at combining all the services needed in the parking process. The vehicle driver selects the parking place using IPS, gets guidance from the service, parks, is guided to the final destination and back, and pays for the parking before leaving the parking place.
Trigger	The vehicle driver needs a parking place for the car near the known destination or along public transportation routes leading to the destination.
Preconditions	The vehicle driver has an agreement with the IPS provider for the usage of the service, and has a terminal device connected to a network and the required application(s) to interact with the IPS. The vehicle driver has allowed storing of login details on the terminal device for automatic login to IPS. The terminal device is either a mobile device used throughout the process, or a combination of two devices allowing fluid transfer of the service between them where needed.
Postconditions	The vehicle driver has succeeded in parking, reached the final destination in time, confirmed payment for the parking, and left the parking place.
Normal flow	1. The traveller (as vehicle driver) launches IPS on the terminal device. 2. IPS opens the default view to the traveller (vehicle driver) 3. The traveller (vehicle driver) uses IPS to select the destination (e.g. by entering the address, selection from the map display, or speech commands) 4. IPS lists suitable parking areas (with free parking places) in order of distance from the final destination, providing additional information where requested. 5. The traveller (vehicle driver) selects one of the parking areas on the list. 6. IPS switches to navigation mode, guiding the traveller (vehicle driver) to the parking place. 7. The traveller (vehicle driver) follows the guidance given by the navigation service, enters the parking area, finds the parking space, and stops the car. (→ **Use Case: Car Navigation**) 8. The navigation service detects that the destination has been reached and returns to IPS (with “destination reached” information). 9. IPS is activated and asks the traveller (vehicle driver) to confirm the start of parking (again giving basic details such as price etc.). 10. The traveller (vehicle driver) confirms the parking, takes the terminal device and exits the vehicle. 11. IPS registers the start of the parking transaction and starts the navigation service (pedestrian mode with information about the final destination). 12. The navigation service starts guiding the traveller (now in the role of pedestrian) to the final destination along a pedestrian route. (→ **Use Case: Pedestrian Navigation**) 13. The traveller (pedestrian) walks to the final destination following the navigation instructions. 14. The navigation service informs the traveller (pedestrian) that the final destination has been reached, and returns to IPS. 15. IPS stays on, awaiting further action.

	16. The traveller (pedestrian) leaves the final destination for the car and activates the navigation service (pedestrian navigation mode, entering the location of the car as the destination) from the IPS view to get guidance back to the parking area. 17. The navigation service activates and starts guiding the traveller (pedestrian) back to the car. (→ **Use Case: Pedestrian Navigation**) 18. The traveller (pedestrian) walks back to the car along the route provided by the navigation service. 19. The navigation service informs the traveller (pedestrian) that the car has been reached, returns to IPS (with “car location reached” information), and stops. 20. IPS activates and (automatically) asks the traveller (now back in the role of vehicle driver) to confirm the end of parking. 21. The traveller (vehicle driver) confirms the end of parking. 22. IPS informs the traveller (vehicle driver) of the parking costs and asks for confirmation of the costs and payment. 23. The traveller (vehicle driver) confirms the payment. 24. IPS informs the user that the parking has successfully been paid for, stores the receipt of the parking transaction for the traveller (vehicle driver), gives instructions on leaving the parking area, and asks the traveller (vehicle driver) whether they wish to stop or continue using the service. 25. The traveller (vehicle driver) stops using IPS and exits the parking area.
Alternative flows	<u>Park.01 (Step 1): Initiation from the navigation service</u> The traveller (vehicle driver) starts IPS from the navigator service after setting the destination. 20. The navigation service asks the traveller (vehicle driver) whether they wish to find a parking place near the selected destination. 21. The driver wishes to find a parking place and chooses to start IPS. 22. The navigation service switches to IPS (with information about the selected destination). 23. IPS starts and continues from Step 4 in the normal flow. <u>Park.02 (Step 5): Automatic selection of the parking place</u> The traveller (vehicle driver) lets IPS automatically choose an optimal parking space (according to the user preferences stored in the system). 24. IPS informs the traveller (vehicle driver) about the selected parking area and continues from Step 7 in the normal flow. <u>Park.03 (Step 5): Alternative ordering of parking places</u> A list of available parking places is ordered according to the traveller’s (vehicle driver’s) preferences some other way. <u>Park.04 (Step 7): Entering an access-controlled parking lot</u> The selected parking place has access control (e.g. barrier gates). 25. The traveller (vehicle driver) follows the guidance given by the navigation service and arrives at the entrance to the parking area. 26. The navigator service detects that the destination has been reached and returns to IPS (with “destination reached” information).

27. IPS is activated and asks the traveller (vehicle driver) to confirm the start of parking (showing again the basic parking details, i.e. price etc.) in order to provide access to the parking area.
28. The traveller (vehicle driver) interacts with the parking infrastructure using IPS, or with a separate UI provided by the parking infrastructure itself, in order to confirm the start of parking and gain access to the parking area. The infrastructure can also be used to automatically identify the vehicle or driver in order to provide (or deny) access.
29. The parking infrastructure provides access to the parking area, e.g. by opening the gate at the entrance.
30. The traveller (vehicle driver) enters the parking area and finds a free parking space (possibly following the guidance given by the parking infrastructure). In addition, the traveller may store the parking place identifier in the IPS (e.g. via RFID tag) in order to locate the car later in large parking facilities.
31. Continue from Step 11 in the normal flow.

<u>Park.05 (Step 8): Change of parking plan on the fly</u>

The traveller (vehicle driver) decides to park somewhere else than in the selected parking area.

32. The traveller (vehicle driver) stops the car in another parking area than the pre-selected one and turns off the engine.
33. The on-board unit (with OBD II connection) informs IPS of the engine shutdown (indicating possible parking).
34. IPS informs the traveller (vehicle driver) that the parking space is not the one selected earlier by the traveller (vehicle driver) and asks whether the traveller (vehicle driver) wishes to discard the earlier parking plan.
35. The traveller (vehicle driver) confirms the change of plan.
36. IPS asks the traveller (vehicle driver) to confirm the start of parking in a new place (showing basic parking details, i.e. price etc.).
37. Continue from Step 11 in the formal flow.

<u>Park.06 (Step 12): Park & ride – continue to the final destination</u>

The parking area is far away from the final destination and public transportation is needed to reach it.

38. IPS asks the traveller (now in pedestrian mode after parking) whether he/she wishes to use public transportation instead of walking.
39. The traveller (pedestrian) confirms the wish to use public transportation.
40. IPS starts the route guidance service (with information about the current and final destinations)
41. The route guidance service starts guiding the traveller (now in pedestrian and passenger modes) towards the final destination on foot and by public transport **(→ Use Case: Route guidance in public transportation**)
42. The traveller (pedestrian/passenger) reaches the final destination following the instructions and using public transportation **(→ Use case: Public Transportation Payment**).
43. The route guide service detects that the traveller (pedestrian/ passenger) has reached the final destination and informs the user.
44. The traveller (pedestrian/passenger) acknowledges the information.

	45. The route guidance service stops and returns to IPS. 46. Continue from Step 15 in the normal flow. Park.07 (Step 15): parking time support The traveller requests e.g. information about spent parking time and costs **(→ Use case: Support during parking time)**. Park.08 (Step 16): Park & ride – return to the parking place The traveller returns to the parking place using public transportation and guidance offered by the route guidance application. 47. The user leaves the final destination and chooses to use public transportation to get back to the parking place. 48. Follow alternative flow similar to **Park.06** to get from the final destination back to the parking place and continue from Step 20 in the normal flow. Park.09 (Step 24): Exiting an access-controlled parking area The traveller (vehicle driver) exits an access-controlled parking facility. 24. IPS informs the traveller (vehicle driver) that the parking has been successfully paid for, creates an electronic exit ticket and gives instructions for leaving the parking area. 25. The traveller (vehicle driver) drives to the exit of the parking area. 26. IPS provides the electronic exit ticket to the parking infrastructure to allow exit from the parking area. 27. The parking Infrastructure provides access out from the parking area. 28. IPS asks the traveller (vehicle driver) whether they wish to stop or continue using the service. 29. Continue from Step 25 in the normal flow. Park.10 (Step 23): Using an external payment operator Payment takes place using a separate real-time payment process. 30. The traveller (vehicle driver) confirms the payment and IPS invokes the payment service (with information on the payment details). 31. The payment service launches, with payment options and transaction details. 32. The traveller (vehicle driver) executes the payment process confirming the payment **(→ Use Case: Service Payment)** 33. The payment service stops and control returns to IPS (with details of successful payment). 34. Continue from Step 24 in the normal flow.
Exceptions	The network connection is interrupted during the use case. • IPS cannot communicate with the back office The traveller's terminal or IPS is closed during the use case. • Closed intentionally • Closed unintentionally (e.g. because of a problem with the vehicle's electrical system or accidental shutdown of the terminal) IPS or traveller's terminal stops functioning during the use case. • Software or hardware fault GNSS-based positioning stops working during the use case.

	• GPS in shade • GPS subjected to jammer interference Some or more of the involved services/functionality do/does not work during the use case. • The use case involves several services, applications or basic functionalities from different service providers. If some of the services (and supporting applications and features) cease to work there must be a workaround. • Consider e.g. the payment process in an access-controlled parking lot.
Includes	Car navigation Pedestrian navigation Route guidance in public transportation Service payment Public transportation payment, Support during parking time
Priority	High
Frequency of use	0–n times/day/user; greatly dependent on the traveller; (n < 5)
Special requirements	• Privacy concerns must be addressed and the user must be able to review and control (when applicable) the information collected from his/her activities and understand who collects the data and for which purposes. • Misuse and fraud concerns must be addressed and detection possible. • Positioning is vulnerable if dependent only on GNSS. Alternative positioning methods must support GNSS-based positioning. • Services involving functionality, applications or functionality from several sources are vulnerable since the failure of one can prevent the use case from working. Thus alternative ways to run the use case during problems are essential. • User actions, transactions, working of the system, money traffic etc. must be verifiable afterwards in a trustworthy manner in order to cope with abnormal situations. Procedures for coping with these situations must be established.
Assumptions	In this use case the traveller is in the vehicle (in the role of vehicle driver) and interacting with the mobile or in-car terminal when starting the use of IPS.
Notes and issues	

B.2.4 Future parking service – involvement of different actors

In the following activity diagrams we sketch the normal flow (red lines) and most of the alternative flows of the use case definition presented in B.2.3. In the diagrams we have also looked deeper into the service provision, and expanded the end-user point of view to cover the main interaction patterns between stakeholders involved behind the scenes. In the real world, the actual patterns may vary and the example (presented in two parts) below shows the complexity of the service needed to support the whole parking process.

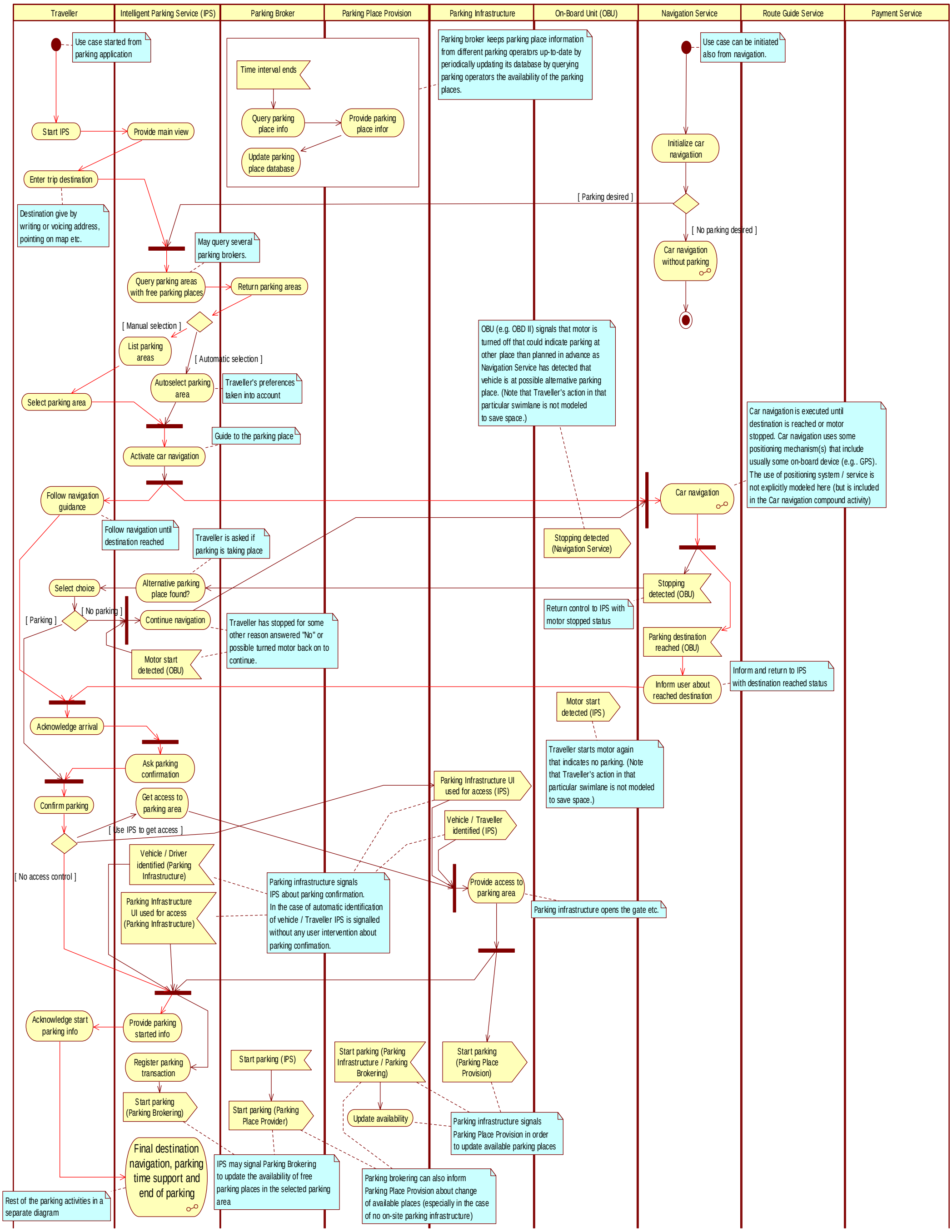

Figure B3. Activity diagram for the Intelligent Parking Service (part 1).

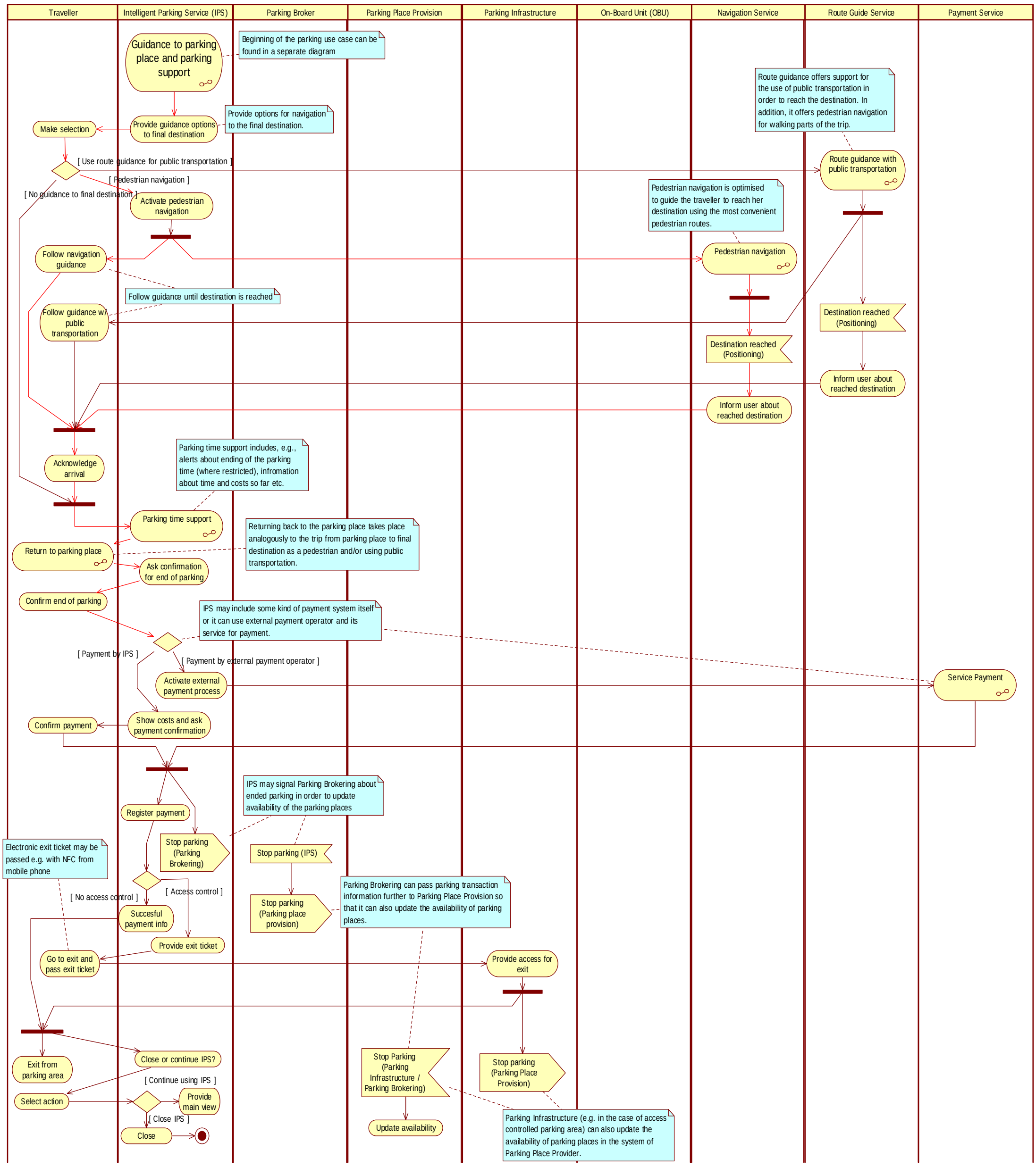

Figure B4. Activity diagram for the Intelligent Parking Service (part 2).

B3. References

EU 2011. EU transport in figures: Statistical pocketbook 2011. European Union, 2011. http://ec.europa.eu/transport/facts-fundings/statistics/doc/2011/pocketbook2011.pdf. Accessed 14.1.2013.

Helsingin kaupunki 2013. Helsingin pysäköintipolitiikka, Luonnos 18.3.2013, Helsingin kaupunkisuunnitteluviraston liikennesuunnitteluosaston selvityksiä 2013:1. http://www.hel.fi/static/public/hela/Kaupunkisuunnittelulautakunta/Suomi/Esitys/2013/Ksv_2013-03-26_Kslk_9_El/94A9129F-D99D-4637-B7BD-7C3D9672EC83/Liite.pdf. Accessed 15.5.2013.

Kodransky, M. & Hermann, G. 2011. Europe's Parking U-Turn: From Accomodation to Regulation, Institute for Transportation & Development Policy (ITDP). http://www.itdp.org/documents/European_Parking_U-Turn.pdf. Accessed 22.4.2013.

Koukkula, M. 2011. Pysäköinti ja kestävä kehitys, diplomityö, Insinööritieteiden korkeakoulu, Aalto-yliopisto. http://lib.tkk.fi/Dipl/2011/urn100531.pdf. Accessed 15.10.2013.

Litman, T. 2013. Parking Management: Comprehensive Implementation Guide, 29 August 2013, Victoria Transport Policy Institute. http://www.vtpi.org/park_man_comp.pdf. Accessed 15.10.2013.

Malmö stad 2010. Parkeringspolicy och Parkeringsnorm för bil, mc och cykel I Malmö, Pr 3087, Malmö stadsbyggnadskontor, Sebtember 2010. http://www.malmo.se/download/18.4027ea8b12af75326fc80003800/Parkeringspolicy+och+parkeringsnorm+slutligt+f%C3%B6rslag+antagen+av+KF.pdf. Accessed 15.10.2013.

Multamäki, M. & Taskinen, J. 2008. Pysäköintipolitiikka ja pysäköinnin hinta Helsingissä, Turussa ja Tampereella, Liikenne- ja viestintäministeriön julkaisuja 47/2007, korj. versio, Liikenne- ja viestintäministeriö, 2008. http://www.lvm.fi/fileserver/LVM47_2007.pdf.

Ojala, K. (ed.) 2005. Liikenne ja väylät I, RIL 165-1. Suomen Rakennusinsinöörien Liitto RIL ry., Helsinki.

Appendix C: Minimum data set for floating car data

Car telematics data markets have strongly evolved over the last decade. A set of professional and globally operating private sector companies have emerged to collect road traffic data by utilizing both publicly maintained fixed infrastructure sensors and moving vehicle probes known as floating cars. Floating car data (FCD) and its extended version (xFCD) is becoming a viable alternative for traffic data provisioning since road users are increasingly utilizing smartphones with sensing abilities and vehicles are equipped with on-board diagnostic (OBD) units capable of providing various real-time sensor data.

The problem with collecting and exploiting sensor data is that interfaces to FCD-capable devices, and the data formats related to them, tend to be more or less proprietary. Instead of costly and low volume proprietary FCD applications, open standards and supporting frameworks with well-defined interfaces are preferable for creating higher volume FCD data markets.

FCD or xFCD provides only a small but very substantial portion of the whole traffic information data needed for advanced Intelligent Traffic System (ITS) services. FCD or xFCD is not an own (or dedicated) service in any architecture – instead, xFCD can be seen more as a service enabler for several ITS services by providing means for collecting data as a basis for further analyses. Services that can be supported by FCD include:

- Road network performance analysis
- Monitoring of the complete network
- Before/after analysis, bottleneck analysis
- Route choice in case of events/detours
- Forecasting/estimation of traffic growth
- Origin-destination (OD) relations & matrices
- Route choice analysis for traffic simulation
- Emission modelling (air & noise pollution)
- Road maintenance and safety analysis
- Fixed sensor maintenance analysis
- Extended FCD provides even more alternatives for services like:
- Weather information
- Traffic incident data (collisions, accidents etc.)
- Road condition information for maintenance
- Vehicle-related applications (distance-related insurance & leasing, car finder & stolen vehicle etc.).

A more complete list of possible xFCD applications can be found elsewhere (Kummer et al. 2012).

The purpose of this study is to describe relationships between ITS and FCD (including xFCD), requirements for xFCD as part of the ITS infrastructure, standardization efforts related to FCD, and requirements for enabling xFCD information dissemination in the Finnish National ITS architecture model. A minimum FCD dataset is defined to enable Finnish companies to provide interoperable FCD datasets that can be used as bases to generate more advanced traffic services.

C1. Finnish TelemArk and domain specific traffic architectures

An ITS architecture is a set of high-level viewpoints that enable integration of ITS applications and services. The European FRAME Architecture was created to provide a minimum stable framework necessary for the deployment of integrated and inter-operable ITS and services within the European Union. The FRAME architecture comprises top-level requirements and functionality (via use cases) for almost all ITS applications.

Using architecture frameworks is usually considered a good practice, since they provide a means to organise thought. Frameworks help to ensure that fundamental questions are raised and weaknesses are analysed beforehand to provide better and more relevant information as a bases for decision-making. Unfortunately these benefits are not free. The selected framework has to be learnt before it can be effectively used. A framework might also be oversized for a target problem, so some feasibility study of framework applicability should be done before rushing to utilize it.

The European Union's FRAME architecture framework tends to be gigantic and tries to be all-inclusive. Consequently, the FRAME framework is hard to understand and master completely and is usually difficult to implement, thus plenty of resources should be available before it can be efficiently utilized. In fact the FRAME documentation explicitly mentions that it is "unlikely to be used in its entirety," so plenty of adaptation is needed.

The Finnish national ITS project TelemArk (1998–2000) produced two high-level architectures: conceptual and logical. The conceptual architecture includes a list of existing systems, stakeholders, process descriptions including process components and data flows, and a refined list of architectural demands. The processes identified in the TelemArk architecture (Mäkinen et al. 2000) are comparable with other ITS architecture and partially compatible with the European Union's Karen/FRAME architecture, although the modelling perspective is slightly different (i.e. more commercially oriented). The logical level architecture describes a structure that fulfils all the requirements defined and modelled in the conceptual architecture, including functions and information content (datasets).

At the conceptual level, TelemArk identifies processes that are classified based on the division given by the ISO project *Steering of the Development of Transport Telematics*. They are:

- Traffic information services: alternative transportation, traffic flow, incident and road works, weather, parking and public transportation information
- Customer demand management: park & ride, demand-responsive public transportation, payment systems etc.

- Traffic management: signal control, local warnings etc.
- Vehicle and transport management: hazardous goods, road maintenance, incident management etc.
- Driver assistance systems: dynamic speed control, route guidance, emergency services
- Enforcement systems: speed limits, transport weight monitoring, lane enforcement etc.

Considering this conceptual level classification, FCD is mostly related to traffic information services by providing means of collecting traffic flow information as well as weather and incident data via extended FCD. For traffic management purposes, xFCD is also a valid source of information when properly interpreted. As stated above, FCD can be used also as a tool for enforcement (e.g. speed limit control), but this application area is so controversial and has so many unresolved privacy and legal issues that it is not dealt with any further in this study.

The TelemArk architecture is very high level and not intended to be comprehensive; thus more detailed domain-specific architectures have been defined in separate projects. Two of these are LihArk (Tiehallinto 2005) and TosiArkki (Siponen et al. 2005), of which the former is related to traffic management issues and the latter is more oriented to the provision of real-time transport information.

The objectives of LihArk (Traffic Management Architecture) were to describe the present and future state of traffic management in Finland, and to identify the development needs for achieving the target traffic management status in 2007. The system architecture presents the functionality and connections between various traffic management sub-systems like traffic monitoring, reception and registration of incident data, variable message signs and data delivery. The traffic management architecture is divided into seven sub-system elements as follows:

- Traffic flow monitoring
- Weather and road condition monitoring
- Provision of traffic disturbances
- Traffic information management
- Traffic announcement
- Traffic information distribution
- Traffic management.

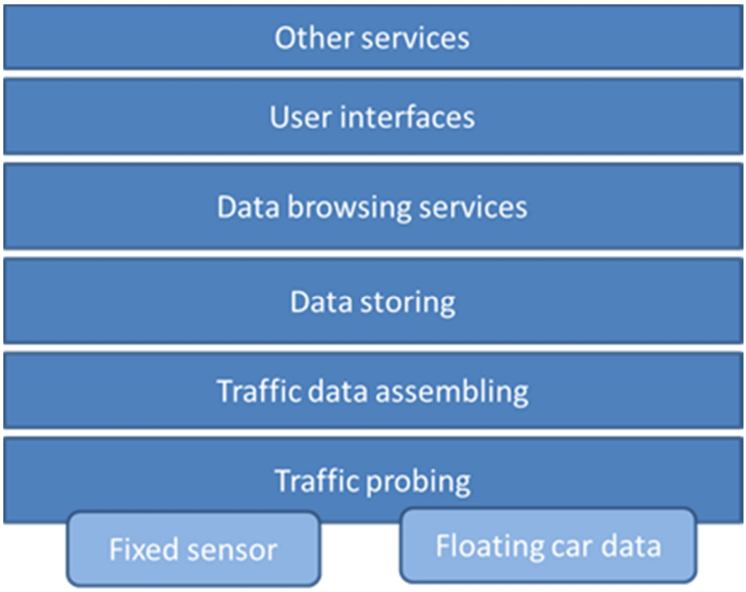

Figure C1. Traffic management subsystem for traffic flow and weather data monitoring.

For each of these subsystems, more precise layered system architectures are described with high-level functional interface definitions. The general structure subsystems monitoring traffic flows and weather data are common to both cases and illustrated in Figure C1.

Lihark's architecture report (Tiehallinto 2005) describes interfaces between layers explicitly when possible. The report specifically mentions the need to develop an FCD interface between the traffic data assembling layer and the traffic probing layer, but neither the actual protocols nor the data formats have been described.

The report also concluded that the important future goals for traffic flow monitoring are to produce information about travel times and develop means of automatic traffic disturbance detection. In these areas FCD might have a focal role once enough high-quality FCD is available. The first practical steps towards these goals have already been taken. In autumn 2012, the Finnish Road Administration announced a public competitive bid for producing travel time information based on cellular FCD on Finnish road networks as a part of DigiTraffic services.

In the Finnish LihArk architecture, the weather subsystem has the same layers (probing and data assemby) as the traffic flow monitoring sub-system architecture, but the extended FCD information has not been recognized as a source of weather data. Fixed weather stations use XML-based formats over various transfer protocols (e.g. HTTP, FTP and SMTP) to convey measurements between probing and data assembly layers. When purchasing weather data from the various sources, this simple XML-based format is required and in fact concerns also extended FCD when this is used to deliver weather information.

The TosiArkki project aimed at clarifying roles and responsibilities in real-time transport information by describing possible value chains for real-time traffic information services, and to elucidate the effects of realizing these services. The architecture describes the service transport mode-independent processes for transport and public transport networks. One of the suggestions for future development that emerged was “distributed traffic information collection”. This method would exploit FCD in such a way that multiple parties could produce a traffic information system for their own repositories, whence data could delivered to the potential user via a standardized interface.

C2. Requirements for the FCD data set

As mentioned above, well-defined interfaces for delivery of FCD and xFCD data is needed for traffic management and weather monitoring purposes in the Finnish ITS system. An interface can be defined officially as a tool and a concept that refers to a point of interaction between boundaries across two independent systems. The interface consists of a common language that both parties can understand and a set of rules that enables communication between the parties. In computer systems an interface contains the structure of messages (data formats) and data transfer protocols to enable exchange of these messages (as protocol payload units).

The *de facto* standard communication protocols today are Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP) and File Transfer Protocol (FTP) to enable data transmission between heterogeneous systems. Specific application protocols with application related messages (with a fine structure and encoding) are conveyed over these generic communication protocols.

In this study we cannot predict all the applications that would use FCD and xFCD information; thus we cannot define all the interfaces needed by FCD applications either. Nor can we define the probe data collection architecture in detail (is it push or pull based etc.). Therefore the main task is to define appropriate data formats for information exchange between communicating parties enabling the transfer of FCD and xFCD information in general.

Besides the ability to carry FCD and xFCD data, the ideal data format should be the following:

- Open; e.g. the definition of the format should be publicly available and have various rights to use associated with it.
- Data items (probe attributes) included should be widely supported from different sets of existing and forthcoming probing devices.
- The format should respond to the data transfer needs of different architectural alternatives like peer-to-peer (P2P, including vehicle-to-vehicle communications), client-server (thin and thick clients, vehicle-to-infrastructure (V2I) etc.) and server-to-server communications.

- Extendable; given the wide range applications that could exploit FCD in the future, the format should be extendable and possible to refine later in a manageable way. Especially xFCD might provide information from vehicle sensors in the future that we cannot predict today. For example, current on-board diagnostic (OBD) systems are able to produce over 250 different vehicle- and environment-related measurements; and advanced sensing elements like camera-based night vision systems and collision avoidance warning systems, which would add further data, have yet to be widely adopted.
- The format should be standards-compliant to ensure interoperability with current and future ITS services. In Europe this means compliance with ISO standards and the Frame and national TelemArk/LihArk/TosiArkki architectures.
- The data format should be unambiguous, easy to understand and definable by current standard means of expression, like extensible mark-up language (XML) or its equivalent.
- Processing the data format should be fluent and adequate processing tools (software, libraries) should be readily available.

More specific requirements for FCD and xFCD systems have been addressed in European Union projects like GST (Enhanced FCD approaches enabled by the Global System for Telematics) (Roos & Burkert 2007), and the co-operative traffic project COOPERS (Scheider & Böhm 2010). COOPERS divided FCD attributes into continuous and event-triggered data, shown in Table C1.

Table C1. Division of continuous and event based floating car data examples according to COOPERS- project.

Continuous vehicle data	Event-triggered data
Vehicle position	Break status
Vehicle heading	ABS/ESP status
Vehicle speed	Wiper status
Vehicle temperature	Hazard warning flasher

Some of the mandatory FCD attributes for a real-time traffic state model were suggested in a Finnish xFCD preliminary study in 2005. The background assumption of this study was that a vehicle's probing device is pretty simple and cannot perform any dedicated computing. All intelligence is centralized to cloud services at the expense of increased communication costs (and need of reliability). In this case the transferred data set should include at least (Kosonen & Pahlman 2005):

- A unique identifier (from which the vehicle can be identified)
- A timestamp

- Location
- Current speed
- Current travel direction
- xFCD options that might include ABS activation (with small braking power), wipers on, fog lights on, an outdoor temperature etc.

As the complete list of attributes was never finalised nor described more closely in the report, the presented set should only be considered as an initial sketch.

FCD and xFCD information exchange might be supported by several architectural systems like P2P or client-server architectures. Mobile ad-hoc networks (MANETs) are P2P-like, self-configuring, infrastructure-less networks of mobile devices joined by wireless connections. Compared to “normal” P2P where network topology is restricted by “known IP addresses”, MANETs are topologically constrained by a radio range. The idea of MANETs is applied to traffic as vehicle area networks (VANETs). Several European Union projects like COOPERs and COMeSafety have studied vehicle-to-vehicle (V2V) and V2I communications based on wireless local area network technologies.

Several standardization bodies are working with VANETs. The ISO approach is called CALM (Communication Access for Land Mobile). ISO TC 204/Working Group 16 has specified a common architecture, network protocols, and communication interface definitions using various access technologies. The European automobile industry Car to Car Consortium (C2C-CC) is also aiming to develop a de facto standard for VANETs using the C2CNet protocol. The US IEEE 1609 WAVE standard defines the architecture, organization, management structure, communication model, security mechanisms and physical access.

All of the previous standards enable communication in P2P-like manner, and xFCD data could be included as a possible payload in protocol messages. The client-server architecture does not set any restrictions either on what can be transferred between parties when standard protocols like HTTP, SMTP and FTP or TCP/IP are used.

C3. ISO floating car data standardization work

The ISO technical committee TC 204 working group WG16 is working on CALM family standards, to which issues concerning FCD are closely related. As part of the standardization work of ITS, ISO has released the ISO 22837 standard “Vehicle probe data for wide area communication”. This ISO standard defines the reference architecture for probe vehicle systems and probe data. The architecture provides a general structure for probe vehicle systems to (ISO 2009):

- Clarify the major building blocks and logical interconnections of probe vehicle systems
- Categorize probe data in accordance with the information model

- Basic data framework for probe data elements and probe data. This framework specifies how to define probe data elements and probe messages.

The adapted ISO 22837 architecture reference model is described in Figure C2. A probe data system contains a set of vehicles equipped with sensors that collect and transmit different types of sensor data using a telecommunication system to convey probe data elements to the data collection centre for further processing.

In ISO 22837 terminology “onboard data source” means either a vehicle’s sensor or other application that can produce probe data. The probe data element generator creates probe data elements from raw sensor data by “normalizing” (processing) it or leaving it in its original form. the probe message generator collects probe data elements as sets, and formats them for transmission to the probe collection element. Vehicles can retain probe data in a local storage called a “referenced data repository”.

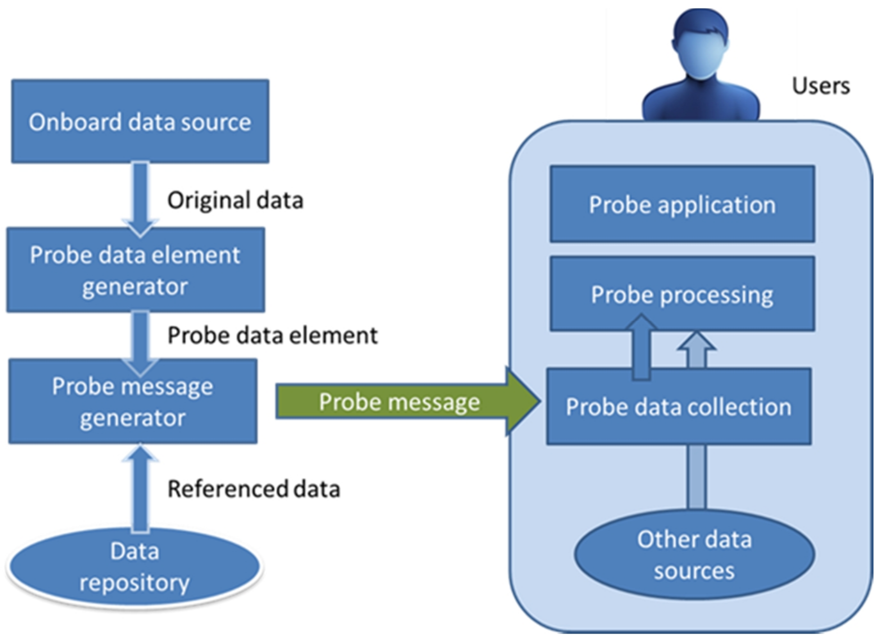

Figure C2. An adapted version of the ISO 22837 conceptual architecture model (ISO 2009).

The information models show the logical structure of entities and concepts involved in probe data:

- The notation for probe data elements/messages in XML
- Rules for using core data elements (e.g. basic descriptive elements intended to appear in every probe message like position and timestamp)

- Rules for using basic data elements and extensions of data elements in application domains such as traffic, weather, and safety
- Required characteristics (attributes) of probe data elements and probe data messages.

The ISO standard provides rules for mapping information models of probe data to probe data elements and messages. The basic data elements of probe data are called “probe data elements” and a combined set of multiple probe data elements is called “a probe message”. Each probe data element has mandatory common attributes as core elements (Figure C3). These are:

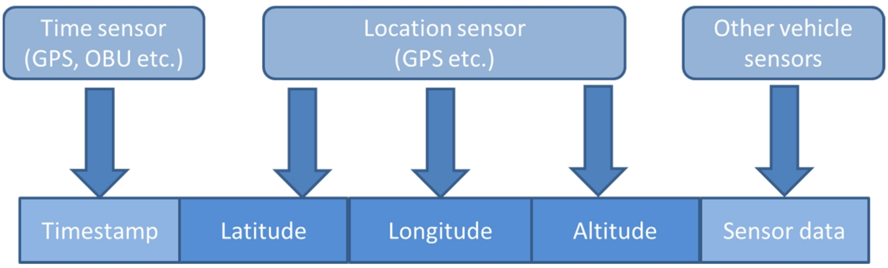

Figure C3. Abstract structure of ISO 22837 probe data element (ISO 2009).

- A timestamp when the sample is in Unix epoch time format (usually available from GPS)
- Positioning data (latitude, longitude, altitude) in decimal degree formats, except for altitude, which is given in metres (positive/negative in relation to sea level)
- Actual sensor data that is “normalized” e.g. presented in a format that ISO 22837 explicitly defines.

The modelling aspect in ISO 22837 is object oriented, i.e. all data entities in the model are presented as object classes and attributes. In addition to core elements, the standard defines sensor data objects that are modelled as three separate packages:

- Vehicle package defines data elements related to it and its components like chassis, engine etc.
- Surrounding package gives the properties of the environment surrounding the vehicle
- Road Network package defines the properties of the road network.

All these packages have their own object model with a fine structure and set of characteristic attributes. The attribute set is pretty comprehensive; for example a composite vehicle class has a related object called BrakingSystem, which in turn has related objects like AntiLockBrakeSystem. Depending on the class, an attribute count varies from one to dozen. Each attribute has a defined type (like Integer) and a valid value range that can be continuous or enumerated.

The standard also defines how to represent these elements in XML, but an actual XML schema definition is not given. ISO 22837 does not use any XML name space either, rather the XML elements are recognizable by their ASN.1 object identifiers. This Abstract Syntax Notation One (ASN.1) is another ISO standard that describes rules and structures for representing, encoding, transmitting, and decoding data in telecommunications and computer networking. ASN.1 notation is also used inside the XML element content to structure some probing attribute values (Figure C4).

```
<probe_data_element descriptive_name="Wiper.status:integer">
<ASN.1_name>Wiper-status</ASN.1_name>
<ASN.1_object_identifier>{ 1 0 22837 000 035 }</ASN.1_object_identifier>
<definition>activation of wipers and operational mode (intermittent, slow,
fast)</definition>
<descriptive_name_context>probe</descriptive_name_context>
<data_concept_type>data element</data_concept_type>
<standard>EMPTY</standard>
<data_type>INTEGER</data_type>
<format>0 = no wiper active, 1 = intermittent, 2 = slow, 3 = fast</format>
<unit_of_measure>concatenated code</unit_of_measure>
<valid_value_rule>integer [0...3]</valid_value_rule>
<data_quality>n.a.</data_quality>
</probe_data_element>
```

Figure C4. ASN.1 coding used inside the XML element structure (ISO22837).

Surprisingly, ISO 22837 does not provide a means for a vehicle's individual identification, which makes trajectory path construction very hard or even impossible. The dropping of IDs seems to be deliberate, since uniform ID generation has some problems related to uniqueness and privacy (as indicated later).

C4. Other FCD formats

Publicly described FCD datasets and formats seem to be pretty rare, to tell from an information-retrieval search, but at least two examples were found. The first relates to a taxi fleet probe system in Shanghai, and the other to an EU Project called Viajeo and a national Italian project called S.I.MO.N.E (executed in Torino area).

In the Shanghai taxi-probe system, real-time location data were collected at 2-second intervals and sent to the traffic centre every 30 seconds. Based on the collected probe data, the real-time traffic situation was calculated for each road segment within 5 minutes (Liu et al. 2012). The data format used in the Shanghai taxi fleet FCD is shown in Table C2.

Table C2. Shanghai taxi fleet probe data fields.

Field	Value example	Details
Date	20121112	8-digit number
Time	080001	6-digit number
City	SH	2-digit letter
Vehicle marking	33072	Vehicle marking of operating company
Longitude	121.52760	Decimal degrees (5 decimals)
Latitude	31.28360	Decimal degrees (5 decimals)
Velocity	34	In km/h, -1 if velocity is null
Direction angle	157	In degrees, -1 if velocity is null
Operation	0	0 = empty, 1 = passenger departure
Available	0	0 = available, 1 = unavailable, -1 if velocity is null

The goal of the Viaejo project (2010–2013) (Sarros et al. 2012) is to design an open platform with interfaces to a wide range of mobility services. One aim of the project was to upgrade the management of the taxi fleet in Athens and alert drivers to incidents that may affect taxi services. Some taxis were equipped with a TBOX probe device for the acquisition and dispatch of FCD to the VIAJEO Service Centre. For FCD acquisition the Viajeo project used the SIMONE protocol developed in the S.I.MO.NE (Innovative System for Management of Mobility in Metropolitan Areas) project in Italy. One of the main aims of S.I.MO.NE (Cocozza et al. 2009) was development of technologies to collect data from moving vehicles (FCD) and harmonizing protocols related to it for the convergence on a common ICT platform.

The SIMONE protocol is defined in accordance with the FRAME architecture and contains six message structures for different purposes:

- Raw data
- Map-matched raw data
- Travel time data
- Origin destination matrices
- Traffic events
- Limited traffic zone data.

Each protocol message is defined from different utilization needs, and depending on the “intelligence” in the vehicle device, different datasets can be provided. Raw data (RD message) is the simplest and also supports “events” (in the COOPERS sense) usually associated with xFCD. Other messages need embedding of road network information into the vehicle device and more complicated calculations performed in the vehicle unit.

SIMONE data elements (example in table X.) rely partially on the NMEA-0183 standard protocol, which is a serial protocol originally developed for data interchange of marine instruments. Currently, numerous Bluetooth-enabled GPS devices support NMEA-0183 to enable the acquirement of positioning data. NMEA-0182 protocol messages and data items (comma separated values) associated with them were defined by Betke (2000). SIMONE protocol messages use these NMEA-0183 data values directly as data elements.

Table C3. SIMONE raw data element structure.

Field Name	Description
Veh	Anonymous vehicle ID
Timestamp	Sampling time referenced to GMT (originated from NMEA-0183)
Lat	Latitude in WGS84 decimal degree format (NMEA-0183)
Lng	Longitude in WGS84 decimal degree format (NMEA-0183)
Alt	Altitude (m)
Heading	Direction respective to geographic north
Speed	Instantaneous speed km/h
Hdop	Precision factor (NMEA-0183 Horizontal Dilution of Precision)
Event	Triggering event (extended floating car data information)
Tracking_distance	Odometric distance respective to the previous measure (m)
Tracking_type	Type of sampling: time, space, mixed
Vehicle_type	Vehicle_type: car, truck, bus, tram etc.
Vehicle_information	Various vehicle parameters: engine type, kW, class etc.

C5. Candidate for minimum data set

Essentially, the basic dataset for FCD seems to be pretty equal in both standardization and practical applications, but the structuring and combination (framing) of messages deviate slightly. In the Finnish case, a minimal common dataset should enable at least probing information for traffic flow analysis and probably also road weather services, but the data format should be extendable to carry all the xFCD data variations. Thus the definition should be pretty flexible. The ISO standard could be used as a guideline to achieve these goals.

Using the ISO 22387 terminology, the first task is to define a Finnish core dataset (FCDS). This is an absolute minimum data element that is transferred within the Finnish ITS architecture. In accordance to the Finnish preliminary study, the basic assumption is that the probing device should be pretty simple without requirements for advanced computing capabilities. In practice, a probing device could be a GNSS capable device that can produce WGS84 compliant information, giving it the ability to tap a wide variety of information sources.

Another source of FCD could be floating cellular data provided by operators. As stated in “Case study: FCS data…” several possible methods of gathering cellular FCD have been developed, and the minimum dataset should be able to support all of them. Since a commensurable presentation method with GNSS-based data should be provided, a location estimate should be given in WGS84 format, and some estimation of accuracy related to positioning is also essential. For example, a cell ID (CID) is quite coarse in general but in city areas pico-cells might give accuracy to within 20 metres, which is comparable to GNSS-based positioning. For this reason accuracy levels, instead of the used positioning technologies, would be a more desirable solution for giving additional information about the reliability of positioning.

The absolute minimum dataset (FCDS) should contain (Figure C5):

- A probe ID (e.g. sample ID) that should be unique and considerate of privacy issues (see later). Additional ID related fields might be needed for privacy support (a trip ID and a sample ID, explained later).
- A timestamp, e.g. indicating when the probe data were sampled. The format is suggested to follow ISO 8601 (e.g. YYYY-MM-DDTHH:SS:MM).
- Positioning data – latitude, longitude and altitude (in strict order), where latitude and longitude are WGS84 decimal degrees in the coordinate system format, and altitude is metres above or below sea level. Note that the coordinate might be artificially generated and the source could be cellular data positioning or some other means.
- The quality of the data classifies its accuracy. Accuracy classes are needed to approximate positioning errors. Here these levels are not defined explicitly, but for example they could be modelled according to Geospatial Positioning Accuracy Standards and be levelled as discrete values.
- The source type of the data can be used to classify its origin, which can refer to further attributes of the sources. This information could be used to provide dedicated services for different transportation modalities like pedestrians, bikes and other means of private and public transportation.

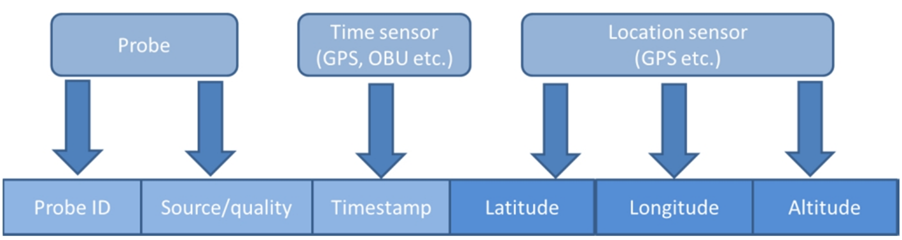

Figure C5. The minimum (core) dataset, FCDS (Finnish Core Data Set).

It should be noted that the minimum dataset is also capable of providing fixed sensor data support. For example, the position of Bluetooth-based probes is well known (and could be given by unique coordinates) and the probe ID could be derived from a Bluetooth MAC address. Also other fixed sensors able to distinguish between unique vehicles (like road-toll stations based on RFID or license plate scanning) could be supported by this minimum dataset.

The probe ID attribute is slightly problematic since it should provide a uniqueness identifier in the system-wide context at least. A probing device can vary from a mobile phone to a vehicle, so the unambiguousness of an ID should sensibly be linked to a physical device address. These MAC-48 (Media Access Control) addresses are globally unique and associated with each Ethernet (WiFi) or Bluetooth network interface. The selection of binding the ID to a MAC address does not imply that the MAC address is actually “the probe ID” related to a sample; more likely it will be a seed for a secured artificial ID generation producing an anonymized probe ID that takes privacy issues into account. RFIDs and licence plates are also unique, and as with MAC addresses they cannot be used as actual system-wide IDs for privacy reasons.

The absolute minimum probe dataset is nearly identical to the ISO 22837 standard, with the exception that a unique probe ID is added. This is done to enable calculation of vehicle trajectories when only a core information set is given. The use of ID raises several important issues that are examined in more detail in the next chapter. Another difference is a timestamp that is not in ISO 22837 UNIX format. Still, the Finnish core set can be easily converted to ISO 22837 by omitting the ID attribute and reformatting the timestamp; other values are directly compatible.

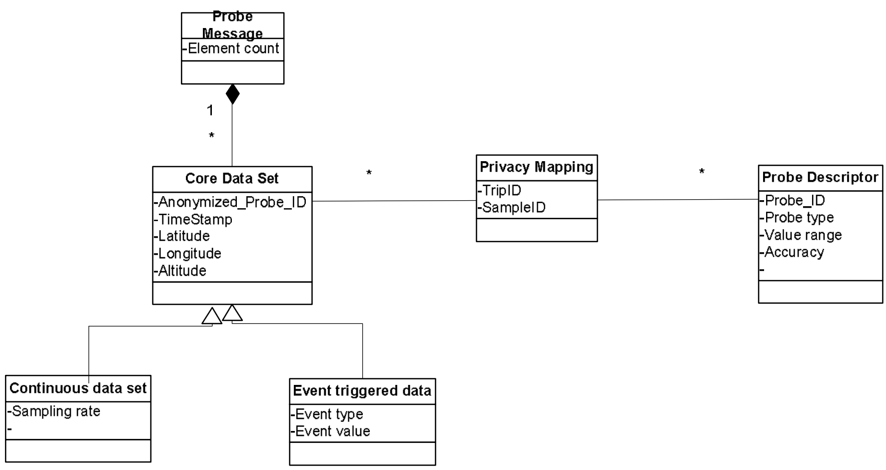

Figure C6. Initial model of the extended dataset.

As stated above, the core dataset enables the use of very simple probing devices, but in order to support more advanced probe devices it should be extended accordingly. Extended sets (Figure C6) are split into continuous data and event-triggered datasets based on how the sample is generated. Continuous data is produced at deterministic sampling intervals, whereas event-triggered data is based on unpredictable occurrences like pressing the brake, opening a door etc.

In the schema, extended datasets inherit core data attributes since all samples have time and space as properties. For each probe dataset, meta-information about the probing device and a possible value range are given as a reference (which in fact does not belong to a dataset). A probe message contains one or more sequential core dataset elements or extended dataset elements (the order of set elements is insignificant).

C6. Privacy, security and probe identifiers requirements

Current location systems can track users automatically and generate an enormous amount of potentially sensitive information. Sensor information provided by a floating car could be used to generate vehicle profiles or even driver specific profiles, unless the data is protected by an intelligent security and privacy system. On the other hand, it is desirable to be able to use valid information (to ensure data integrity) provided by a recognized (authenticated) sensor platform rather than accept information from any source.

Privacy is defined by Wikipedia as: “Privacy (from Latin: privatus "separated from the rest, deprived of something, esp. office, participation in the government", from privo "to deprive") is the ability of an individual or group to seclude themselves or information about themselves and thereby reveal themselves selectively”. Location privacy is a special type of information privacy that can be defined as the ability to prevent other parties from learning one’s current or past location. Privacy of location information is about controlling access to this information.

The European Union Data Protection Directive (95/46/EC) regulates the processing of personal data within the EU regardless of whether such processing is automated or not. According to it, personal data is defined as "any information relating to an identified or identifiable natural person ("data subject"); an identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to his physical, physiological, mental, economic, cultural or social identity.”

Authentication is a process of determining whether someone or something is, in fact, who or what it is declared to be. In a sense it is a process of proving genuineness and there are several ways to perform it. A popular method is use of credential information like username and password to authenticate a user, or use of digital signing to authenticate data originated from a certain user. Highly reliable (and secure) cryptographic authentication methods like digital signature and challenge-response authentication have been developed to alleviate authentication related issues.

A user study by Pell et al. (2012) revealed that most interviewed people do not resist tracking of routes driven with company cars (58% acceptance), but they are more concerned about being tracked when using private ones (45% acceptance). Trepidation is related to illegal data usage and several threats were mentioned:

- Data transformation into anonymous form could fail

- Personal habits and choices could be kept under surveillance
- Personal data could be published on the internet without permission
- Speeding penalties could be communicated with the police.

As the study indicates, privacy is not considered a very serious issue with fleet management systems, but new valuable data sources are becoming available when private vehicles are equipped with GPS units for navigation, automatic emergency systems (eCall) or electronic toll collection. It should be noted that even though users of fleet management systems are not concerned with traceability as a serious issue, opening fleet data to public usage might expose information that the provider's competitors might consider advantageous to know.

Some respondents in Pell's study had the opinion that only public institutions should be able to collect personal mobile data like xFCD, because private companies could sell that data for marketing purposes. On the other hand, some of those interviewed were afraid that public institutions could use the data to fine a driver for speeding.

Another study (Klasnja et al. 2009) indicated that 42% of the participants were concerned about global positioning systems (GPS) recording their position all day. Almost half of the participants in this study disapproved of the use of permanently-stored GPS data, but surprisingly 96% did not object to the use of GPS information deleted immediately after its use. It also seems that younger people are more willing to share their personal information than older ones.

The goals of other stakeholders are the opposite: FCD service providers want to get trajectory information from reliable and identified sources, but drivers want to be anonymous. One concern is that drivers might be unwilling to participate and utilize xFCD-based services when negative effects outweigh possible benefits. According to Pell's study, one of the main problems is a lack of confidence in the process of anonymization.

xFCD anonymization is the problem (or process) of hiding a user's identity from the probe data in such a way that usual patterns or other privacy related information are not revealed. Rass et al. (2008) suggest that use of provable anonymity and protection of private information may convince more active voluntary user participation. The solution proposed by Rass et al. makes the basic assumption that each probe unit (OBU, On Board Unit) has a system-wide (global) unique identifier. From this unique identifier two different types of pseudonyms can be generated. The first is a trip identifier pseudonym (TID) that is directly derived from the OBU_ID, and the second is a sample identifier pseudonym SID, derived from TID. The mathematical handling and proofing is not presented in detail here but the solution proposed provides (Rass et al. 2008):

- Uniqueness of pseudonyms, i.e. no two identical pseudonyms may exist at any given time
- Hiding of the user's identity, i.e. the owner's OBU_ID cannot be derived from pseudonyms

- Unlinkability of pseudonyms, i.e. two pseudonyms cannot be identified as belonging to the same owner
- Optional lnking, i.e. pseudonym linking is possible with the owner's explicit permission in the case of legal action.

Rass et al.'s proposal is targeted to FCD information and pretty easy to implement. The only practical issue needing some additional attention might be the system-wide uniqueness of OBU-IDs. As mentioned earlier, pretty much any communication capable device has some unique ID, like Ethernet or Bluetooth MAC identifier or similar that can be used as a seed of unique ID generation.

The definition of a trip also needs some attention. A trip is officially defined as "going from one place to another; a journey" e.g. an entity having a start and an endpoint (in time and space). A trip associated with the use of a motor vehicle like a car could be pretty easily defined as a session from starting an engine to shutting it down, which should also easily be detected by most vehicle bound probing devices. But a mobile phone or other continuously used hand-held source of traffic information raises the question of how a trip (or session) should be defined and detected from a series of GPS or cellular positioning location samples.

One of the GPS raw data post-processing tasks is activity and mode detection (Schuessler & Axhausen 2007, 2008). Activity detection tries to resolve whether a certain GPS sample belongs to an activity or not. One resolving technique is use of a criterion called dwell time, i.e. a minimum time difference between two GPS samples after it is assumed that an activity took place. Another one is to utilize bundles of GPS points, i.e. points that locate very near each other, and to conclude sample inclusion based on that. Another heuristics method is to measure a zero speed (>0.01 m/s) for a certain time and direction changes near 180 degrees. Methods related to GPS data post processing are still under investigation, but for example by tuning previous inclusion parameters accordingly and using fuzzy logic, it is pretty reliable to detect trips and used travelling modes (Upadhyay 2008, Xu et al. 2010).

Another anonymization principle (Eichler 2006) related to FCD was developed in the GST (Global System for Telematics) EU project. The approach is based on the idea of Control Centres (CC) operated by vehicle manufacturers and all collected xFCD data; these centres manage all service subscriptions and user preferences. Also all authentication and authorization related issues based on a public-key infrastructure (PKI) and certificates for authentication and authorization operations are managed by CCs. User interface and vehicle systems are modelled as the Client System (CS) that provides services to the user and collects FCD from vehicle. The third system block is the Service Centre (SC), which is operated by a service provider, and multiple SCs can be related to the same CC providing their services to subscribed users.

The security architecture related to Eiclher's study uses the circle-of-trust concept for providing single sign-on functionality to users. This means that after successful authentication, a user can access all the subscribed services irrespective of the service centre providing the service (Eichler 2007). FCD security is based on an idea where service providers reward users for the amount of data they submit, so the service centre needs to identify the origin of a data message.

Hoh and colleagues (Hoh et al. 2006) have raised the question of whether anonymous data collection is enough to protect user privacy. Current clustering algorithms are so efficient that they can smooth out noisy GPS traces and automatically identify often-visited places. For example, location samples with near zero speed are good candidates for trace endpoints, and estimated destinations can be derived from this information. See more about moving estimations presented by Patterson et al. (2003). Privacy risks decrease when the sampling rate (a probe device's position updating) is lowered, but as mentioned earlier, the quality of traffic situation estimations will fall.

C7. Summary

The aim of this study was to try to identify a minimum FCD dataset that enables observation of traffic situations in real time. General requirements for a FCD-based system were examined and an "absolute minimum dataset" was constructed. The dataset contains only uniquely identifiable location samples with a timestamp and some location quality attributes. The interpretation of these measurements is left to data mining applications that should be able to deduce desirable results like congestions, travelling speeds etc. on road-segment based samples. To enable these measurements, sampling values should be associated with certain IDs for trip construction, which will cause some privacy issues.

The xFCD platform (or more generally ITS platform) needs user identification, authentication, and authorization as well as communication security. If very rigid security and privacy qualities are needed, it also affects the system architecture, because data collection and storage elements should be separate entities using strong public key cryptography and infrastructure (PKI) for user authentication and privacy (Hoh et al. 2006). ETSI has also defined security service and architecture guidelines for ITS services based on PKI (ETSI 2010). The document is more like a functional description of security elements related to vehicle to vehicle/infrastructure and on a conceptual level defines entities related to secure messaging.

Problems related to privacy and probe identifiers need serious further study and should be taken into account when ITS system architecture is being designed and implemented.

C8. References

Betke, K. 2000. The NMEA 0183 Protocol. 28 p. http://www.elisanet.fi/master.navigator/nmeadescription.pdf.

Cocozza, M., Foti, G. & Arneodo, F. 2009. S.I.MO.NE. Innovative System for Metropolitan Area Mobility Management. 16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services. 8 p.

Eichler, S. 2006. Anonymous and authenticated data provisioning for floating car data systems. In Proceedings of the 10th IEEE International Conference on Communication Systems (ICCS), October 2006.

Eichler, S. 2007. A Security Architecture Concept for Vehicular Network Nodes. Information, Communications & Signal Processing, 2007 6th International Conference on. 10–13 Dec. 2007. Pp. 1–5.

ETSI 2010. Intelligent Transport Systems(ITS); Security; Security Services and Architecture. ETSI TS 102 731 v.1.1.1. 2010-09. 67 p.

Hoh, B., Gruteser, M., Hui Xiong & Alrabady, A. 2006. Enhancing Security and Privacy in Traffic-Monitoring Systems. IEEE Pervasice computing, Vol. 5, Issue 4. Pp. 38–46.

ISO 2009. Vehicle probe data for wide area communications. International Standard ISO 22837:2009(E). 60 p.

Klasnja, P., Consolvo, S., Choudhury, T., Beckwith, R. & Hightower, J. 2009. Exploring Privacy Concerns about Personal Sensing. Proceedings of the 7th International Conference on Pervasive Computing (Pervasive '09). Springer-Verlag, Berlin, Heidelberg. Pp. 176–183.

Kosonen, I. & Pahlman, S. 2005. Ajantasaisen liikennetilamallin tuottaminen floating car -datan avulla. Esiselvitys. Aino-julkaisuja 12/2005. 57 p.

Kummer, S., Edlmair, R., Fürst, E. & Wilhelm M. 2012. Spin-Off Functions of XFCD Technology for Public Use in ITS. In Proceedings of the 19th ITS World Congress, Hrsg. ITS, 22-26. Wien Spin-Off Functions of XFCD Technology for Public Use in ITS. 19th ITS World Congress, Vienna, Austria, 22/26 October 2012. Pp. 22–26.

Liu, C., Li, N., Huang, M. & Wu, H. 2012. City Routing Velocity Estimation Model under the Environment of Lack of Floating Car Data. Journal of Geographic Information Systems, Vol. 4., No. 1, pp. 55–61.

Mäkinen, P., Ruoti, K., Lähesmaa, J., Lehtonen, M., Oinas, J., Ristola, T. & Appel, K. 2000. Liikennetelematiikan kansallinen järjestelmäarkkitehtuuri – TelemArk. Arkkitehtuurikuvaus. Liikenneministeriön mietintöjä ja muistioita B 5/2000. 94 p. http://www.transportal.fi/Hankkeet/tetra/TETRA/H8/t8_15.pdf.

Patterson, D., Liao, L., Fox, D. & Kautz, H. 2003. Inferring High-Level Behavior from Low-Level Sensors. UbiComp 2003: Ubiquitous Computing Lecture Notes in Computer Science, Vol. 2864, pp. 73–89.

Pell A., Starkl, F. & Menrad, M. 2012. A Field Study on the Acceptance of Extended Floating Car Data for Real-time Monitoring Traffic Conditions. Proceedings of the 2012 IEEE International Symposium on Sustainable Systems and Technology, Boston, MA, USA. Pp. 122–125.

Rass, S., Fuchs, S., Schaffer, M. & Kyamakya, K. 2008. How to protect Privacy in Floating Car Data Systems. Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking (VANET '08). ACM, New York, NY, USA. Pp. 17–22.

Roos, A. & Burkert, A. 2007. Enhanced floating car data approaches enabled by the Global System for Telematics. GST EFCD project. 16 p.

Sarros, G., Tyrinopoulos, Y., Milli, M., Locuratolo, L., Mizaras, V., Chatziathanasiou, M. & Kortsari, A. 2012. Open traffic data collection methods for transport strategic planning and travel information in large metropolitan cities; the case of Athens. 19th ITS World Congress, Vienna, Austria, 22/26 October 2012. 7 p.

Scheider, T. & Böhm, M. 2010. Extended Floating Car Data in Co-operative Traffic Management. In: Barceló, J. & Kuwahara, M. (eds.) Traffic Data Collection and its Standardization. International Series in Operations Research & Management Science 144. Springer Science+Business Media, LLC. Pp. 161–170.

Schuessler, N. & Axhausen, K. 2007. Identifying trips and activities and their characteristics from GPS raw data without further information. IVT ETH Zurich. 28 p. http://e-collection.library.ethz.ch/eserv/eth:30471/eth-30471-01.pdf. Accessed 16.10.2013.

Schuessler, N. & Axhausen, K. 2008. Processing GPS Raw Data Without Additional Information. IVT ETH Zurich. 16 p. http://e-collection.library.ethz.ch/eserv/eth:30636/eth-30636-01.pdf. Accessed 16.10.2013.

Siponen, A., Higgins, A., Lehtonen, M.J., Levo, J., Lähesmaa, J., Mäkinen, P. & Öörni, R. 2005. Ajantasaisen liikennetiedon arkkitehtuuri. Loppuraportti, VIKING MIP 2005: Real-time Traffic Information Architecture Final Report. AINO-julkaisuja 20/2005. 117 p. In Finnish. http://www.durbit.se/Archives/ExternalPDF/MIP2005/Domain8/Deliverables/VIKING_MIP05_D8_1_S_96_Real_time_TI_architecture.pdf.

Tiehallinto 2005. Liikenteen hallinnan arkkitehtuuri – Tavoitteellinen järjestelmäarkkitehtuuri Tiehallinnon sisäisiä julkaisuja 45/2005. 32 p. (in Finnish). http://alk.tiehallinto.fi/julkaisut/pdf/4000481-v-liikhall_jarjarkkiteht.pdf.

Upadhyay, D., Schuessler, N., Axhausen, K.W., Flamm, M. & Kaufmann, V. 2008. Optimal parameter values for GPS post-processing: An experiment. Working paper, 507. ETH Zurich, Zurich.

Xu, S, Ji, M., Chen, W. & Zhang, Z. 2010. Identifying Travel Mode from GPS Trajectories through Fuzzy Pattern Recognition. 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2010). Pp. 889–893.

Appendix D: Commercial multiservice offerings

In-car information and entertainment systems, commonly called IVIs (In-Vehicle Infotainment), were once among the main differentiators of premium end cars. Nowadays the biggest trends in the car industry include bringing this technology to all car models to access phones, PDAs, music, maps, etc. through dashboard controls or via mobile phones (Stockwell 2006). All leading car manufacturers are developing their own proprietary systems that are not inter-operable. According to ABI research the total number of “OEM-installed connected car telematics systems,” including IVI systems, will increase from around 7.8 million at the end of 2012 to 46.8 million units by the end of 2018 (Brown 2013).

IVIs have a main unit that contains a CPU and memory, a digital signal processor (DSP) for processing audio, MP3 decoding, graphic computation for 2D effects, and a GPS receiver to support navigation. Odometers, other sensors and controllers are usually connected by a CAN bus. The operating systems applied are embedded ones such as QNX (Poliak 2011), VxWorks or Microsoft Windows Auto. Recently also a platform-independent OSGi and Java-based service framework has gained popularity mostly to ease system management and service updates.

LCD displays are used in a centre console or dashboard, and auxiliary display units for passengers serve entertainment purposes. The user interface for IVIs is either based on touch screens (menu driven selection) or drive controls located on a console (or partially in a driving wheel) as well as voice control (Figure D1).

Figure D1. IVI system installed in the centre console of a Merzedes Benz.

Two main business sectors of IVIs developments can be identified as:

- Enhanced audio systems, moving beyond conventional car radios
- Safety and comfort systems including driver assistance, navigation and communication.

The latter will make these systems directly competitive with “standardized ITS services” and multiservice platforms. Considering these safety and comfort related services, GM’s OnStar is the oldest one (launched in 1995) and has the widest user base. GM OnStar can be either inbuilt in a car or can be purchased as an aftermarket unit.

Ford SYNC (Ford 2013) is a factory-installed, integrated in-vehicle communications and entertainment system that runs on the Microsoft Windows Embedded Automotive operating system and was developed jointly with third-party developers. Ford had a 2-year exclusivity agreement with Microsoft that has now expired. Kia Motors is developing a rival to SYNC called UVO (Kia 2013), which is based on the same Windows platform and is expected to appear in the near future. Other manufacturers that use this Microsoft’s Windows CE derivative (Microsoft’s Windows Mobile for Automotive or Windows Embedded Auto) operating system are Fiat (Blue&Me) and Nissan.

Toyota has also recently introduced an infotainment system that's similar to SYNC called Entune (Toyota 2013). All other main manufacturers also have their own solutions: Audi’s system is based on QNX Neutrino RTOS (Real Time Operating System) and BMW, Daewoo, Honda, Porche and Renault are also using it. GM’s Onstar is also QNX based, running both Neutrino and QNX developed Aviage Acoustic Processing Suite. Mercedes Benz has introduced a system called MBrace2 (MBrace 2013), which has similar functionalities to the other systems.

In the following chapters we briefly describe some of the features and future directions of these systems. The technological details and architectures of these systems are mostly proprietary; unfortunately, publicly available information is minimal for non-certified developers thus only an overall description could be given. Since several car manufactures are using Microsoft Embedded Auto, we take a brief look at the system architecture of this platform.

D1. GM OnStar

General Motors founded OnStar (OnStar 2013) in 1995 to bring traffic-related telematics services to customers via cellular networks. OnStar provides subscription-based communications, in-vehicle security, hands-free calling, turn-by-turn navigation and remote diagnostics systems using CDMA mobile phone voice and data communication. Location information provision is based on GPS integrated with the vehicle’s ABS module. The ABS module sends differential wheel speed and directional information to the OnStar system to provide location fixes even in environments where GPS satellite information is not available.

Figure D2. OnStar integrated as part of the Chevrolet Volt infotainment system.

The OnStar service relies on a cellular network that drivers or passengers can use to contact representatives for emergency services or directions. The main emphasis is user security. If a collision is detected by airbag triggering or other sensors, Advanced Automatic Collision Notification features can automatically send information about the vehicle's condition and GPS location to OnStar call centres. In case of an emergency a user can also contact the service centre manually with a push button.

OnStar turn-by-turn navigation allows drivers to call an OnStar operator and ask for audio directions. Directions are downloaded to the vehicle unit and a computerized voice guides the driver to their destination. Audio directions are automatically played through the vehicle’s audio system synchronised with OnStar’s GPS capabilities. It is claimed by OnStar that the system is easy to use and safer than screen-based navigation, because drivers never have to take their eyes off the road.

Figure D3. GM OnStar FMV aftermarket unit replacing a standard rear-view mirror (courtesy of OnStar).

The OnStar Vehicle Diagnostics System automatically performs certain diagnostic checks on GM vehicles covering engine/transmission, anti-lock brake and airbag, and automatically sends the results via email to the owner each month. The email report also gives maintenance reminders based on the vehicle's odometer readings, remaining engine-oil life and other relevant vehicle ownership information. OnStar subscribers can also request a Remote Diagnostics check at any time through the OnStar system. These real-time remote diagnostics checks are used by subscribers about 27,000 times a month.

Over the years, OnStar has added features such as the ability to open locked doors remotely and even track stolen vehicles, providing the police with the vehicle's exact location, speed and direction of movement. OnStar also provides the Stolen Vehicle Slowdown feature, which allows the stolen vehicle to be slowed remotely. Another feature is Remote Ignition Block, which enables remote deactivation of the ignition, preventing the stolen vehicle from being restarted once it is shut down. Some insurance companies recognize these features and offer discounts to OnStar users. With the subscriber's permission, OnStar may also send the vehicle's odometer reading every month to an insurance company and a low-mileage driver may benefit from an insurance discount.

At first, OnStar was available only for certain GM car models in the USA, Canada and China and the same system was marketed as ChevyStar in Latin America. Current OnStar units are either fully integrated in certain car models manufactured by GM, or can be purchased as aftermarket devices (OnStar FMV) in the form of a rear-view mirror that can be installed in non-GM models.

The aftermarket version does not have all of the features of the embedded version. It does not have access to the car's OBD-II port, and thus cannot offer remote door unlock or engine start or remote diagnostics and vehicle health. These features are therefore absent from the OnStar FMV service pallet. Collision detection is supported by the mirror's accelerometer, which triggers OnStar's emergency collision response. For non-GM cars the FMV unit is marketed essentially as a safety system.

GM has claimed that over the past 15 years OnStar service centres have resolved 346 million OnStar-related events including over 160,000 vehicle crashes, unlocked doors more than 5 million times, provided 2.6 million customers with roadside assistance, and routed customers to their destination more than 70 million times.

In 2011 OnStar kicked off a student developer challenge “To find the next big in-vehicle, voice-enabled application that would provide a greater level of connectivity to its subscribers”. Participants were from MIT, Carnegie-Mellon and several other universities and were using OnStar API services. The winning team, from MIT, developed an application called EatOn (MIT 2011) that automatically identifies restaurants nearby and allows drivers to listen to their reviews and ratings, make table reservations, and receive turn-by-turn directions straight to the restaurant via a voice interface. OnStar has also piloted an OnStar Voice Communications App for Android mobile phones that works in GM cars with Bluetooth or the FMV mirror. Via Bluetooth drivers can send and receive text messages and Facebook updates using voice-only.

OnStar has also released a RemoteLink/MyLink app (Figure D4) for Android and iPhone that co-operates with the car’s onboard OnStar system and offers location-based services, such as a GPS parking reminder and vehicle reference material. The MyLink app also supports a searchable database of warning lights, dashboard indicator lights and a variety of commonly searched features from the owner's manual. A customized version for Chevrolet, RemoteLink (myChevrolet), allows users to remotely lock or unlock doors, start the engine and access a wealth of diagnostic information. The application shows up-to-date vehicle information such as oil level, tyre pressure, fuel level and lifetime miles per gallon.

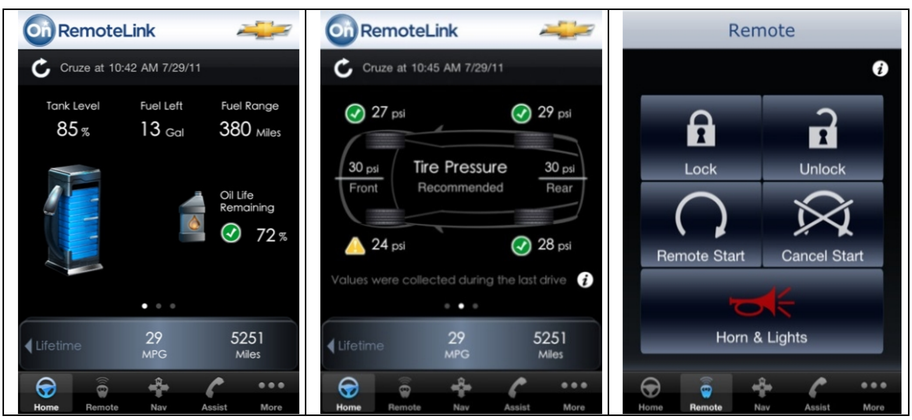

Figure D4. OnStar RemoteLink app for iPhone (courtecy of OnStar).

OnStar’s RemoteLink app also has integrated support for navigation. The service subscriber is able to search a destination by using the mobile phone and can then send it directly to the car’s unit. The destination can be selected either by typing it in, via a voice interface, or a search of the contacts list. While driving the user receives audio directions via OnStar Turn-by-Turn or through a dashboard navigation system. The apps have been downloaded more than 450,000 times in North America and have generated more than 4.5 million service interactions.

Recently (in 2012), GM also announced opening up of OnStar to 3rd party developers, much like Apple's iPhone or Google's Android markets. OnStar services are enabled by its Advanced Telematics Operating Management System (ATOMS) cloud platform. OnStar will make its API available to anyone, but GM will have exclusive rights to approve or deny participation for security reasons (e.g. developers cannot develop apps that would endanger drivers' safety).

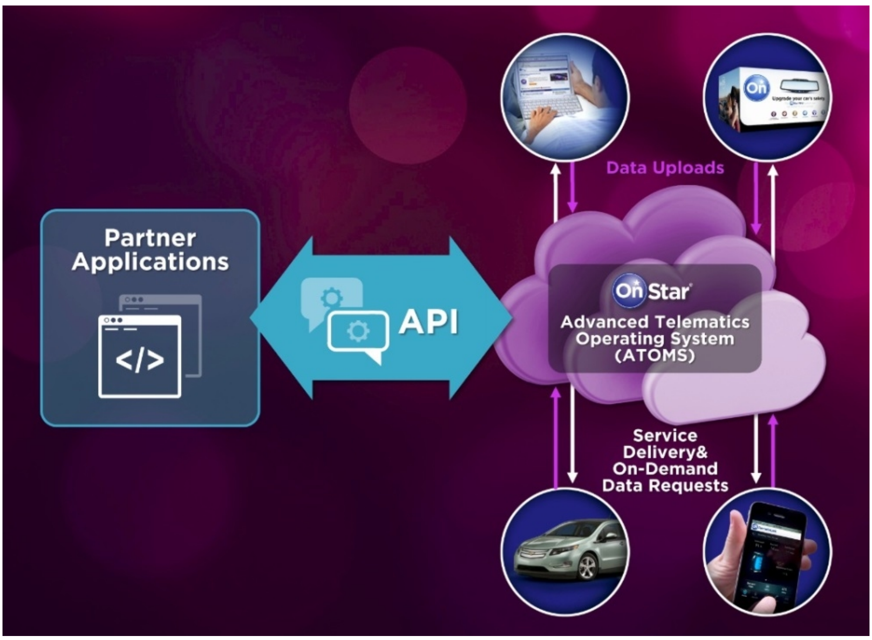

Figure D5. OnStar ATOMS cloud service (courtesy of Telematics News).

The first partner to apply OnStar's API is San Francisco-based RelayRides (RelayRides 2013), the first P2P car sharing marketplace in the world. By using RelayRides's services, GM vehicle owners will be able to rent out their idle cars by allowing owners to control rates and availability. Using the RelayRides mobile app for iPhone or Android, the owner can announce the vehicle's availability for rent within certain periods (especially if the car is a Chevrolet Volt, the car could be set to function in the specified timeframe). Once a rental time is agreed, customers can use RelayRide to find the Volt via GPS, and remotely unlock the car to access the keys. A customer can also unlock the car via a text message. Renters can search the availability of cars based on vehicle type, proximity, timeframe, price etc. Rental rates as low as $5 per hour include a million-dollar insurance policy and GM claims that vehicle owners could earn up to $250 a month.

OnStar also has plans to allow API access for developers to broaden the ONStar service potential for integrating Electronic Vehicles (EV) such as the Chevrolet Volt with Smart Grid technology. Hundreds of regional utility employees will start to use Chevy Volts every day, all as part of an effort to gather real-time data on how EVs affect the grid. The pilot will use ATOMS to remotely offer utilities information about a vehicle's overall charge level and charging history (time and location).

D2. Windows Auto based system and Ford SYNC

Windows Embedded Automotive (Microsoft 2013), previously also known as Microsoft Auto, Windows CE for Automotive, Windows Automotive, and Windows Mobile for Automotive, is an embedded operating system based on Windows CE developed jointly by the Automotive Business Unit and Clarion. The first version was released in 1998 and was based on Windows CE 2.0. Since then it has gone through several evolutive steps and the latest version, "Windows Embedded Automotive", released in autumn 2010 (Figure D6), is based on the Windows 7 platform.

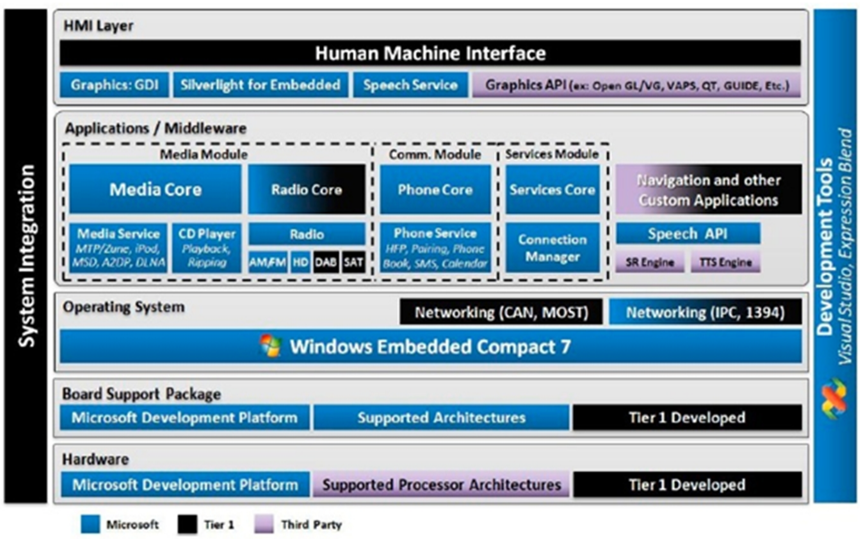

Figure D6. Windows Embedded Automotive (courtesy of Microsoft).

The Windows Automotive operating system (Windows Embedded Compact 7) is compatible with several CPU architectures including Intel multi-core IA, ARM v7 and SH4. Networking support includes de-facto industrial standards CAN (Controller Area Network) and MOST (Media Oriented System Transport) as well as IEEE 1394 (FireWire) serial bus. Also Bluetooth support is complete, including all notable protocols and profiles giving abilities for audio streaming and phonebook access. The media interface is also extensive, all significant media formats and codecs being supported, also with iPhone/iPad firmware level interfacing. The user interface is based on the Silverlight for Embedded version that gives abilities to apply natural user interfacing including voice (English, Spanish, French and Korean), text to speech and customizable gesture input, as well as usual graphical interfaces based on the Microsoft Graphics Device Interface (GDI).

The first adapter of Windows Automotive was Ford. Ford SYNC is a factory-installed, integrated in-vehicle communications and entertainment system now present in more than 2 million Ford, Lincoln, and Mercury vehicles in the USA (since 2007) that allows users e.g.:

- to make hands-free telephone calls and control music and other functions using voice commands or the vehicle's steering wheel or radio controls
- to receive text messages and read them aloud using a digitized voice
- to use USB ports and RCA inputs that can charge and (play from) an iPhone, or to plug in a USB drive and browse their music by genre, album, artist, playlist or song title. In addition to the USB inputs, the Edge has an SD card slot and RCA inputs to connect just about any video player including a camcorder, video game or video iPod.

In 2010 Ford released SYNC AppLink, a downloadable software upgrade for Ford SYNC systems that supports hands-free voice control of applications for BlackBerry and Android smartphones. Ford also launched an SDK (System Development Kit) to create a SYNC Developer Network leveraging smartphones and connectivity into vehicles with best-in-class voice and HMI interfaces. With the SDK and SYNC API's, mobile application developers are able to:

- Create voice user interfaces for applications using the in-vehicle speech recognition system
- Write information on the radio head display or in-vehicle touchscreen
- Speak text using the text-to-speech engine
- Use the in-vehicle menu system to provide commands or options for a mobile app
- Interpret button presses from the radio and steering wheel controls
- Receive vehicle data like speed, GPS location, fuel economy etc.

Applications developed using Ford Sync AppLink will presumably be comparable with GM Onstar RemoteLink applications.

Fiat's Blue&Me (Microssoft 2010) is another infotainment based on Microsoft Automotive and was developed in partnership with Magneti Marelli. Similar features are offered, like Bluetooth and USB connectivity to mobile phones and personal media players, and voice control.

D3. Future directions

As previous examples indicate, computerized systems and software are playing an increasingly important role in the automotive industry. Today, low-end cars use 30 to 50 electronic control units (ECUs) embedded in the car for monitoring and controlling functions and high-end cars like the S-class Mercedes-Benz rely on more than 20 million lines of code (Charette 2009). The complexity of code will increase substantially (estimated to be over 300 million lines) in the future as the vision of “intelligent connected” cars evolves to exploit ITS and Internet based services. All the leading manufactures have current offerings, partnerships and research agendas focused on developing the connected car as an asset to competitive advantage. As a result there will undoubtedly be multiple, non-interoperable, de-facto standards and competing “car software ecosystems” similar to the mobile phone industry today.

Currently all car manufacturers rely on commercial operating systems like Microsoft Automotive or QNX, but in the future open source systems will gain some favour among the auto industry currently committed to proprietary systems (Poliak 2011). An open source IVI platform will offer benefits like faster speed to market, reductions in production costs, and increased options for integrating and customizing solutions. Also a wider developer base will boost application provisioning. The most notable effort in this area is GENIVI, a non-profit industry alliance trying to define a globally competitive open source operating system, middleware and platform for the automotive industry.

The studies mentioned in previous chapters reflect the current trend among many car manufacturers to enable pairing phone devices with their vehicles' IVI systems. Since it seems that the mobile device will be used as a hub for information and entertainment services, a solution for this pairing will be needed, and one appropriate candidate is MirrorLink (formerly known as Terminal Mode). Portable devices like mobile phones and aftermarket satellite navigation (that also have some IVI related functions) units are appealing because they:

- are much cheaper than OEM IVI units purchased with a car
- are transportable from car to car
- can keep favourite POIs in the unit, which are updatable at home via PC as needed.

There is a serious mismatch considering the car’s embedded IVI systems and mobile devices like smartphones and future satellite navigation system development. Typically vehicle development cycles are from 4 to 14 years depending on how big the model modification is. Typically a car’s lifetime in Europe is 13 to 19 years and embedded IVI systems are not easily upgradable. Conversely, consumer electronics like home electronic systems and mobile phones have a lifecycle of only 18 months to 3 years, which requires new product releases to maintain the manufacturer’s profit levels. This requires some means either to upgrade the IVI system faster or to fit a connection bus from portable devices to the car's IVI systems. The latter approach might be more probable.

D.3.1 GENIVI

The GENIVI Alliance (Genivi 2013) was founded in 2009 by BMW, Delphi, GM, Intel, Magneti-Marelli, PSA Peugeot Citroen, Visteon, and Wind River Systems. Since then the consortium has grown to include over 160 members from the automotive manufacturing industry, automotive related device manufacturers, semiconductor manufacturers, and service and software companies. The GENIVI alliance aims to serve following main purposes (Genivi 2013):

- To provide the forum within which the required critical mass would be established. The current fragmentation means that the automotive OEM must trigger the development and the software developer has to accept that the potential product volume is limited.
- To host the technical programmes within which the open infotainment platform would be developed, maintained, and promoted and its users protected.
- To establish a compliance programme that would enable GENIVI-based products to be obtained from multiple open source and commercial sources.

The goal of the effort is to establish a globally competitive, Linux-based operating system, middleware and platform for the automotive in-vehicle infotainment (IVI) industry.

The GENIVI software architecture is utilizing existing open source software components as a base and some automotive-specific software is also added to finish the GENIVI architecture (Figure D7). GENIVI alliance has several public open source projects including (Genivi 2013):

- GENIVI Yocto – a Linux distribution for a variety of embedded devices
- GENIVI Baserock – an optimised build approach to create Linux-based appliance solutions. Baserock open source project which aims to be a great way to build appliance systems with Linux (www.baserock.org).
- Web API Vehicle – a proof of concept that demonstrates an interface to the vehicle accessible from HTML5
- SmartDeviceLink – a project which intends to standardize many in-vehicle interfaces which may exist in the automotive context.

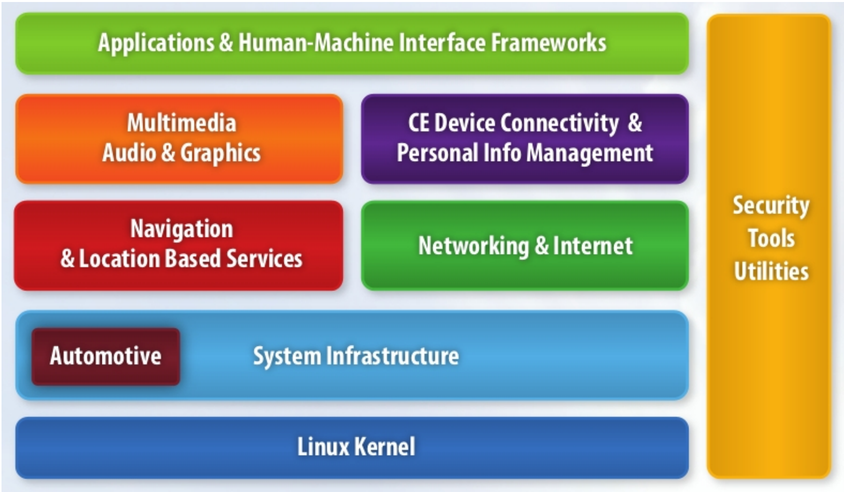

Figure D7. GENIVI architecture (Hoffman 2013).

Compared to the traditional car manufacturer lead approach the open source efforts like GENIVI have several advantages (Visteon 2013):

- Elimination the replication of non-value add R&D. Earlier diagnostics, drivers, protocol stack were developed again and again, by each supplier, for each car maker, for every embedded device
- Reduction of R&D costs and time to marke since projects start with an established and comprehensive libraries
- Significant reduction in the burden of inter-supplier communication by avoiding typical protracted negotiations and delays sharing system data
- Open source model drives cumulative improvements since each developer community adds and builds upon the previous. The enhancements, bug fixes and performance patches get re-deposited in open source library for free adoption.

Albeit several GENIVI compliant platform exists (GENIVI 2013) ,currently non-of the real world Linux based IVI system like Jaguar-Land-Rover's , GM's Cadillac and Toyota's Lexus IVIsare GENIVI compliant. According to Hoffman (Hoffman 2013), BMW will be the first car manufacturer that will introduce a complete GENIVI compliant IVI platform later this year. The demonstration platform has already been presented in July 2013 and production model should be released in GENIVI Summit in November 2013.

The importance and future of GENIVI is hard to estimate, since IVI systems are evolving very fast. For example Google's Android is also gaining momentum – Renault has introduced an IVI system called R-Link based on Android Linux. The biggest advantages of Android are application libraries and advanced GUI, already existing ecosystem with developers, applications and application stores. The drawbacks are the lack of automotive related feature like interfaces to in-car buses, responsiveness needed by automotive electronics and possible security vulnerabilities (Shah 2013).

D.3.2 MirrorLink/Terminal Mode

In 2010, Nokia started the development of an integration framework between mobile phones and IVI systems called Terminal Mode. The basic idea of the system is that the mobile device (a smartphone) becomes the application platform for the automotive environment, whereas the IVI system is responsible for user input and output (Bose et al. 2010). In this concept the mobile device executes all the applications and also acts as a communication means to network based (cloud) services. A vehicle's IVI system offers user interface hardware and physical input/output devices by providing one or more displays (touch screens, buttons and multi-functional knobs, audio playback and voice input systems depending on the vehicle's IVI build-up.

This arrangement is preferable for several reasons (Bose et al. 2010, 2011):

- Factory fitted IVI systems used to be quite expensive (a differentiation factor) and the lifecycle of the car including the IVI system is much longer that mass-market products like retrofitted navigators or smartphones. The factory-fitted IVI systems will be outdated both in terms of hardware and software properties long before the end of the car's lifecycle is reached. By using a smartphone as an execution environment, systems and software could easily be updated and upgraded when needed.
- Although efforts like GENIVI try to alleviate the problem, still most IVI systems are closed systems and do not have a proper third party developer eco-system to generate enough developer interest for enabling versions of apps that the users are accustomed to. By using a smartphone as the execution environment, all familiar applications (without any modification) can be offered at any time to provide good quality user experience – if the framework is designed as application agnostic.
- Use of the IVI system's user interface guarantees that the physical UI components such as displays and controls are optimized specifically for use by the driver and/or passengers in a vehicle, and that they comply with laws and regulation governing driver distraction and driver/passenger safety. Problems related to interaction with small screens and controls during a drive (endangering traffic safety) are solved.
- New commercial opportunities emerge. Automotive manufacturers do not need to set up their own IVI specific application stores; instead existing App Stores like Android Market, MS ApplicationStore, AppleStore etc.

can be used as distribution channels for their own IVI apps. The framework also opens up new application opportunities for smartphone developers, especially if the connection to the vehicle's telemetry information (via IVI or OBD-II etc.) is enabled.

Technically, TerminalMode (renamed MirrorLink) is based on a set of protocols and services. Both IVI systems and mobile devices should support the protocol stack shown in Figure D8.

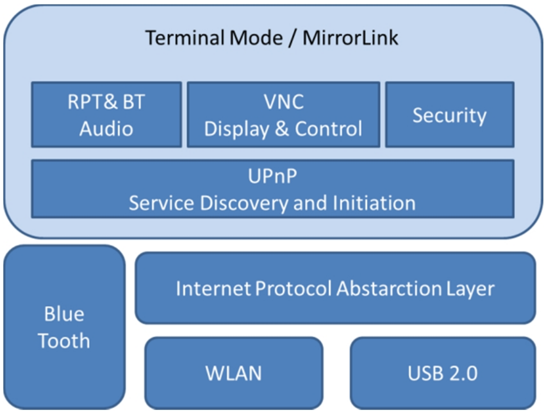

Figure D8. Terminal Mode / MirrorLink protocol stack (Bose2010).

In the physical communication layer, USB, WiFi and Bluetooth connections are used. A cable based USB 2.0 connection support with Communications Device Class/Network Control Model (CDC/NCM) is mandatory, enabling the use of multiple Ethernet packages within a single USB transfer, and also supporting energy supply/terminal charging while a device is connected. Currently, WiFi (IEEE 802.8 b/g) and Bluetooth Hands-Free Profile (HFP) and Advanced Audio Distribution Profile (A2DP), which support enabling wireless communication, are optional.

TerminalMode/MirrorLink uses TCP/IP networking on a transport layer enabling the use of wide variation of content stream protocol over a physical connection. Automatic network establishment and IP address distribution is based on DHCP (Dynamic Host Configuration Protocol) support, eliminating the need for manual configuration.

Automatic service discovery, access and configuration are handled by the Universal Plug and Play Protocol (UPnP vers.1.1). When a connection between an IVI system and a mobile device is established, the mobile device advertises MirrorLink services to the IVI system. The IVI system as an UPnP Control Point can then start invoking these discovered services offered by the Tm Application Server located in the mobile device. The IVI system is a remote user interface client enabling the use of touch screens, buttons, audio-input, video display and audio. For this purpose, Terminal Mode uses the Virtual Network Computing (VNC) protocol where the VNC server resides on the mobile device and the VNC client resides on the IVI system.

Terminal Mode's VNC is extended to handle certain features like display scaling and rotation and input events supported by a mobile device (like popping up a virtual keyboard when text input is needed). Audio input and output are supported by Bluetooth or Real-Time Protocols (RTP). Bluetooth HFP is used for receiving the voice input from the IVI system and the mobile device can stream audio to the IVI system using either HFP or the Advanced Audio Distribution Profile (A2DP). If RTP is applied, both input and output streams with 16 bit 48 KHZ stereo PCM encoding use it.

Security is handled partially by the network and Terminal Mode layer levels. Link Layer connections over WLAN use WPA2 (WiFI Protection Access 2), the USB cable connection is physical peer-to-peer and does not raise security issues. Bluetooth offers several security modes since the introduction of Secure Simple Pairing in Bluetooth v2.1. UPnP device implementations lack authentication mechanisms, and by default assume local systems and their users are completely trustworthy. The Device Attestation Protocol (DAP), based on the ISO X.509 standard, ensures that IVI systems only connect to Terminal Mode approved and compliant devices.

Development of the MirrorLink framework is led by the Car Connectivity Consortium (CCC) formed in 2011 (Car Connectivity Consortium 2013). The first car industry members were GM, Daimler, Honda, Hyundai, Toyota, and Volkswagen; participating IVI manufacturers were Alpine, Panasonic Clarion, JVC, Kenwood, Garmin and Sony, and phone manufacturers LG Electronics, Nokia and Samsung. The first functional prototype was presented jointly in 2010 by Volkswagen and Nokia.

Figure D9. Android mobile phone display on an IVI screen (courtesy of ZDNet).

In May 2012, CCC had 56 members and included almost all major automobile and mobile device manufacturers worldwide. CCC provides support for extensive evaluation of both sides of the MirrorLink™ technology. Smartphones are evaluated as

MirrorLink™ servers and IVIs as MirrorLink™ clients by independent third-party ISO 17025 accredited test laboratories authorized by CCC.

In Europe, MirrorLink compatible devices appeared to the public already in 2011, but due to liability issues its adoption on US markets was delayed. The first MirrorLink compatible devices in the USA were released by Sony and Alpine last year. Also, limited functionality and support only for certain eco-systems like Symbian and Android has hindered the spread of MirrorLink. There are also obvious commercial reasons for factory-installed IVI systems not having adopted it so far. In November 2012, CCC announced that more than 40 products are now MirrorLink™ certified, most of them being mobile devices like Nokia and Samsung (Drivelink enabled) smartphones.

In 2013 ABI Research forecasted that the user base of MirrorLink and GENIVI systems will increase from around 10,000 at the end of 2012 to 27.9 million in North America, Western Europe, and the Asia-Pacific region by the end of 2018 (ABI Research 2013).

D4. References

ABI Research 2013. Automotive Operating Systems: QNX Versus Windows and Open Source. 14 p.

Bose, R., Brakensiek, J. & Park, K. 2010. Terminal Mode – Transforming Mobile Devices into.

Bose, R., Brakensiek, J., Keun-Young Park & Lester, J. 2011. Morphing Smartphones into Automotive Application Platforms. IEEE Computer (Special Issue on Mobile Computing) Cover Feature, May 2011, pp. 53–61.

Brown, E. 2013. The rise of Linux in in-vehicle infotainment (IVI). http://linuxgizmos.com/linux-based-in-vehicle-infotainment-on-the-rise/. Accessed 16.10.2013.

Car Connectivity Consortium 2013. http://www.mirrorlink.com/. Accessed 14.10.2013.

Charette, R.N. 2009. This Car Runs on Code. IEEE Spectrum. (Posted 1 Feb 2009). http://spectrum.ieee.org/green-tech/advanced-cars/this-car-runs-on-code. Accessed 15.10.2013.

Ford 2013. Ford SYNC technology. http://www.ford.com/technology/sync/. Accessed 14.10.2013.

Genivi 2013. GENIVI Alliance. http://www.genivi.org/. Accessed 14.10.2013.

Hoffmann, J. & Wong, W. Interview: Joel Hoffmann Discusses Infotainment And The GENIVI Alliance. http://electronicdesign.com/embedded/interview-joel-hoffmann-discusses-infotainment-and-genivi-alliance. Accessed 14.10.2013

Kia 2013. Kia Motors UVO eServices. http://www.kia.com/us/uvo/#/home. Accessed 14.10.2013.

Mbrace2TM 2013. http://www.mbusa.com/mercedes/mbrace. Accessed 14.10.2013.

Microsoft 2010. A Technical Companion to Windows Embedded Automotive 7. 100 p. Accessed 14.10.2013.

Microsoft 2013. Windows Embedded Automotive. http://www.microsoft.com/windowsembedded/en-us/windows-embedded-automotive-7.aspx. Accessed 14.10.2013.

MIT 2011. MIT team wins OnStar app contest. http://web.mit.edu/newsoffice/2011/onstar-app-contest.html. Accessed 14.10.2013.

OnStar 2013. Auto security. https://www.onstar.com/web/portal/home?g=1. Accessed 14.10.2013.

Poliak, A. 2011. Emerging Trends and Business Models for the Connected Car. QNX Software Systems. 23 p.

RelayRides 2013. Rent your perfect car. https://relayrides.com/. Accessed 14.10.2013.

Shah, K. 2013. Android vs Linux in automotive apps: a role of each? http://linuxgizmos.com/android-vs-linux-in-automotive-systems/. Accessed 14.10.2013.

Stockwell, P. 2006. In-car Infotainment – Developments in information and entertainment for the driver & Passengers for the IET North Hants. 52 p.

Toyota 2013. Entune™. http://www.toyota.com/entune/. Accessed 14.10.2013.

Visteon 2013. Visteon GENIVI presentation. http://elinux.org/images/e/ec/Temmos-GeniviVisteon.pdf. Accessed 14.10.2013.

Series title and number
VTT Technology 142

Title	**Multi-Service Architecture for mobility services**
Author(s)	Olli Pihlajamaa, Immo Heino & Armi Vilkman
Abstract	The work done in the SUNTIO2 project and reported in this document aims at providing assets for realizing the Multi-Service Model for the creation, provision and supply of mobility services. It is continuation for the work done in PASTORI and SUNTIO projects in which the Multi-Service Model has been developed. The central result of the work culminates to the set of requirements for the system realizing Multi-Service Model and the functional architecture for such a system. The requirements, and hence the functional architecture, are based on several sources. First, the participants of the SUNTIO2 project formed a representative selection of potential roles in the Multi-Service Model and were, thus, provided invaluable sources of stakeholder aspirations. In addition, discussions with numerous other stakeholders during the project complemented this view. The extensive survey of the state-of-the-art as well as past and on-going efforts in the field of ITS and mobility services gave also important input to the work. Furthermore, the analysis of two service use cases helped greatly in the concretization of the requirements. The architecture for Multi-Service Facilitation (MSF) created on the base of the requirements introduces the high-level functionality required for the realization of the Multi-Service Model. It includes such central elements as: MSF Centre for server side functionality, MSF Client Framework for offering user interface for MSF Centre and building service frontends, MSF Service Framework for service providers to co-operate in service provision and B2B Marketplace to facilitate the business ecosystem.
ISBN, ISSN	ISBN 978-951-38-8077-4 (Soft back ed.) ISBN 978-951-38-8078-1 (URL: http://www.vtt.fi/publications/index.jsp) ISSN-L 2242-1211 ISSN 2242-1211 (Print) ISSN 2242-122X (Online)
Date	December 2013
Language	English
Pages	99 p. + app. 71 p.
Name of the project	
Commissioned by	
Keywords	Multi-Service Model, Intelligent Transportation Systems (ITS), mobility services, requirements, architecture, state-of-the-art
Publisher	VTT Technical Research Centre of Finland P.O. Box 1000, FI-02044 VTT, Finland, Tel. 020 722 111

Multi-Service Architecture for mobility services

VTT TECHNOLOGY 142 Multi-Service Architecture for mobility services

ISBN 978-951-38-8077-4 (Soft back ed.)
ISBN 978-951-38-8078-1 (URL: http://www.vtt.fi/publications/index.jsp)
ISSN-L 2242-1211
ISSN 2242-1211 (Print)
ISSN 2242-122X (Online)