I. Bošnjak, D. Domandžić, Lj. Šimunović: Multi-layered Model of Intelligent Transport Systems and Services Interoperability

ABSTRACT

The paper discusses the problem of Intelligent Transport Systems (ITS) and services interoperability in the EU environment. The problem is associated with multi-level models and functional system design according to the defined requirements. The first result of the research is a formalised comprehensive procedure for ITS functional system formulation. The national ITS architecture development is suggested as a necessary systematic framework for the deployment of dozens of new ITS applications in CSEE countries.

KEY WORDS

intelligent transport systems, functional design, models

1. INTRODUCTION

The development of Road Transport Telematics in Central and South-eastern European Countries (CSEE) in the first decade of 21st century will evolve from partial traffic signal automation to complex of Intelligent Transport Systems (ITS) with more than 30 user services and applications. Effective deployment of Intelligent Transport Systems and services requires proper beyond-technological harmonization of ITS requirements as a starting point to technical and technological interoperability.

As are very complex orgware/bioware/hardware/software systems, ITS cannot be designed, planned, built and exploited without proper systems layered model [2], [6]. There is not one “right” view of ITS complex, therefore systems engineers must develop proper models for defined problem context and purpose.

This paper is organized as follows. First the hierarchical system design and different aspects of interoperability are discussed. Then in the following section, the functional system design procedure which includes interoperability requirements is formally defined. Section 4 highlights the decomposition of ITS into functional areas and functions according to the EU projects. Finally, in Section 5 the concluding remarks are given and directions for future works suggested.

2. DIFFERENT ASPECTS OF INTEROPERABILITY

Several models for transport telematics development and deployment are oriented primarily to one aspect of interoperability, such as information exchange, physical/electrical interfaces, etc. ITS architecture development is good opportunity for system harmonization and consensus building among ITS users, service operators, road network operators, public agencies and other stakeholders. System engineers have a role to state the problem comprehensively and to apply the design procedures in which different disciplinary and specialist knowledge can be integrated.

In general, we can identify four main (different) aspects of ITS interoperability:

- technical interoperability,
- functional/logical interoperability,
- contractual/institutional interoperability,
- policy measures for interoperability.

For example, the European smartcard users want to use their cards in every town at the national and trans-national level. In this case technical interoperability requirements include:

- security mechanism between reader and smart-card,
- use of ISO 14443 standard (contactless transmission),
- use of ENV 1545 standard (data specification), etc.
Functional/logical interoperability includes:
- common functional reference system,
- data exchange formats, etc.

Contractual/institutional interoperability includes:
- organisation of sales and customer relations,
- sharing of revenues, etc.

Policy measures for interoperability include:
- legal appraisal for procedures of approval guaranteeing interoperability,
- encouraging the combined use of various types of transport and other services, etc.

3. FUNCTIONAL SYSTEM DESIGN WITH INTEROPERABILITY REQUIREMENTS

The interoperability problem ought to be stated in a comprehensive, complete and consistent manner. Such a problem can be achieved by systems engineers working within the interdisciplinary team looking at all aspects of the interoperability problem. ITS functional analysis has to resolve iteratively the “top-level” system design into simpler system design problems with the objective of defining consistently simpler (low-level) system functions that can be performed by physical components: people (bioware), machines (hardware), software.

According to the systems engineering methodology ([1], [7]) ITS functional analysis is a three-step iterative process:

FA-Step1. Decompose the top-level requirements into a set of functional requirements each defined in the same format and consistent with the top-level (interoperability) requirement. This decomposition has to be based on guidance from physical synthesis (guidance is in the form of the system design concept).

FA-Step2. Decompose requirements in all other categories of the top-level system design and allocate them to the functional (interoperability) requirements defined in Step 1. This step result in a resolution of the top-level design problem into interrelated low-level system design problems.

FA-Step3. Validate that the system design problem resolution is consistent with the top-level system design problem according to the defined criteria and additional consistency condition.

Simple (low-level) functions, when properly organized and integrated in buildable system, will achieve the requirements and performance of the top-level system function.

An ITS design problem (SDR) with explicitly formulated interoperability requirement can be defined as follows:

$$\text{SDR} = (\text{USR}, \text{OPR}, \text{FCN}, \text{THR}, \text{PER}, \text{COR}, \text{TOR}, \text{STR})_{p,s,t}$$

where:
- \(\text{USR}\) denotes user needs and other stakeholders requirements of ITS,
- \(\text{OPR}\) denotes interoperability requirements,
- \(\text{FCN}\) denotes ITS functions,
- \(\text{THR}\) denotes technology,
- \(\text{PER}\) denotes performance,
- \(\text{COR}\) denotes cost,
- \(\text{TOR}\) denotes time,
- \(\text{STR}\) denotes space.

The space of all functional and buildable design with interoperability requirements can be represented as:

$$\text{FSD} = (Z, DSZ, TSZ, SSZ)$$

Figure 1 - Illustration of functional design problem with interoperability requirements
PER denotes performance requirements, 
COR denotes the cost requirements, 
TOR denotes the trade-off requirements, 
STR denotes the system test and integration requirement,
p, s, t denotes population, space and time as back drop variables.

The space of functional system design (with interoperability requirements) of ITS is denoted as 
FS(ITS) and defined as follows:
FS(ITS) = {FSD : FSD = (Z, DSZ, TSZ, SSZ)}
Z is “example” system that satisfies requirements with respect to initial state of Z and time & space subscales (TSZ & SSZ) 
Each element FSD_i ∈ FS(ITS) is called the functional system design. It can be said that Z is in the functionality space.

The functional system design problem with interoperability requirements is illustrated in Figure 1.
After conceptual development and system function specification, the functional system design must be implementable by buildable system designs. It has a mode of behavior that exhibits the functionality of the functional system design. The components of the buildable system are hardware, software and bioware components in the specified technology.

4. DECOMPOSITION OF ITS INTO FUNCTIONAL AREAS AND FUNCTIONS

Planning, designing and integrating Intelligent Transport Systems in the EU environment, from the initial control signalised road intersections, traffic and transport telematics evolve to sophisticated traffic management and information system covering whole towns or large areas. It is not only one system, but several ITS are developed with functionalities for defined user needs. Common European framework for planning is defined in Keystone Architecture Required for European Network (→ KAREN).

User needs defined in the KAREN project represent what both internal and external (EU) users want the System to do both now and over the next ten years. Every country has to define the country-specific ITS requirements using common (KAREN) framework and ISO specifications of 32 fundamental user services. The defined user needs and interfaces that systems need are the starting point to design Functional ITS architecture and Physical ITS architecture. The next step is to define communications requirements between “terminators” (travelers, centres, vehicles, roadside) in terms of standards and required media (→ Communications Architecture).

ITS architecture must cover multi-system and multi-organisational requirements and therefore multi-level approach is necessary. CONVERGE multilevel model for analysis of ITS architecture defines four levels:
- multi-agency interoperability properties (“level 3”),
- single agency system properties (“level 2”),
- system structure (“level 1”),
- subsystem/component structure (“level 0”).

KAREN does not provide component and system design specification, but can be useful as the starting point in the systems engineering design.
ITS development in the CSEE country requires subsequently modification and adaptation of user needs including consideration of specific functional and non-functional requirements. The European functional architecture of ITS at the highest level identify eight functional areas (A1 . . . A8) in the order different from ISO specifications [5]. These are:
A1 Provide Electronic Payment Facilities
A2 Provide Safety and Emergency Facilities
A3 Manage Traffic
A4 Manage Public Transport Operations
A5 Provide Advanced Driver Assistance Systems
A6 Provide Traveler Journey Assistance
A7 Provide Support for Low Enforcement
A8 Manage Freight and Fleet Operations.

All the Functional Areas themselves are broken down into sets of High-level Functions (Fx) and Data Stores, with Data Flows between them. The interface between the Functional Areas (and between functions) is provided through data that flow between them.
High-level functions are divided into Low-level functions. Both types of functions are numbered and take the first digit from the Functional Area in which they reside. There will be no duplication of High-level Function, but it is possible for Low Level Function to be duplicated. Systems decomposition is illustrated in Table 1 and Figure 2 for functional area A3 - Manage Traffic and high-level function F.33 Manage Demand.

Data Flow (DFD) shows how the functionalities are divided and how functions are linked to each other and to the “terminators” through Data Flows. All of the Data Flows have names composed by the defined rules. “Trigger” Data Flow is used for control, i.e. to effectively “start” a function, enable a function to continue, etc.
Example of High-level Function: “Manage Demand” with low-level function and data flow interfaces is given in Fig. 3. [5].
### Table 1 - Decomposition of Functional area

<table>
<thead>
<tr>
<th>A3 - MANAGE TRAFFIC</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>F.3.1.</td>
<td>Provide Traffic Management (Control)</td>
</tr>
<tr>
<td>F.3.1.1.</td>
<td>Provide Urban Traffic Management</td>
</tr>
<tr>
<td>F.3.1.2.</td>
<td>Provide Inter-urban Traffic Management</td>
</tr>
<tr>
<td>F.3.1.3.</td>
<td>Provide Traffic Management for Bridges and Tunnels</td>
</tr>
<tr>
<td>F.3.2.</td>
<td>Manage Incidents</td>
</tr>
<tr>
<td>F.3.2.1.</td>
<td>Detect incidents</td>
</tr>
<tr>
<td>F.3.2.2.</td>
<td>Identify and Classify Incidents</td>
</tr>
<tr>
<td>F.3.2.3.</td>
<td>Assess Incidents and Determine Responses</td>
</tr>
<tr>
<td>F.3.2.4.</td>
<td>Manage Incident Data</td>
</tr>
<tr>
<td>F.3.2.5.</td>
<td>Provide Incident Management Operator Interface</td>
</tr>
<tr>
<td>F.3.3.</td>
<td>Manage Demand</td>
</tr>
<tr>
<td>F.3.3.1.</td>
<td>Receive Information on Travel Factors</td>
</tr>
<tr>
<td>F.3.3.2.</td>
<td>Develop Demand Management Strategy</td>
</tr>
<tr>
<td>F.3.3.3.</td>
<td>Implement Demand Management Strategy</td>
</tr>
<tr>
<td>F.3.3.4.</td>
<td>Manage Demand Data Store</td>
</tr>
<tr>
<td>F.3.3.5.</td>
<td>Provide Demand Management Operator Interface</td>
</tr>
<tr>
<td>F.3.4.</td>
<td>Provide Environmental Information</td>
</tr>
<tr>
<td>F.3.4.1.</td>
<td>Monitor Weather Conditions</td>
</tr>
<tr>
<td>F.3.4.2.</td>
<td>Monitor Atmospheric Pollution</td>
</tr>
<tr>
<td>F.3.4.3.</td>
<td>Monitor Noise Pollution</td>
</tr>
<tr>
<td>F.3.4.4.</td>
<td>Predict Environmental Conditions</td>
</tr>
<tr>
<td>F.3.4.5.</td>
<td>Provide Environmental Conditions Operator Interface</td>
</tr>
<tr>
<td>F.3.4.6.</td>
<td>Manage Environmental Conditions Data</td>
</tr>
<tr>
<td>F.3.5.</td>
<td>Manage Maintenance</td>
</tr>
<tr>
<td>F.3.5.1.</td>
<td>Evaluate Short Term Maintenance Needs</td>
</tr>
<tr>
<td>F.3.5.2.</td>
<td>Evaluate Long Term Maintenance Needs</td>
</tr>
<tr>
<td>F.3.5.3.</td>
<td>Evaluate Equipment Maintenance Needs</td>
</tr>
<tr>
<td>F.3.5.4.</td>
<td>Evaluate De-icing Need</td>
</tr>
<tr>
<td>F.3.5.5.</td>
<td>Provide Operator Maintenance Operations Interface</td>
</tr>
<tr>
<td>F.3.5.6.</td>
<td>Manage Maintenance Data Store</td>
</tr>
</tbody>
</table>
Figure 21 DFD 3.3 Manage Demand

3.3.1 Receive Information on Travel Factors

3.3.4 Manage Demand Data Store

3.3.5 Provide Demand Management Operator Interface

3.3.3 Develop Demand Management Strategy

3.3.2 Implement Demand Management Strategy

Figure 2 - DFD 3.3 Manage Demand
5. CONCLUSION

This paper considers the problem of Intelligent Transport Systems and services operability using systems engineering methodology. The first result of our research is a formalised procedure for functional system design with interoperability requirements. The existing functional architecture defined in KAREN and the European national ITS architectures are useful for CSEE countries but these projects do not provide full functional system and component design. The review of EU ITS Functional area with decomposition in high-level and low-level functions, data stores and data flows can be a good background for new ITS functional system design in concrete environment. Work on ITS National Architecture development is crucial for effective deployment of ITS in CSEE countries.

SAŽETAK

MODEL VIŠERAZINSKE INTEROPERABILNOSTI INTELIGENTNIH TRANSPORTNIH SUSTAVA I USLUGA

U radu je razmotren problem interoperabilnosti Intelligentnih transportnih sustava i sutuga u EU okruženju. Problem je asociiran s višerazinskim modelima i funkcionalnim dizajnom sustava prema definiranim zahtjevima. Prvi rezultat istraživanja je formalizirana jezgrovita procedura za funkcionalni sustavski dizajn ITS-a sa eksplicitnom formulacijom zahtjeva interoperabilnosti. Sugeriran je razvoj nacionalnih ITS arhitekture kao nužne sustavске podloge za provođenje desetaka novih ITS aplikacija u zemljama CSEE.

LITERATURE