

# Future Internet Technology for the Future of Transport and Logistics

Rod Franklin<sup>1</sup>, Andreas Metzger<sup>2</sup>, Michael Stollberg<sup>3</sup>, Yagil Engel<sup>4</sup>, Kay Fjørtoft<sup>5</sup>,  
René Fleischhauer<sup>3</sup>, Clarissa Marquezan<sup>2</sup>, Lone Sletbakk Ramstad<sup>5</sup>

<sup>1</sup>Kühne + Nagel Management AG, Schindellegi, Switzerland &  
Kühne Logistics University, Hamburg, Germany  
rod.franklin@kuehne-nagel.com

<sup>2</sup>Paluno (The Ruhr Institute for Software Technology) University of Duisburg-Essen, Essen, Germany  
{andreas.metzger, clarissa.marquezan@paluno.uni-due.de}

<sup>3</sup>SAP Research, Dresden, Germany  
{michael.stollberg, rene.fleischhauer@sap.com}@kuehne-nagel.com

<sup>4</sup>IBM Research Labs, Haifa, Israel  
yagile@il.ibm.com

<sup>5</sup>MARINTEK (Norwegian Marine Technology Research Institute), Trondheim, Norway  
{Kay.Fjortoft, LoneSletbakk.Ramstad}@marintek.sintef.no

**Abstract.** International freight transport is the foundation of global trade, representing a large and growing industry where various stakeholders collaborate to transport goods around the world. The ICT infrastructures currently employed throughout logistics business networks are limited and the use of manual systems is common. This drastically hampers the operational efficiency of logistic service providers, carriers, and the various other stakeholders involved in transport processes. This paper presents an initial conceptual architecture for an ICT platform to overcome these deficiencies. The architecture is built on top of Future Internet technologies that provide generic capabilities for the efficient and effective development of cloud-based applications based on the Internet of Services, Internet of Things, and Internet of Contents with integrated security and privacy mechanisms.

**Keywords:** Future Internet, Transport, Logistics, Conceptual Architecture, Business Collaboration, E-Contracting, Event Handling, Transport Planning

## 1 Introduction and Motivation

The efficient operation of international transport and logistics networks is a critical success factor for sustainable growth in global trade. Such inefficient operation creates barriers to trade by causing shipment delays and raising trading costs. Since transport and logistics activities account for 10% to 20% of a country's Gross Domestic Product, increases in the efficiency of these activities can dramatically improve a country's competitiveness. In addition, environmental impacts resulting from the operation of transport and logistics activities are significant, so any improvement in

efficiency within a logistics network positively contributes to sustainability objectives.

While the transport and logistics industry has made great strides in attempting to improve its efficiency, limitations in technology, transport infrastructure and regulatory regime incompatibilities have created significant barriers to future improvements. Overcoming these barriers requires new information and communications technologies that allow organizations to rapidly assemble collaborative logistics networks that can efficiently and effectively execute international trading activities. The Future Internet, with its promise of ubiquitous operation and information access, provides a potential platform for overcoming the limitations of current ICTs.

Building on the proposed capabilities of the Future Internet being developed under the European Union's Future Internet Public Private Partnership program (FI PPP), the FInest Use Case project ([www.finest-ppp.eu](http://www.finest-ppp.eu)) is designing a collaboration and integration platform for the transport and logistics industry. The *FInest platform* leverages generic capabilities, so called generic enablers, provided by the Future Internet and implements a domain-specific configurable set of services for the transport and logistics domain.

This paper presents the initial conceptual architecture of the FInest platform. In Section 2 the high level architecture of the platform is introduced. Section 3 refines this architecture by describing the initial set of services which are being implemented for the transport and logistics domain.

## 2 FInest High Level Architecture

The FInest platform consists of three layers shown in Figure 1. The layers and modules that reside within these layers are interconnected by using service-oriented technology. Service-oriented technology facilitates interoperability, openness and extensibility through standard interfaces. The use of integrated security and privacy management mechanisms ensures the secure and reliable exchange of confidential and business-critical information.

In the following we outline the three layers of the FInest architecture and discuss how we envision to exploit Future Internet technologies for the implementation of the platform.

### 2.1 Front End Layer

The front end layer of the FInest platform provides users with role specific, secure, ubiquitous access from different devices to information concerning the operation of the transport and logistics network. Roles that are supported through this front end layer include, but are not limited to, the following:

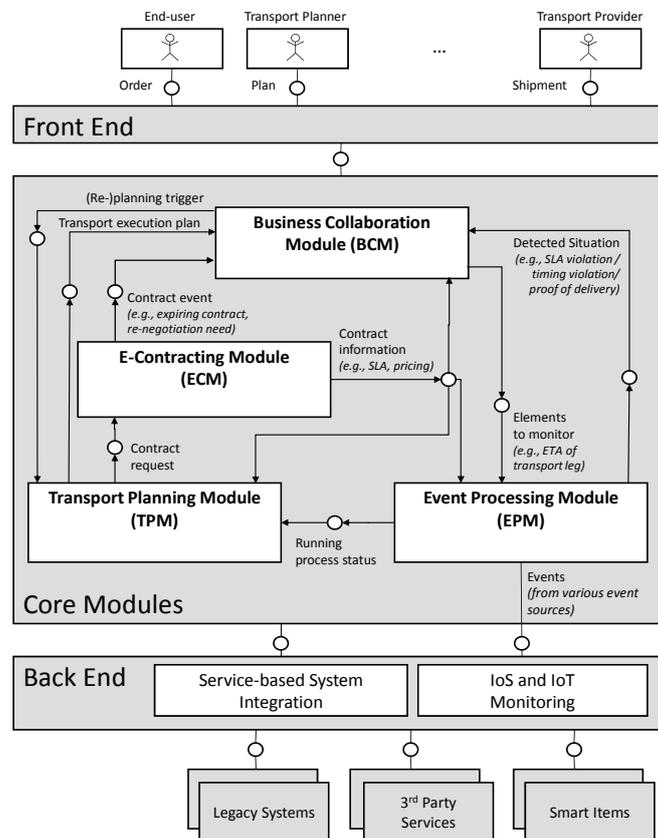
- end-users (customers) – can issue or trace their orders.
- transport planners – can develop, monitor and update transport plans based on end-user/customer orders. Transport plans may include the individual legs of a

shipment, the mode used for that leg of the shipment and the provider of the transport service for that leg of the shipment.

- logistics service providers – can provide offers for transport and logistics services and retrieve demands for these services via dedicated marketplaces.

## 2.2 Back End Layer

The back end layer of the FInest platform provides access to, and integration with, legacy systems, third party services and any Internet of Things (IoT) devices that may provide information during the transport lifecycle. Legacy system integration is facilitated by service-oriented technology, e.g., by exposing features of legacy systems as services, or by offering access to legacy systems via the “Software as a Service” [1, 2] delivery model.



**Figure 1.** FInest platform high level conceptual architecture<sup>1</sup>

<sup>1</sup> The notation used throughout this document is TAM (Technical Architecture Modeling), see <http://www.fmc-modeling.org/fmc-and-tam>

### 2.3 Core Modules Layer

The core layer of the FInest platform is composed of independent transport and logistics service modules integrated through the Business Collaboration Module. The independent service modules are “cloud-based” applications that provide essential domain services for the shipment of goods. The initial set of core modules that are being developed for the FInest platform include the following:

- **Business Collaboration Module (BCM)** – the central module of the FInest platform that supports the inter-organizational collaboration between transport and logistics network partners and acts as an integration service between these partners and the various cloud based components selected to manage the efficient flow of goods between the partners.
- **E-Contracting Module (ECM)** – this module provides computer support for service provider selection, contract negotiation and agreement, contract management and the provision of contract related service requirements to other modules that utilize this information for ensuring the effective and efficient network operation.
- **Event Processing Module (EPM)** – this module provides end-to-end visibility of shipments through event-driven monitoring across domains and transportation modality. The module is also responsible for SLA monitoring (based on data from the ECM) and triggers transport re-planning when needed.
- **Transport Planning Module (TPM)** – this module provides support for dynamic transport planning and re-planning activities, exploiting real time event data provided through the EPM and with respect to contracts between business partners that are managed within the ECM component. Re-planning of shipments occurs when real-time signals from the EPM indicate that a current transport plan cannot be achieved because of some event that has arisen in the shipment process.

### 2.4 Usage of Future Internet Technologies

The FInest Platform will be developed using Future Internet technologies that provide a foundation for cost-efficient and rapid development of cloud-based end-user applications based on emerging techniques for the Internet of Services, Internet of Things, and Internet of Content. The following list describes the Future Internet technologies of primary importance for implementing the FInest Platform. Most of these are addressed within the FI-WARE project, which develops the Future Internet Core Platform within the FI PPP program ([www.fi-ware.eu](http://www.fi-ware.eu)):

- **Infrastructure, methodology, and tools for cloud-based platform and application development**, including an infrastructure for deploying the FInest Platform and its components on public or private clouds along with methodology and tool support for developing additional end-user services for individual transport and logistics stakeholders;
- **Language and tool support for the Internet of Services**, including a service description language that covers both technical and business requirements

along with integrated tool support for the provisioning, management, and consumption of services; this shall be used for realizing the back-end layer of the FInest platform and for managing the interaction of the FInest core modules;

- **Access to real-world data from the Internet of Things**, enabling the integration and technical handling of real-world data obtained from sensor networks for real-time monitoring and tracking during the execution phase of transport and logistics processes;
- **Facilities for data and event processing in the Internet of Contents**, allowing to process huge amounts of data to retrieve insights into relevant scenarios, as well to analyse real-time event data to quickly determine relevant situations and instantly trigger actions.
- **An integrated framework for security and privacy management for the Future Internet**, including identity management, authentication and authorization facilities, non-repudiation services and policy management for user profiles as the basis for ensuring the security, privacy and confidentiality of information exchanged between business partners, which is a pre-requisite for employing the FInest Platform in real-world business environments.

### 3 FInest Platform Core Modules

The details for each of the FInest platform core modules, as introduced above, are further elaborated in the following sections. The focus in these descriptions is on the conceptual architecture and the main capabilities of the core modules; the technical realization and actual usage of Future Internet technologies is subject to future work.

#### 3.1 FInest Business Collaboration Module (BCM)

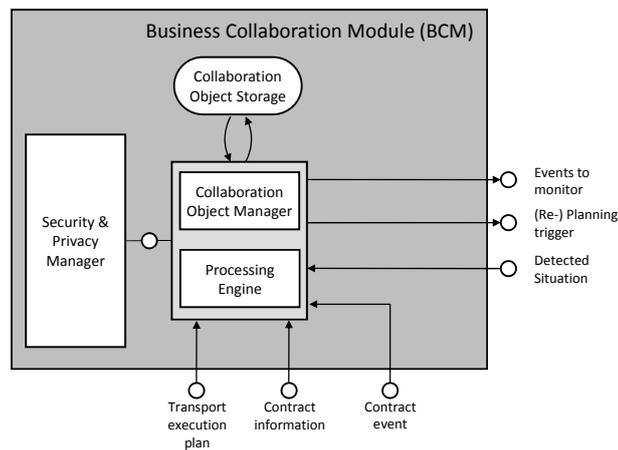
Logistics processes are distributed and involve numerous different stakeholders. Stakeholders may include customers (such as the consignor and the consignee), one or more transport planners, a set of actual transport providers (carriers or shippers), insurance companies and other legal parties, governance authorities (e.g., customs or border control) as well as other partners. Each one of these parties needs information about the goods being shipped in order to successfully conduct a transport process. However, currently employed ICT systems have been developed for intra-organizational management and do not provide easy access for external partners. Coordination between the different stakeholders, therefore, requires manual intervention in order to share information. The large amount of manual intervention required hampers effective supply chain management and increases the likelihood of errors as well as shipping costs.

The FInest Business Collaboration Module (BCM) provides transport and logistics network partners with the ability to securely manage their end-to-end networks by integrating component based services for e-contracting, planning and event monitoring. The BCM is the central module of the FInest platform for interaction with business partners. It holds all necessary information about the logistics processes and

provides specialized user interfaces so that necessary information can be presented while confidential information remains undisclosed.

To enable this, the BCM uses so-called Collaboration Objects (CO) which implement a data-centric modelling approach [3, 4]. Each CO encapsulates information about a certain aspect of the overall transport and logistics chain (e.g., a certain transportation leg or an involved carrier) and the process fragment associated with this aspect. Hence, a CO consists of two different elements: a data element and a process or lifecycle element. The combination of different COs describes the end-to-end transportation process and establishes a global view of the entire process. In addition, the distribution of information about the various aspects of the transport process over multiple COs enables privacy management due to the fact that only the information that is contained in the particular process aspect which a stakeholder is authorized to see is actually presented to this very stakeholder.

The general functionality of the BCM can be described as follows and is depicted in the conceptual architecture shown in Figure 2 below:



**Figure 2.** Business Collaboration Module (BCM) conceptual architecture

- Create a representation of the end-to-end transport and logistics process – To create a representation of the end-to-end process the BCM initializes the COs by integrating data from existing business (legacy) systems, transport plans from the transport planning system and end-user inputs.
- Store the CO-based process representation – The BCM stores the CO-based representation of the end-to-end process in a shared database.
- Provide secure access to process data – Business partners have access to the shared database, but only to those stored COs that their user rights allow. The BCM uses the scattering of information among the COs to ensure security and only discloses the necessary objects to the client.
- Provide value added services – The BCM provides stakeholders with access to other FInest modules enhancing their ability to manage their network. Access to information about the current status of a logistic process is provided by the EPM, which can be integrated to automatically update the status of the logistic

process. Planning and re-planning information for the logistic process can be obtained through the integration of the TPM by the BCM. Information about contract details can be obtained through the integration of the ECM.

### **3.2 Finest E-Contracting Module (ECM)**

Contracting within the transport and logistics domain for complex international movement of goods is currently a manual and time consuming process. The process begins with the identification of a need to ship something. Needs identification is followed by partner identification and qualification, partner bid development and bidding, bid evaluation and tentative partner selection. Once a tentative partner has been selected a contract is negotiated and agreed between various members of the contracting parties. The contract specifies all legal terms and conditions for the carriage of goods and SLA conditions such as: escalation processes for those occasions when problems arise, payment schedules and service level requirements. Unfortunately, all of this information is contained in a paper based document that is not generally available to the downstream individuals who are responsible for executing the contract.

The Finest E-Contracting Module (ECM) is being designed to address the highly manual nature of transport and logistics contracting and the problem of downstream transparency to contracted SLA conditions by exploiting solutions from e-contracting [7, 8]. It is important to remark that the legal terms and conditions of a contract are not in the focus of the ECM. The e-contracting module is envisioned as providing support for:

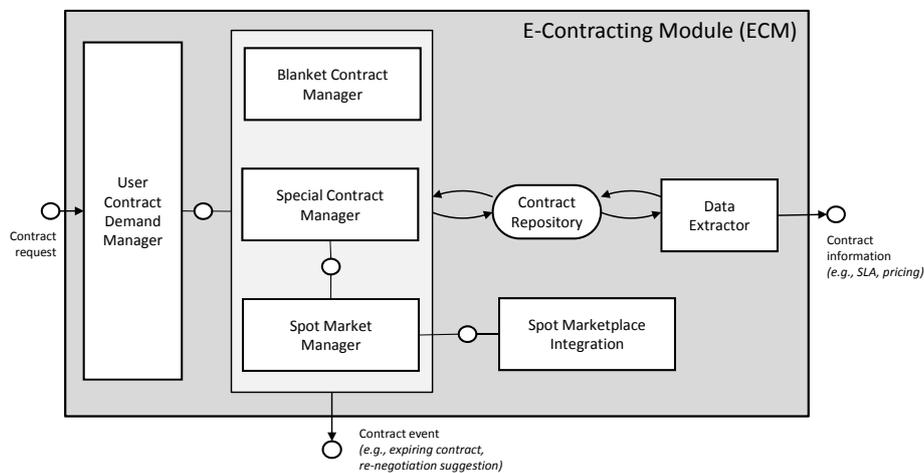
- A dynamic marketplace to support partner selection, bidding and spot market requirements;
- Semi-automated execution of contract negotiation, establishment and management;
- Electronic distribution of contract-specified execution information (e.g., SLAs, pricing, escalation processes, etc.);
- On-line management and review of contracts with automatic notification of contract end dates and renegotiation time fences.

The key architectural elements that are planned for the ECM include:

- Contract Repository – data repository for all established transport and logistics contracts, including a set of contract primitives (such as general attributes that characterize transport and logistics contracts);
- User Contract Demand Manager – single interface where actors (human or electronic) interact with the ECM and inform it of the type of contract (e.g., blanket, spot, etc.) to be negotiated;
- Blanket Contract Manager – service responsible for assembling the electronic form of a blanket contract, selection of partners (via auctioning or other processes) and contract creation;

- Spot Market Contract Manager – service responsible for selecting qualified bidding partners for a spot contract, developing and establishing the spot contract;
- Spot Marketplace – space for executing an auction process (offering, bidding, selecting, etc.);
- Special Contract Manager – service responsible for handling user requests associated with SLA violations or requests to handle special issues not covered under blanket contracts;
- Data Extractor – service responsible for generating information about/from contracts for other external modules.

A high level architecture of the ECM is shown in Figure 3 below.



**Figure 3.** E-Contracting Module (ECM) conceptual architecture

### 3.3 Finest Event Processing Module (EPM)

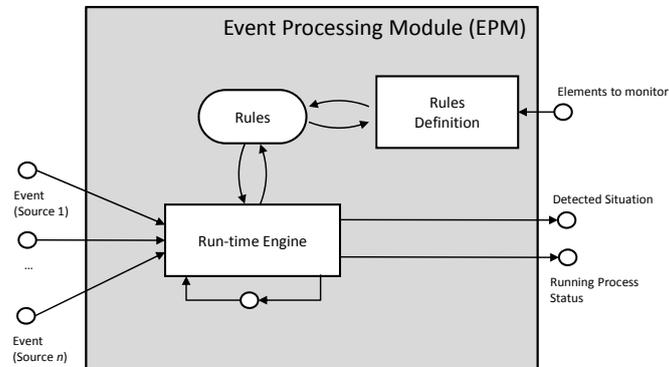
Event-driven architectures support applications that are reactive in nature, ones in which processing is triggered in response to events, contrary to traditional responsive applications, in which processing is triggered in response to an explicit request [5]. In Finest, an event-driven architecture is employed for the purpose of end-to-end monitoring of a logistics process and to facilitate immediate and proactive response to problems and potential deviations occurring during execution time. The functionality can be described at three levels.

- On the surface, event processing provides visibility into the current status of the logistics process: the location of a shipment, whether it is on a carrier or in a warehouse, whether or not it was customs-cleared, etc.

- Beyond the functionality of mere track-and-notify, event-processing employs rules that encapsulate specific logic applied to events. The basic functionality of rules exists in indicating whether or not the logistics process progresses as it should, or whether something has gone wrong.
- At a deeper level, events potentially provide insights regarding parts of the scenario that have not yet been reached; for example, stormy weather near a seaport may indicate that a ship carrying the managed containers will be delayed in entering the port. Security alerts at an airport may imply flight delays. Detecting those events relevant to the scenario at an early stage allows the system to respond to events *before* they occur, and thus to surface *proactive event-driven computing* functionality [6].

The Event Processing Module (EPM) can be characterized according to four elements, which are identified in Figure 4:

- Event Sources – FInest differentiates between two types of event sources. The first type refers to various existing (but usually incompatible) systems. These include airport systems, sea freight systems employed by ports and vessels, scanning and tracking systems of packages, and others. The second type refers to sources that will be provided by the Future Internet infrastructure and will allow more accurate monitoring, as well as predictive capabilities; these include, for instance, RFID tags, smart cameras on roads and other smart items.
- Events – FInest also distinguishes between two types of events: the events emitted by existing sources and events emitted from Internet of Things artefacts. While the latter must be defined and characterized in order to generate requirements from the Future Internet, existing events are described by domain sources such as *Cargo 2000*, which is an airline industry standard.
- Run-time Engine – The run-time engine exploits a set of rules to determine situations. Rules can either be permanent (for any scenario) or instantiated for a specific scenario, e.g., according to SLA parameters provided by the ECM or according to information about the execution of the transport plan (e.g., ETA for individual transport legs).
- Determined Situations – In FInest, the results of event processing can be directed either to the human interface, and / or to one of the other technical components in order to trigger actions. The TPM should get information regarding future events that might trigger re-planning. The ECM and BCM should be notified of potential breaches of SLAs.



**Figure 4.** Event Processing Module (EPM) conceptual architecture

### 3.4 FInest Transport Planning Module (TPM)

Efficient and effective transport planning and re-planning is all about making sure that relevant information is available at the right time and place to support supply chain operations. Planning consists of resource and information requirements which are tightly linked to resource status, availability, and configurability. Resources can be found both inside a logistics service provider (LSP) and outside an LSP, e.g., in ports or customs agencies. The ability to incorporate or “wrap-in” this information into the planning process is essential. Today this is mainly done through manual inter-organizational collaboration processes [9, 10].

The FInest Transport Planning Module (TPM) will make real-time information about resource status available across actors and organizational boundaries, which constitutes a significant improvement in planning and optimization processes [11, 12]. The TPM considers all elements that are part of an end-to-end supply chain planning process, structured according to the following four stages:

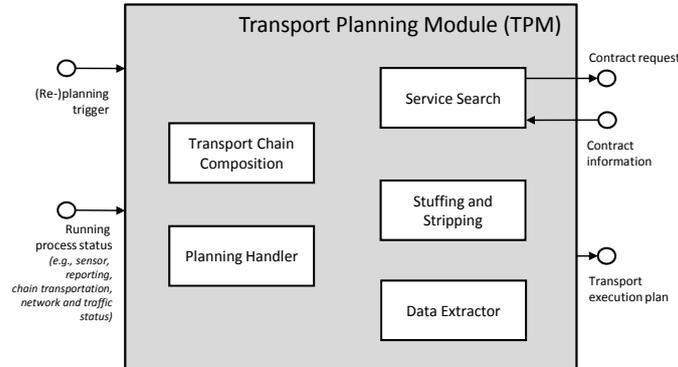
- Stage 1: Marketing of services – Relevant information includes: service category and type (vehicle services, terminal services, sea services, etc.); operation areas (location or district that the service should cover); environmental profile of the service; service capacities such as weight restrictions, dangerous goods limitations, availability; price information.
- Stage 2: Planning of a shipment – Relevant information includes: selection and negotiation of transport services to be included; pre-booking and booking of services; contracting with the service providers and with customs; reservation of space on the transport mean(s); definition of transport items (goods to be transported); split and joint booking activities; stuffing and stripping activities.
- Stage 3: Execution – Relevant information includes: status information and deviation reporting; information reporting to authorities.
- Stage 4: Completion – Relevant information includes: proof of delivery; invoicing; claims and deviation management; contract evaluation.

The TPM is being developed in order to support the planning processes regarding all these information aspects while being supported by the other modules of the FInest platform. For instance, the ECM will provide capabilities for contracting with service providers, as well as selection and negotiation of transport services. The EPM, as a further example, allows monitoring of progress and obtaining status information and deviations from the plan.

The TPM architecture includes several service components as shown in Figure 5 below:

- Service search – This component supports searching for available services by interacting with the ECM. If no contracts are available, additional contracts (e.g., spot market contracts) can be requested.
- Transport chain composition – This component is used to create an end-to-end transport plan in which many services are included.
- Stuffing and stripping – This component assists in planning the stowage of goods, e.g., in a container on a ship.
- Planning handler and data extractor – These components provide statuses and can be used to store information based on the planned transport.

Based on the information processed through the TPM, a Transport Execution Plan is created that provides a complete overview of the planned transport for both the items to be transported and the services to be used that together make up the transport plan [13,14].



**Figure 5.** Transport Planning Module (TPM) conceptual architecture

## 4 Conclusion

This paper has introduced the FInest platform, a Future Internet based services platform for the transport and logistics domain that addresses many problems that currently exist with ICTs employed in this domain. The FInest platform integrates various value added “cloud-based” components through a Business Collaboration Manager. Cloud-based service components for contracting, visibility and pro-active event

management, and planning are integrated to provide transport and logistics domain actors with access to real-time statuses on shipments. The platform facilitates the dynamic re-planning and contracting of services for domain participants providing a clear step forward in capabilities over existing technologies and processes. The FInest services are realized by leveraging Future Internet services based on the Internet of Things, Internet of Services and Internet of Content currently under development through the European Union's Future Internet Public Private Partnership (FI PPP) program. Refinement of the FInest platform will occur as the platform services move beyond the conceptual stage described in this paper and are deployed in live use case pilots.

**Acknowledgements:** The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement 285598 (FInest).

## References

1. Turner, M., Budgen, D., Brereton, P. Turning Software into a Service, *Computer*, 36(10), 2003.
2. Mietzner, R., Metzger, A., Leymann, F., Pohl, K. Variability modeling to support customization and deployment of multi-tenant-aware Software as a Service applications, In: *Proceedings of the ICSE 2009 Workshop on Principles of Engineering Service Oriented Systems (PESOS)*, 2009.
3. Meijler T.D. et al. Coordinating Variable Collaboration Processes in Logistics, In: *13th International Conference on Modern Information Technology in the Innovation Processes of Industrial Enterprises*, Trondheim, Norway, 2011.
4. Hull, R. et al. Introducing the Guard-Stage-Milestone Approach for Specifying Business Entity Lifecycles, In: *7th International Workshop WS-FM 2010*, Hoboken, USA, 2010.
5. Etzion O., Niblett, P. *Event Processing in Action*, Manning, 2010.
6. Engel, Y., Etzion, O. Towards Proactive Event-Driven Computing, In: *5th ACM International Conference on Distributed Event-Based Systems (DEBS)*, 2011
7. Papazoglou, M., Pohl, K., Parkin, M., Metzger, A. (Eds.) *Service Research Challenges and Solutions for the Future Internet: S-Cube – Towards Mechanisms and Methods for Engineering, Managing, and Adapting Service-Based Systems*. Springer, 2010.
8. Nitto, E. D., Ghezzi, C., Metzger, A., Papazoglou, P., Pohl, K. A journey to highly dynamic, self-adaptive service-based applications, *Autom. Softw. Eng.*, 15(3-4), 2008.
9. Stevens, G.C. Integrating the Supply Chain, *International Journal of Physical Distribution & Logistics Management*, 19(8), 1993.
10. Al-Mashari, M., Al-Mudimigh, A., Zairi, M. Enterprise resource planning: A taxonomy of critical factors, *European Journal of Operational Research*, 146(2), 2003.
11. Fjørtoft, K. et al. MIS - Identification and organization of MIS users and processes, *Delivery A*, MARINTEK, 2010.
12. Sleire, H., Wahl, A.M. *Integrated Planning - One road to reach Integrated Operations*. SPE Bergen Conference, 2008.
13. The FREIGHTWISE project; <http://www.freightwise.info/cms/>
14. The TCMS project; <http://www.sintef.no/Home/MARINTEK/Software-developed-at-MARINTEK/TCMS/>