Information and Communication Technologies for Integrated Operations of Ships

Liping Mu

Information and Communication Technologies for Integrated Operations of Ships

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of Philosophiae Doctor (PhD) in Information and Communication Technology

University of Agder
Faculty of Engineering and Science
2013

Doctoral Dissertation by the University of Agder 74
ISBN: 978-82-7117-753-9
ISSN: 1504-9272

Printed by the Printing Office, University of Agder
Kristiansand

To my parents

Preface and Acknowledgments

This dissertation is a result of the research work carried out at the Department of Information and Communication Technology (ICT), University of Agder (UiA) in Grimstad, Norway, from April 2009 to October 2012. This work is part of the project entitled “Maritime Communications - broadband at sea (MarCom)”. The MarCom project is a joint initiative from several research and development institutions, universities and colleges, public authorities and industry. The project is funded by the industry itself, and the Norwegian Research Council.

First of all, I want to express my sincere thanks to my principal supervisor, Professor Andreas Prinz. His valuable guidance, suggestions and continuous support helped me from the very beginning till the completion of this work. I believe that the lessons I learned from working together with him will always guide me forward in the future. I also must say thanks a lot to my co-supervisor, Professor Frank Reichert. Without his encouragement and supervision, I could not have developed an overview understanding of this subject and mapped it into the thesis.

I am very thankful to many former and ongoing fellow PhD students, Terje Gjøsæter, Jaziar Radiant, Merete Skjelten Tveit, Jan Pettersen Nytun, Selo Sulistyo, Trinh Hoang Nguyen, Yohannes Baptista Dafferianto Trinugroho, Anis Yazidi. They were always available to answer my questions and help deal with difficulties in the PhD study. Cheelim Ng, Yuanyuan Ma, Ram Kumar, Martin Choux, Peter Wesseh, Morten Kollerup Bak, all of them are worthy of a very special round of applause. Their company has made my stay in Norway full of fun. We were like a bunch of nerds acting real-life “Friends”. My sincere gratitude goes to following individuals: Xin He, Xuan Zhang, Lei Jiao, Meng Li, Michelle Khoo, who are like my family members abroad. I also want to thank my family in China: my parents, Wenbin Mu and Runlan Li, and my sister Lihua Mu. Their support is the consistent source that leads me to approaching my life happily and peacefully.

Liping Mu
November 2012
Oslo, Norway

Abstract

Over the past three decades, information and communication technologies have filled our daily life with great comfort and convenience. As the technology keeps evolving, user expectations for more challenging cases that can benefit from advanced information and communication technologies are increasing, e.g., the scenario of Integrated Operations (IO) for ships in the maritime domain.

However, to realize integrated operations for ships is a complex task that involves addressing problems such as interoperability among heterogeneous operation applications and connectivity within harsh maritime communication environments. The common approach was to tackle these challenges separately by service integration and communication integration, respectively: each utilizes optimized and independent implementations. Separate solutions work fine within their own contexts, whereas conflicts and inconsistencies can be identified by integrating them together for specific maritime scenarios. Therefore, connection between separate solutions needs to be studied.

In this dissertation, we first take a look at complex systems to obtain useful methodologies applied to integrated operations for ships. Then we study IO of ships from different perspectives and divide the complex task into sub-tasks. We explore separate approaches to these sub-tasks, examine the connection in between, resolve inconsistencies if there are any, and continue the exploration process till a compatible and integrated solution can be accomplished. In general, this journey represents our argument for an integration-oriented complex system development approach. In concrete, it shows the way on how to achieve IO of ships by both providing connectivity in harsh communication environments and allowing interoperability among heterogeneous operation applications, and most importantly by ensuring the synergy in between. This synergy also gives hints on the evolution towards a next generation network architecture for the future Internet.

List of Publications

The author of this dissertation is the principal contributor and the first author of eight out of ten papers listed below. Papers A-F in the first set are selected to represent the main research achievements and are reproduced as Part II of this dissertation. Papers 7-10 listed in the second set are complementary to the main focus. They are not included in this dissertation because of either overlapping work or addressing relatively different topics from the focus of the dissertation.

Papers Included in the Dissertation

Paper A L. Mu, A. Prinz, and C. Erik. Moe, "Towards Integration-Oriented Complex System Development," in *Proc. Norsk konferanse for organisasjoners bruk av informasjonsteknologi (NOKOBIT)*, Tromsø, Norway, 21-23 November 2011.

Paper B L. Mu, L. Kuang and A. Prinz, "Maritime Data Integration Using Standard ISO 15926," in *Proc. the 20th International Offshore and Polar Engineering Conference (ISOPE)*, Beijing, China, 20-26 June 2010.

Paper C L. Mu, R. Kumar, and A. Prinz, "An Integrated Wireless Communication Architecture for Maritime Sector," in *Proc. the 4th International Workshop on Multiple Access Communications (MACOM)*, Trento, Italy, 11-13 September 2010.

Paper D L. Mu, and A. Prinz, "Delay-Oriented Data Traffic Migration in Maritime Mobile Communication Environments," in *Proc. the 4th Fourth International Conference on Ubiquitous and Future Networks (ICUFN)*, Phuket, Thailand, 04-06 July 2012.

Paper E L. Mu, "A Hybrid Network for Maritime on-board Communications," in *Proc. the 8th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Barcelona, Spain, 08-10 October 2012.

Paper F L. Mu, A. Prinz, and F. Reichert, "Towards Integrated Operations for Ships," accepted by *the 9th International Wireless Communications & Mobile Computing Conference (IWCMC)*, Cagliari, Sardinia, Italy, 01-05 July 2013.

Papers Not Included in the Dissertation

Paper 7 L. Mu, and N. Garmann-Johnsen, "On-board Communication Challenges (LAN, SOA and wireless communication)," *Marine Navigation and Safety of Sea Transportation*, 2009.

Paper 8 L. Mu, T. Gjøsæter, A. Prinz, and M. S. Tveit "Specification of Modelling Languages in a flexible meta-model Architecture," in *Proc. the 4th European Conference on Software Architecture*, Copenhagen, Denmark, 23-26 August 2010.

Paper 9 X. He, R. Kumar, L. Mu, and T. Gjøsæter, "Formal Verification of a Cooperative Automatic Repeat Request MAC Protocol," in *Proc. the Norwegian Informatikkonferanse (NIK)*, November 2010.

Paper 10 X. He, R. Kumar, L. Mu, T. Gjøsæter, and F. Y. Li "Formal Verification of a Cooperative Automatic Repeat Request MAC Protocol," *Computer Standards & Interfaces*, December 2011.

Contents

List of Figures

Part I

Chapter 1

Introduction

1.1 The Concept of Integrated Operations for Ships

The term *Integrated Operations* (IO) is widely used in the Oil & Gas industry, referring to integration of people, work processes and technology to deliver smarter decisions and better execution, to act on frequently captured data in real-time, and to ensure more efficient working together [1, 2]. The Norwegian government has defined IO as the use of information technology to achieve better decisions, remote operations of equipment and processes and to move functions as well as people onshore [3]. In the process industry in general, the term is used to describe the increased cooperation, independent of location, between operators, maintenance personnel, electricians, production management, business management and suppliers to provide a more streamlined plant operation [4].

Specifically, operations for ships are requiring increased safety & security and comfort & convenience through advanced information and communication technologies [5]. Newer security and transport related applications such as video surveillance for piracy prevention and real-time updates of navigational data are highly demanded on board. Navigation and operation systems need to be updated frequently with new and revised forms and requirements and fed with new information. Crew and passengers are longing for better Internet services, improved voice and video contact to home and friends, and more connectivity choices for entertainment purposes, etc. Beyond these individual application scenarios is a vision of integrated operations for ships, where functions and personnel can be relocated to shore based on efficient land-based control, surveillance, and management. Examples include real-time or near real-time updates of navigational data from a land-based assistance system, remote surveillance of on-board equipment and devices, remote control, maintenance and the like.

Figure 1.1 gives an abstract vision of integrated operations for ships. With operations that are being truly integrated between ship and shore, the following benefits can be partially or fully obtained: improved decision making, higher efficiency and flexibility of ship operations, increased accuracy and consistency of information, optimized navigation, better ship monitoring and maintenance, better resource exploitation, increased heath and safety, improved regulatory and legal obligation, etc. While the IO vision for ships is beneficial, turning it into reality is a complex task. It requires adopting modern information and communication technologies under the maritime context to achieve ubiquitous operation, communication and real-time information access.

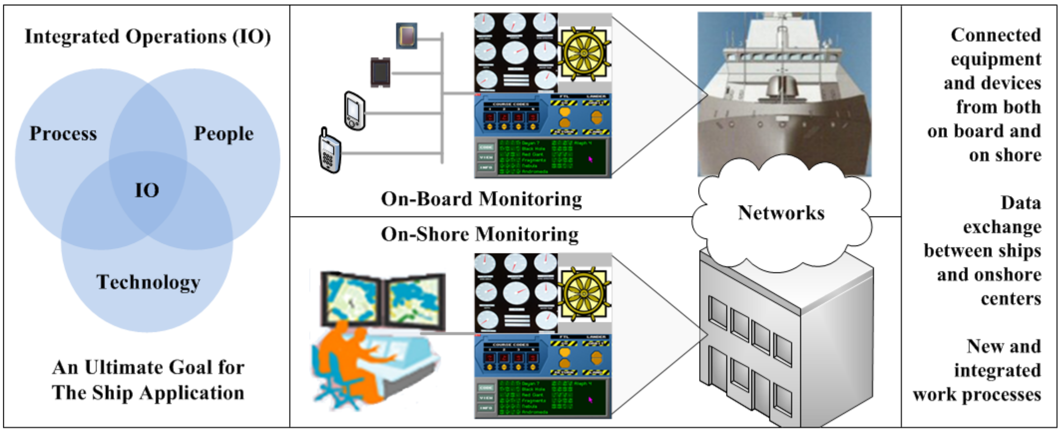

Figure 1.1: The Vision of Integrated Operations for Ships.

1.2 Research Challenges and Objectives

As realizing IO for ships is a complex task, the first objective of this dissertation is to investigate methodologies that can be used for dealing with complex tasks, particularly for developing complex software systems. The complexity of realizing IO for ships comes primarily from: integrating heterogeneous work processes which are isolated, having monolithic functionalities and incompatible data formats, and connecting diverse equipment/devices under harsh maritime communication environments. Two main challenges can be used to describe the complexity: 1) interoperability among heterogeneous process applications or services and 2) connectivity through challenged communication networks. Hence, the second objective is to study existing information and communication technologies and to propose a set of mechanisms for handling these two challenges.

The research work starts with analysing previous methodologies on design and development of complex software systems in order to derive an appropriate approach to integrated operations for ships. Then, detailed proposals and solutions for tackling interoperability and connectivity challenges are examined. More specifically, we attempt to answer the following research questions:

- **Question 1:** How to approach complex systems and especially how to solve the concrete task of integrated operations for ships?
- **Question 2:** How to ensure interoperability among heterogeneous operation applications with isolated, monolithic functionalities and incompatible data formats?
- **Question 3:** How to obtain required connectivity through multiple maritime wireless networks under harsh mobile communication conditions?
- **Question 4:** If both interoperability and connectivity challenges are addressed independently, does it mean that the goal of integrated operations for ships is achieved? If not, what are the inherent reasons and what are the possible solutions?

The first question listed above is to view the research problem from an overall perspective, and the second two questions are targeting at individual challenges. The last question is to revisit the research problem as a whole after the individual challenges have been addressed separately.

Based on the above research questions, detailed literature surveys were performed, existing solutions were studied, potential technologies were investigated, and our tailored approaches, specific mechanisms were proposed. The proposed solutions and results were presented in six published research papers. Figure 1.2 illustrates how those papers answer the research questions one by one, and all the papers are included in this dissertation.

As shown in the figure, we use a divide & conquer policy to deal with the complex research task. Two primary perspectives are explored, i.e., application and network perspectives, where interoperability and connectivity are the dominant challenges. Then, separate service integration and communication integration solutions are utilized to address these two challenges. Within service integration, both data and application integration approaches are examined and a case study for data integration in the maritime domain is conducted. For communication integration, the ship-to-shore and on-board communication scenarios are analysed, and various mechanisms for handling connectivity challenges in both scenarios are proposed. Finally, the connection between service integration and communication integration is studied in order to ensure compatibility and the synergy in between separate solutions.

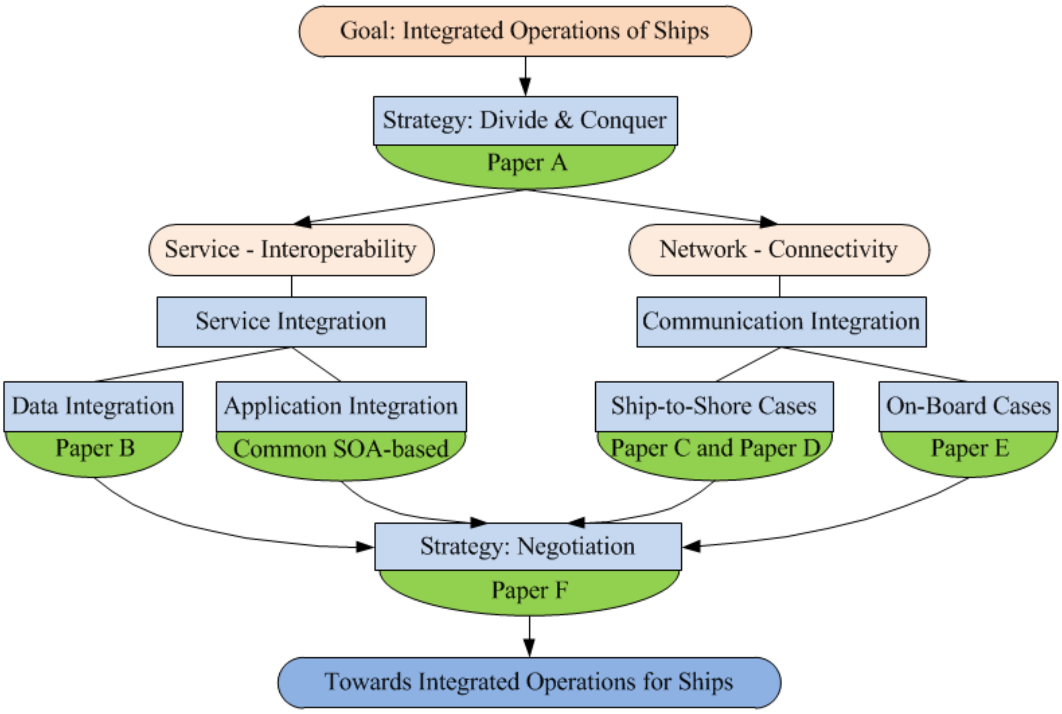

Figure 1.2: The Structure of the Research Work and Connected Papers.

To address the above research questions in detail, four research goals are identified as follows, and they are achieved through the scientific contributions of the thesis highlighted in Papers A-F. The methods to evaluate the proposed mechanisms include literature review, theoretical analysis and computer simulations.

- **Goal 1:** To explore existing methodologies on complexity handling, to propose a solution for modern complex information systems development in order to guide the process of moving towards integrated operations for ships.

- **Goal 2:** To investigate existing service interoperability approaches, specifically on data integration and application integration methods because of the heterogeneity on data source and application logic levels.

- **Goal 3:** To survey current network connectivity mechanisms, and propose novel designs for providing efficient communication services in the maritime environment for both ship-to-shore and on-board cases.

- **Goal 4:** To connect service interoperability and network connectivity approaches together under the maritime context, to examine underlying reasons and propose potential solutions to inconsistencies between the separate approaches if any incompatibilities or conflicts exist.

1.3 Organization of the Dissertation

The dissertation is organized into two parts: Part I and Part II. Part I consists of six chapters providing an overview of the PhD work, and Part II is organized as a collection of six scientific papers (A, B, C, D, E, F). The following chapters are included in Part I:

- Chapter 1 Introduction
- Chapter 2 Research Methodology (based on paper A)
- Chapter 3 Service Integration (based on paper B and the SOA paradigm)
- Chapter 4 Communication Integration (based on papers C, D, E)
- Chapter 5 Negotiation between Independent Solutions (based on paper F)
- Chapter 6 Conclusions and Further Outlook

Chapter 2 focuses on complex systems design and development, especially for modern software information systems which are inherently large and complex. The divide & conquer method is adopted with an emphasis on examining the connection between sub-problems and individual solutions. The first research question is addressed and explored in this chapter. Moreover, it provides the methodology used in the following chapters. Chapter 3 studies service integration approaches focusing on data integration and application integration mainly for the interoperability on data and application logic levels. A case study on data integration in the maritime domain is performed and the common SOA-based (Service Oriented Architecture) integration paradigm is investigated. Then a potential trend for future (maritime) integration solutions is predicted with the consideration of current major integration activities. The second research question is addressed in this chapter. Chapter 4 presents various connectivity optimization solutions within challenging communication environments like the maritime case. Both ship-to-shore and on-board scenarios are analysed and respective connectivity enhancement mechanisms are proposed. The third research question is answered in this chapter. Chapter 5 examines the connection between separate service integration and communication integration solutions, and ensures the compatibility and synergy in between based on a negotiation process in case any inconsistencies are identified. The fourth research question is addressed in this chapter. Chapter 6 summarizes the main contributions of this dissertation and concludes the whole thesis by highlighting some future research directions that require further attention.

The following papers are reproduced in Part II:

Paper A: Towards Integration-Oriented Complex System Development
Paper B: Maritime Data Integration Using Standard ISO 15926
Paper C: An Integrated Wireless Communication Architecture for Maritime Sector
Paper D: Delay-Oriented Data Traffic Migration in Maritime Mobile Environments
Paper E: A Hybrid Network for Maritime On-Board Communications
Paper F: Towards Integrated Operations for Ships

Chapter 2

Research Methodology for Complexity Handling

Integrated operations of ships is to achieve ubiquitous operation and information access based on modern information and communication technologies. The ultimate product is an advanced information system distributed between ship and shore, that can hold data and transform raw data into information, for disseminating knowledge and for supporting integrated processes. The analysis, design and implementation of such an information system are complex. Therefore, it is of great importance that methodologies for complexity handling are investigated first and a guideline for addressing complexities involved in approaching integrated operations of ships can be derived.

In this chapter, we start with a brief introduction to complexities associated with developing modern information systems, present existing efforts on addressing them. Then we point out limitations of the current work and propose an integration-oriented approach to complex systems development which is detailed in Paper A. This approach is applied to the analysis, design and implementation of integrated operations for ships in the following chapters.

2.1 Introduction to Complexities Associated with the Development of Modern Information Systems

The process of developing a modern software information system is a difficult activity requiring contributions from multiple disciplines. According to [6], complexity of such a development process comes from two primary areas: 1) the area of the problem being solved (the problem domain) and 2) the area of constructing a software solution (the solution or implementation domain). Complexity in the problem domain is essential, whereas complexity in the solution domain exhibits accidental characters, e.g., difficulties added to the problem as a result of the way the problem is formulated, the tools adopted for solving it, the technology used to materialize the described solution, etc. The complexity is called accidental because these difficulties are not inherent to the actual problem being solved [7].

Essential complexity represents the difficulty inherent in the problem itself, which is often intertwined with the concept of complex systems. There are many definitions of a complex system. Based on the descriptions in [8, 9], we define a complex system as *a constantly evolving system that is made up of a large number of nonlinearly interacting parts which will make the overall behavior changeable and unpredictable*. When it comes to the development of such a system, especially if the traditional top-down design approach is adopted, two major features of the essential complexity - incompressibility and changeability - can cause problems. Incompressibility means that there is no accurate (or rather, perfect) representation of a complex system which is simpler than the system itself [10]. Therefore, system models or representations will have to "conserve" the complexity and be as complex as the system itself. Changeability indicates dynamic behaviors of a system, influenced by the evolution or changes of its heterogeneous elements and the intricate interaction between them. Changeability requires the system to be adaptable and self-organizable.

2.2 Existing Solutions on Addressing Complexities

Traditional system development approaches focused on addressing accidental complexity from the solution domain, e.g., by advances in programming Languages, middleware and platform technologies, and systematic development methodologies. This can be illustrated from the technical and social aspects, to differentiate the computer technology and the human management factor involved in the process of constructing a software solution. Essential complexity was often partially handled from the changeability aspect, whereas incompressibility was neglected. For example, model based system development approaches represented by the model-driven engineering [11] and the Model Driven Architecture (MDA) [12] are trying to automate the implementation processes in order to easily embrace changes from both problem and solution domains. Traditional waterfall linear system development models are continually replaced by non-linear ones, like the iterative model (Scrum [13]), to provide more ease-of-change during the system development. These traditional development approaches with focusing on addressing accidental complexity and part of the essential complexity is shown in figure 2.1.

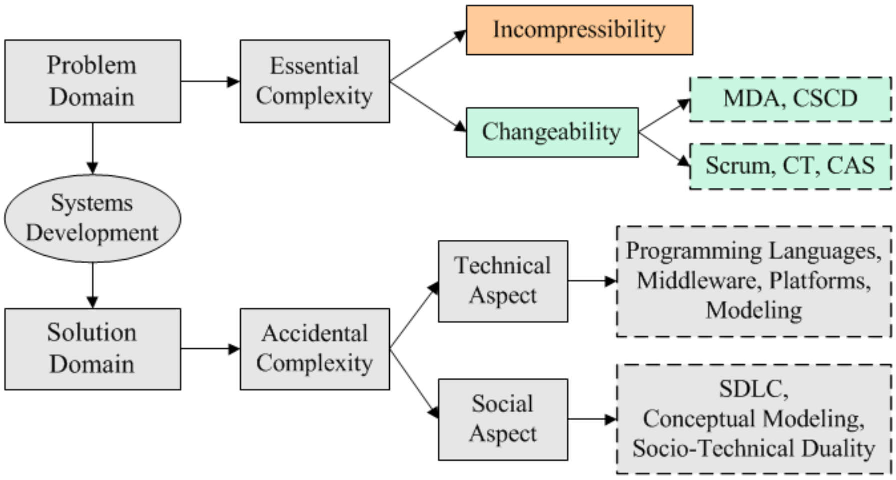

Figure 2.1: Existing Solutions on Addressing Complexities.

Information system development approaches still very much follow a traditional trend by concentrating on the solution domain and the changeability. As modern information systems are becoming highly complex in the problem domain, it has caused many information system development failures [14]. Therefore, growing attention has to be paid to addressing the essential complexity. To decide precisely what to build is always the hardest part of building a software system [15], and incompressibility is probably the single most important aspect when considering the development of any analytical methodology or epistemology for coping with a complex one [8]. Hence, ignorance of incompressibility should come to an end.

Incompressibility occurs when a traditional top-down approach is followed by developing modern large complex information systems. A top-down method attempts to define a model of the target system covering all views of it during system conceptualization. After the conceptualization step, a complete abstract version of the system should be developed and all assumptions are validated. Incompressibility implies that if the to-be-constructed model of a complex system wants to capture all the possible behaviors contained, that model must be at least as complex as the system itself. In other words, the best representation of a complex system is the system itself, and any representation or abstraction of that system will be incomplete and can lead to an incomplete or even wrong understanding. At the same time, changeability of different parts of a complex system can make the top-down development even more impractical because of resulted unpredictability of the whole system. Therefore, for modern complex information systems, a traditional top-down development approach is not appropriate any more; essential complexity and the incompressibility have to be taken into consideration. This inspired our idea of an integration-oriented complex system development methodology, introduced in the next section, which was originally presented in Paper A.

2.3 An Integration-Oriented Approach towards Large Complex Systems

The incompressibility of a complex system can result in an extreme solution: "don't bother". Since there is no accurate representation or abstraction of a complex system which is simpler than the system itself, the only solution appears to be not bothering. However, as we rely heavily on the abstraction of a system to initiate development actions, we need the compressibility of a complex system so that design decisions can be justified and implementation processes can be activated. This compressibility can be achieved or the incompressibility problem of a complex system can be avoided through an integration-oriented approach, as shown in figure 2.2.

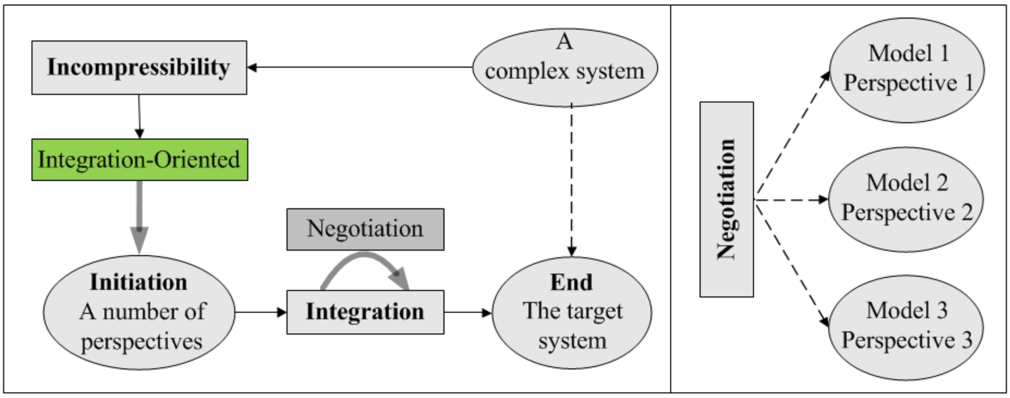

Figure 2.2: Integration-Oriented Complex Systems Development.

If there is no single representation or perspective which can completely capture the inherent intricacies of a complex system, a number of perspectives can be explored with each one capturing parts or an aspect of the whole system. Together, multiple perspectives can develop a richer understanding (if not a complete one) of the targeted system. But it needs to be kept in mind that these perspectives are provisional and context-dependent due to the system changeability. As conditions evolve, we need to review and possibly change the perspectives or rethink of the implementation design itself. Moreover, because of the multiple perspectives' exploration, inconsistencies and conflicts in between different representations/models may happen. Therefore, the whole development process must be treated as a system integration journey, implementing a negotiation or trade-off process to ensure the synergy between different system representations. The connection between multiple representations will be examined in relation to their different perspectives.

It is worth mentioning that our integration-oriented complex system development approach is compatible with the idea of Domain-Driven Design (DDD) [16]. DDD has the prerequisites that the domain is not trivial and there is an iterative process in place. It supports multiple models by maintaining three important concepts: bounded context, continuous integration and context map, see figure 2.3. While the two ideas are similar, the intent is somewhat different. DDD is concerned more with the practice of defining better domain models and we concentrate on the initiation of large complex systems development.

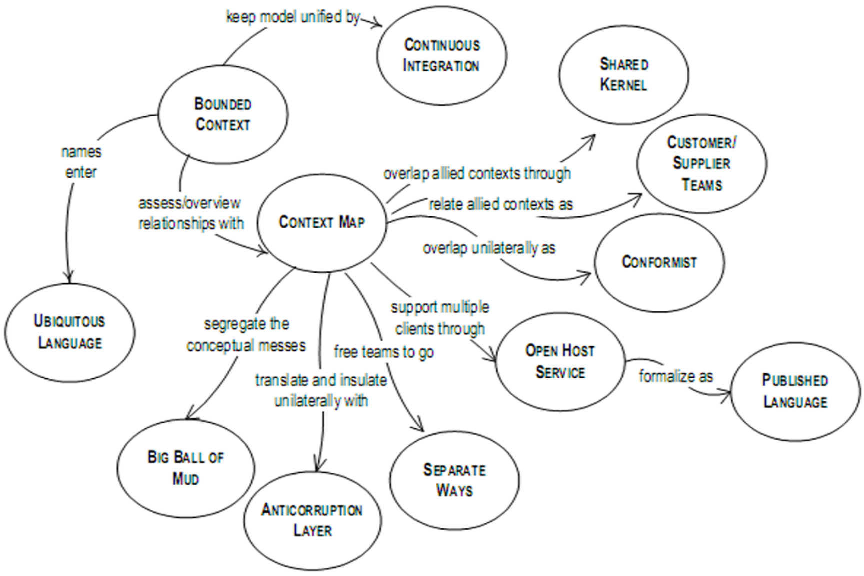

Figure 2.3: Maintaining Model Integrity in Domain-Driven Design [16].

2.3.1 For Large Information Systems

Regarding the development of a large information system, the integration-oriented approach can be considered as to integrate multiple heterogeneous data sources that may belong to different sub information systems. Therefore, a system development problem is translated into a data integration task, and the proposed development methodology can be evaluated by a comparison of diverse data integration solutions.

There are three well-known models for data integration: warehousing, federation and mediation. In warehousing, data must be extracted, transformed and loaded from remote sources to a local central repository named "data warehouse". The central repository provides a single access point to a collection of data copied from heterogeneous sources. This model is very popular for data integration so far, but it is not appropriate for large complex systems, because it tries to design a complete system model before any implementations, ignoring the incompressibility problem. In federation, database systems are distributed and independent, can communicate with each other directly and data can be retrieved via a middleware component. This approach complies with our integration-oriented policy for complex systems, but without enough flexibility, as a conceptual global data model is still required. In mediation, there is a mediator who does not store any data on its own, but rather provides a virtual view of the integrated sources. Wrappers are often used to translate data access and manipulation requests between the mediator and data sources. The mediator splits a user query into sub-queries, sends the sub-queries to appropriate wrappers and integrates the query results locally. Mediation can be considered as an advanced case of federating, by raising the level of federation from a single function or set of data to that of an entire external data source [17]. Therefore, the mediation approach, which does not require a conceptual global model but rather examines the connection between different sub-models, is more flexible than the other two approaches. Connection between local models is used by the mediator to conduct the negotiation process for integration of large information systems.

2.3.2 For Integrated Operations of Ships

Paper A was based on conceptual analysis and review of the literature to give hints on the design and development of complex systems, which can be used as a road map towards integrated operations for ships. We mentioned in chapter 1 that integrated operations for ships is a complex task to accomplish, involving integration of many domains and areas. Therefore, we can treat this task as the development of a complex system and use our integration-oriented methodology, see figure 2.4. We

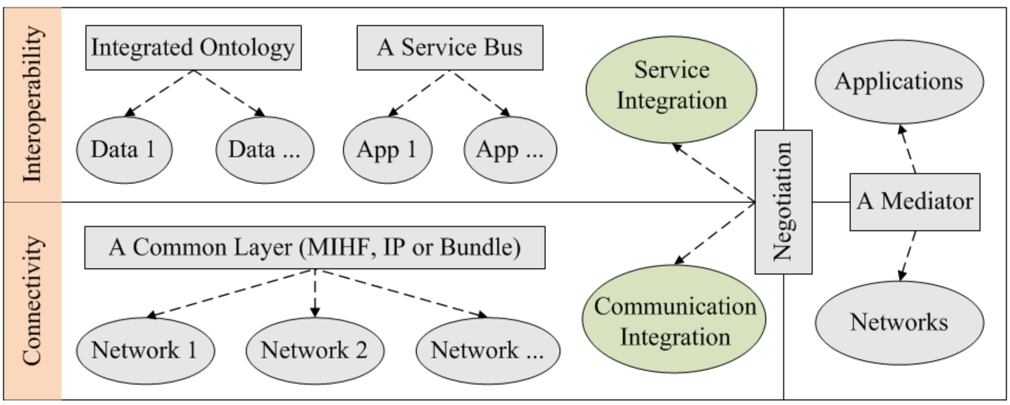

Figure 2.4: Integration-Oriented Approach to Integrated Operations of Ships.

explore the task from multiple perspectives in order to divide it into different sub-tasks and the development process becomes integration among these sub-tasks. As complexity involved in moving towards integrated operations for ships can be generalized as mainly the interoperability challenge between heterogeneous operation services and the connectivity challenge through difficult maritime networks, sub-tasks are derived primarily as service integration to enhance interoperability and communication integration to improve connectivity. Different methods (e.g., advanced information technologies and optimized communication mechanisms) can be used to solve these two sub-tasks, but the connection in between needs to be investigated carefully and negotiation becomes necessary if inconsistencies appear. Within each sub-task, the integration-oriented approach can be applied further even if the system is not as complex as we defined earlier.

We first look at integrated operations for ships from the application and network perspectives, and identify the associated application interoperability and network connectivity challenges. Then we use service integration to tackle the interoperability challenge detailed in chapter 3 and utilize communication integration to ensure connectivity in chapter 4. Within service integration, both data integration and application integration solutions are considered. For data integration, we exploit a mediator-based mechanism to do the integration where an integrated ontology is maintained as the connection between different data sources. Similarly, we follow a service-bus approach to integrating heterogeneous applications, and common integration layers are recommended for connecting heterogeneous communication networks. Finally, the relation between service integration and communication integration sub-tasks will be examined and a negotiation process will be conducted, which is described in chapter 5. The negotiation can be implemented by a communication mediator in between applications and network infrastructures for assisting the interactions among them.

2.4 Chapter Summary

As modern computer-based systems are becoming more and more essentially complex, complexity handling is showing increased importance especially for developing such large systems. This chapter began with an introduction of complexities involved in modern information system development, particularly the essential complexity, and existing methods for dealing with them. We observed that the essential complexity of modern software systems implies an incompressibility problem, when traditional top-down system development approaches are used. The top-down solution tries to build a complete abstract representation/model of the targeted system during the system conceptualization process, which has caused the failure of many large information system development projects. To tackle that, we have proposed an integration-oriented complex system development approach by viewing the system from multiple perspectives based on multiple derived representations and diverse methods. Therefore, system development is translated into integration among different representations. Negotiation/trade-off between different representations is important as inconsistencies may happen. Ontologies can be used to maintain the connection between different perspectives and a mediation mechanism can be applied to implementing the negotiation. This integration-oriented complex system development approach is originally reported in Paper A. The main contribution comes from not only the hints on development of modern large information systems but also its concrete role in guiding the movement towards integrated operations for ships.

Chapter 3

Service Integration for the Interoperability of Heterogeneous Operation Applications

Chapter 3 is about service integration for the interoperability enhancement of heterogeneous operation services, including both data and application integration solutions. We first introduce these two concepts and then summarize existing integration techniques. Then we present our ontology-based data integration which exploits the standard ISO 15926 for the maritime domain. Considering that application integration is becoming more important especially for scenarios where real-time management and control is required, suggestions on how to facilitate maritime application integration are also given. As the target is mainly for ships, we identify challenges that lie behind the data and application integration between ship and shore, which leads to research work that will be presented in the next two chapters.

3.1 Data Integration and Application Integration

In order to perform integrated operations for ships, frequent data exchange between a ship and onshore offices is needed. Information interoperability is desired and excessive human interactions involved in the exchange of data should be minimized, where data integration solutions can be useful. Data integration is about the integration of multiple information systems, aiming at combining selected systems so that they form a unified new whole and give users the illusion of interacting with one single information system [18]. In this thesis, we particularly refer data integration to solutions that are provided by the database community.

Data integration happens after the operation transaction and typically in either inter-day or intra-day batches, so it does not apply to situations where real-time interactions are needed, such as remote management for control, navigation and other critical operations of a ship. Besides, there is more and more data fragmented among distributed applications other than databases. Therefore, data integration alone is not enough any more for the aim of integrated operations of ships; integration among applications has to be taken into consideration. Application integration is to link systems together at the application logic level by sharing processes and/or information between different applications. It can respond to rapidly changing requirements such as new data feeds, changes in logic, and new functions.

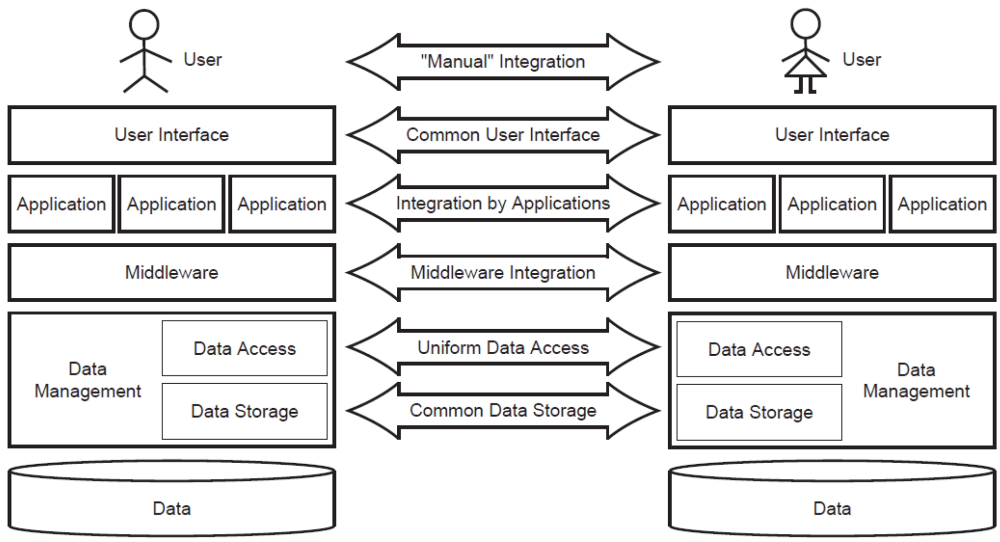

Figure 3.1: Integration Approaches on Different Architectural Levels [19].

Data integration and application integration are approaches to integration between systems from different aspects. According to [19], system integration approaches can be separated by different architectural levels: (1) manual level, (2) user interface level, (3) application level, (4) middleware level, (5) data access level, (6) data storage level, see figure 3.1 (reprinted from [19]). For manual integration, all the integration work is done by the end users. With a common user interface, end users could use an interface like the World Wide Web to make an information query. Application level approaches rely on application processes to do the integration job. Middleware provides reusable functionality used to solve dedicated aspects of the integration problem. For the uniform data access approach, a unified global view of distributed data is provided. The common data storage method has to transfer data to a new data storage with local ones being abandoned or remaining operational.

In this chapter, we first focus on data integration techniques from the database community. We propose an ontology-based data integration for the maritime domain applying the standard ISO 15926. As application integration is critical in the context of integrated operations for ships, we also investigate integration approaches from the application level and pay special attention to the Service-Oriented Architecture (SOA) paradigm due to its enormous popularity. We furthermore identify challenges for achieving integration from an underlying communication perspective.

3.2 Existing Integration Approaches

3.2.1 Data Integration

Data integration, sometimes referred to as information integration, fundamentally provides a layer of abstraction from the underlying data sources, to gather and combine data residing in multiple heterogeneous data sources and present these data in a unified view to users [20] in order to achieve the transparent manipulation of information.

There are different ways to classify existing data integration approaches, for example, the three architectural models mentioned in the previous chapter: warehousing, federation and mediation. Federation and mediation approaches are proposed to solve the problem of tightly-coupled architecture with warehousing. Mediation implemented by wrappers provides the most powerful and flexible infrastructure for federation. It can integrate both function and data by raising the level of federation from a single function or set of data to that of an entire external data source [17]. Wrappers are powerful, but rely on more advanced capabilities of the external source and require a more advanced skill set to implement.

Data integration solutions can be also categorized according to the handling of heterogeneity, i.e., syntactic heterogeneity, schematic (structural) heterogeneity and semantic heterogeneity.Traditional integration solutions are typically using shared information models in formats such as XML, developing common XML interfaces to handle syntactic and schematic heterogeneities. However, they cannot resolve semantic conflicts between heterogeneous data sources. Ontology-based approaches are widely used to address semantic heterogeneity through explicitly defined schema terms and concepts. Ontology-based data integration can be classified into: single ontology approaches, multiple ontologies and hybrid approaches [21].

- Single ontology approaches: All information sources are directly related to one global ontology which provides a shared vocabulary for the specification of the semantics. This approach requires that all information sources have nearly the same view on a domain, with the same level of granularity. This approach is vulnerable to changes in the information sources.

- Multiple ontologies: Each information source is described by its own ontology separately and mapped to each other. This ontology architecture is compatible with change, but an additional representation formalism defining the inter-ontology mappings is necessary.

- Hybrid approaches: A combination of the two preceding approaches is used to overcome the drawbacks of them. A local ontology is built for each information source and mapped to a global shared vocabulary. Sometimes the shared vocabulary is also an ontology. New sources can be easily added with no need for modifying existing mappings. The acquisition and evolution of ontologies are also supported.

3.2.2 Application Integration

Application integration, denotes the process of bringing data or functions together among different applications to achieve shared transactions within an enterprise (sometimes referred to as enterprise application integration) or across different enterprises (sometimes referred to as business-to-business integration) [22]. It is concerned with building and evolving an integration backbone capability that enables fast assembly and dis-assembly of business software components [23]. Within this thesis, we focus on application integration approaches from the process level in order to distinguish them from information/data integration solutions. In practice, process integration and information integration approaches are largely overlapping and supplement each other [24]. We can broadly categorize existing application integration solutions into RPC-based, message-oriented and service-oriented approaches [22]. RPC-based approaches rely on the Remote Procedure Call technique to enable calling operations on remote interfaces to integrate applications. Message-oriented approaches, especially the message-oriented middleware, fosters integration through establishing a shared communication medium between parties, and the development of adapters. Service-oriented approaches depend on the Service Oriented Architecture (SOA) paradigm to simplify integration. Compared with the other two approaches, SOA is more an architectual style that can be used to achieve integration at all architectural levels and handles different kinds of heterogeneity. Service-oriented application integration approach has become essential in the software industry, especially the business domain where new complex applications are likely to base on the composition and collaboration of other services. Therefore, we pay close attention to SOA-based integration technologies.

3.2.2.1 Service Oriented Architecture

Figure 3.2 describes the service oriented architecture and its two implementation mechanisms. SOA was designed for the next generation of distributed software systems and middleware systems by embracing heterogeneity and change that were difficult to handle by previous IT architectures [25]. The basic building block of SOA is a service and there are two key roles: a service consumer (requester) and a service provider, see the left part of figure 3.2. They communicate using bind/invoke interactions via service requests. A service broker may also be involved in order to help a service requester to find published services. It can also help determine which one out of a number of potential service providers should be selected, by maintaining an index of available service providers. The interaction between services can be based on a point-to-point (P2P) pull mode where a service consumer has to pull repeatedly for new information. The interaction can also rely on a push mode where a mediator (service bus) receives published information from the service provider and sends notifications to subscribed service consumers. The pull and push modes are shown in the right part of figure 3.2. The mediated SOA implementation mechanism has more flexibility compared to a P2P mode, and it allows participation of applications/functionalities that are implemented with other (conventional) technologies.

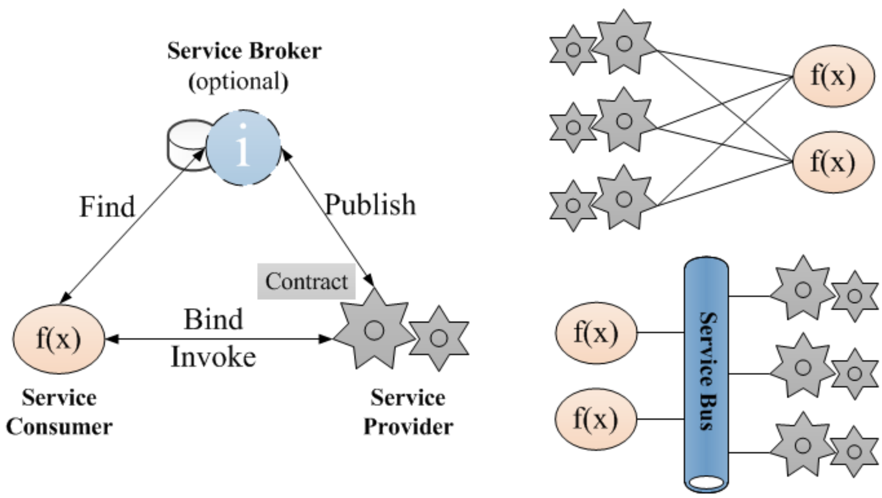

Figure 3.2: Service Oriented Architecture.

The purpose of SOA is to address the requirements of application development for distributed information systems, overcome challenges including application integration, transaction management, security policies, while allowing multiple platforms and protocols and leveraging numerous access devices and legacy systems [26]. In a SOA paradigm, software applications are packaged as "services". Guidelines are provided on how these services are described, discovered and used [26, 23], i.e., services should be standards-based, platform-, protocol-independent and self-contained modules that can interact with each other in heterogeneous environments.

In addition to wrapping applications as "services" to provide application interoperability over networks or more notably the Internet, distributed data sources can also be abstracted as "services" to allow remote data access where the integration is mainly at the data level. This SOA-based data level integration explores "data services" [27] to provide data mediation, integration and also an abstraction for the underlying data sources. Current "data services" implementations mostly adopt the REST architecture [28]. In this thesis, we are focusing on service-oriented integration approaches at the application logic level, specifically the Web services family.

3.2.2.2 Web Services

A Web service defined by (W3C) is: "*a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.*" [29].

Web services, which utilize standards such as Web Services Description Language (WSDL), Simple Object Access Protocol (SOAP), and Universal Description, Discovery and Integration registry (UDDI), together with HTTP and XML, are the most popular type of services available today [23], with maximum service sharing, reuse, and interoperability. Figure 3.3 shows how Web services realize the SOA paradigm, and the sequence flow is as follows:

1. A service provider implements a Web service and describes its interfaces using WSDL. Further, the Web service is published with central service registry.

2. A service consumer looks up for the Web services from a centralized service registry (UDDI) using WSDL.

3. The service consumer binds to a specific service provider for a Web service using a WSDL file.

4. The service consumer creates a service proxy (WSDL).

5. The service consumer communicates with the service provider using SOAP.

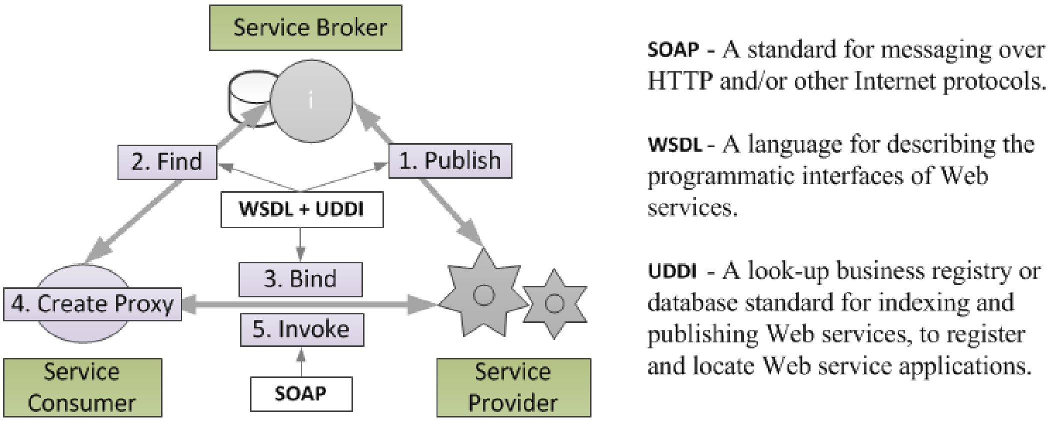

Figure 3.3: The Web Services Realization of SOA.

3.2.2.3 Enterprise Service Bus

Web services denote one important technology for realizing SOA. Other technologies, such as commands in classical SOA and events with event-driven architecture [30], can be also utilized as long as the services obey implementation restrictions of the SOA design principle (e.g., Microsoft's four tenets of service-orientation). Therefore, in order to integrate services with all kinds of implementations and for legacy systems as well, a mechanism must be offered to solve the technology mismatches. Fundamentally, there are two options, a point-to-point (P2P) solution or a middleware mechanism.

The P2P solution requires to develop an "interface" for each connection, introducing a tight form of coupling to harmonize transport protocols, document formats, interaction styles, etc. [31], which is hard to manage and maintain. The second approach introduces a middleware layer that must support interoperability among, and coexist with deployed infrastructure and applications. This middleware layer under the SOA context is well known as a Enterprise Service Bus (ESB) [32, 33], see figure 3.4.

The main aim of an ESB is "to provide virtualization of the enterprise resources, allowing the business logic of the enterprise to be developed and managed independently of the infrastructure, network, and provision of those business services" [34]. Conceptually, the ESB has evolved from the store and forward mechanism found in middleware products, e.g., message oriented middleware, and combines conventional enterprise application integration technologies with Web services, orchestration, choreography technologies, etc. Physically, an ESB provides an implementa-

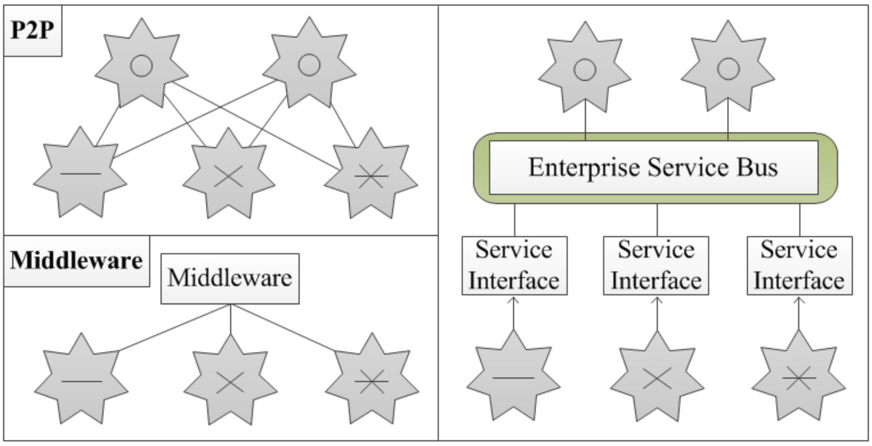

Figure 3.4: Two Integration Options and the Enterprise Service Bus Scenario.

tion backbone for SOA. It establishes proper control of messaging as well as fulfils the needs for security, policy, reliability, and accounting, in a SOA architecture, to support a wide variety of communications patterns over multiple transport protocols and deliver value-added capabilities for SOA applications [23].

3.3 Maritime Data Integration Using the Standard ISO 15926

For the ship and offshore operations where safety and security are a fundamental requirement, a Safety Instrument System (SIS) is often needed to detect and prevent potential dangerous situations. A SIS contains several subsystems: IMS (Information Management System), PCS (Process Control System), PSD (Process shutdown), F&G (Fire & Gas), and ESD (Emergency shutdown system). Each subsystem has a real-time database that stores logged data which can be used to verify the SIS functionality. As these data sources are often heterogeneous in terms of syntax, schema and semantics, it is an advantage to implement integration between them for improved information interoperability.

In order to realize data integration in the maritime domain, especially for overcoming the semantic heterogeneity, we propose an ontology-based data integration approach using the standard ISO 15926 together with Semantic Web technologies. ISO 15926 titled as "Industrial automation systems and integration - Integration of life-cycle data for process plants including Oil & Gas production facilities", is a standard defined for data integration, sharing, exchange, and hand-over between computer systems, to reduce redundant and inconsistent information in sharing data between different companies or organizations.

The methodology of ISO 15926 has been used in several research projects funded by Norwegian Research Council and important stakeholders in the Norwegian offshore industry for developing an Oil & Gas ontology. The Oil & Gas ontology might be expressed in several technologies and is standardized to be a part of ISO 15926. The ISO 15926 has defined syntax, graphical representation and formal semantics, yet not proper implementations. In this case, Semantic Web with its main technologies like RDF (Resource Description Framework), RDF Schema (RDFS), OWL (Web Ontology Language) could play an important role in transforming the ISO 15926 to an ontology language so as to facilitate semantic reasoning.

Therefore, our maritime data integration approach is to use the ISO 15926 standard as a methodology, adopt the Oil & Gas ontology as the integrated ontology, and exploit Semantic Web technologies for expressing ontologies. Figure 3.5 shows the designed system hierarchy which can be considered as a middleware level integration approach, started with identifying common information in diverse data sources. This hierarchy combines the hybrid ontology approach with the mediated data integration model, where local ontologies are used by wrappers for each information source, and the global ontology is the connection between different local ontologies managed by a mediator. A case study based on this data integration approach has been conducted, which is detailed in Paper B included in Part II of this thesis.

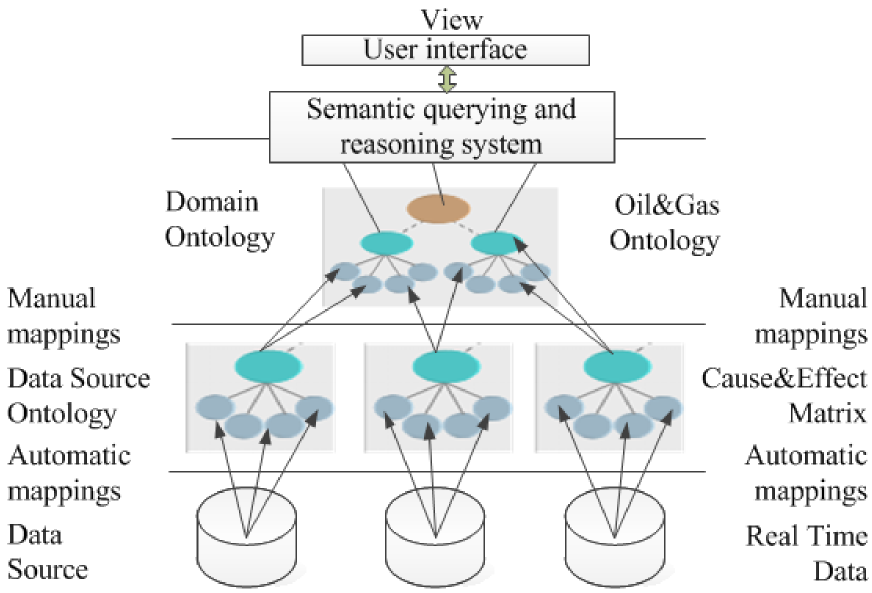

Figure 3.5: Data Integration Hierarchy Implementing the Oil & Gas Ontology.

The case study has shown that the ISO15926 standard is very useful for realizing data integration of the safety instrument system, and a combination of the Oil &

Gas ontology and Semantic Web technologies is effective for handling semantic heterogeneity. Nevertheless, such ontology-based data integration systems may still require semi-automatic or manual mappings at different abstraction levels with the help of domain experts.

3.4 Maritime Application Integration and Challenges

In addition to maritime data integration which could provide a common understanding of the domain, integration at the application logic level is also needed for moving towards integrated operations of ships. Application integration can deal with real-time operations, such as ship monitoring and emergency alarming, remote ship navigation, fuel efficiency management, port and coast control, etc. Different onboard monitoring systems often use different sensor networks from different vendors; onshore management applications are usually based on diverse infrastructures (e.g., for center offices, different port or coastal authorities), and they have to be connected through Internet or enterprise specific solutions. Therefore, SOA-based application integration methodology is preferred because of its capability of dealing with heterogeneity, changeability, and its widely adopted Web services implementation in the industry. A service bus mediated model is suggested to implement SOA-based application integration in the maritime area, shown in figure 3.6.

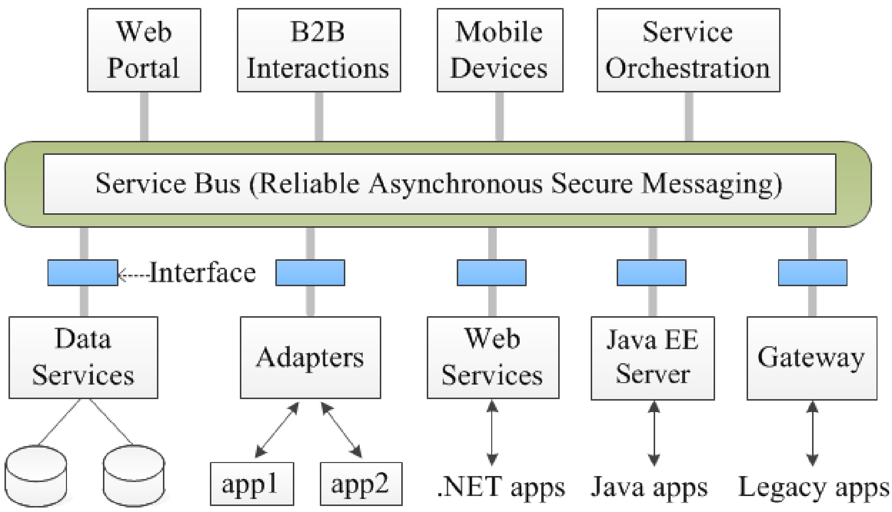

Figure 3.6: Service Bus Connecting Diverse Applications and Technologies.

With a SOA integration solution that combines Web services standards and an ESB, the following requirements can be fulfilled from the application/service perspective: (1) expose a common standard mechanism through which applications can interact, (2) shield services from underlying different operating systems, programming languages, and other technologies, (3) allow for reuse of existing and newly developed services, and (4) offer backwards compatibility and migration to future solutions. Ontology can be further adopted here to understand the specific maritime domain, that is given by the standard ISO 15926 described in Paper B.

However, as integrated operations for ships is not only for well-connected scenarios, but mainly for ship-to-shore cases where connectivity underneath exhibits as a big challenge, e.g., transmission of data and control flows among applications between ship and shore. Therefore, maritime communication challenges have to be addressed as well in order to integrate applications seamlessly, which will be discussed in the next chapter.

3.5 Chapter Summary

In this chapter, data integration and application integration concepts are introduced first, and then various data integration techniques are presented. As application integration becomes more and more important, the service-oriented integration approach is particularly studied due to its popularity. Web services and ESB technologies are briefly depicted. Then, we introduce an ontology-based data integration mechanism targeting at maritime scenarios which is originally reported in Paper B. This approach takes advantage of the well-established ISO15926 standard and the Semantic Web technologies to handle heterogeneities exhibited in maritime data integration cases, especially the semantic heterogeneity. The study demonstrates that for a specific domain such as the Oil & Gas industry, an ontology-based data integration approach is preferable and tends to work, but shows limitations when it comes to real-time operations. These real-time operations require integration at the application logic level, where an appropriate application integration mechanism is necessary. Considering that SOA-based integration technologies have gained much success in the software industry from the last decade, we feel the SOA idea can be directly applied to maritime scenarios. Therefore, the SOA integration methodology which is not covered by attached papers in Part II, is quite thoroughly elaborated in this chapter. The main challenge of applying SOA-based application integration mechanisms in the maritime environment lies in the underlying communication problems, which leads to investigations presented in the following chapters.

Chapter 4

Communication Integration for the Connectivity in Challenging Maritime Environments

In the previous chapter, different data and application integration solutions are explored in order to facilitate maritime information interoperability and interoperability among applications. We observed that another underlying challenge of enabling real-time integration between ship and shore is the communication problem. In this chapter, we investigate various maritime communication challenges extensively and propose corresponding mechanisms to address them. Both ship-to-shore and on-board cases are studied so as to cover most communication scenarios involved in fulfilling integrated operations of ships.

4.1 Maritime Communication Challenges and System Requirements

Due to the difficulty of deploying cellular systems at sea to achieve high data-rate transmission, maritime communication has to live with limited bandwidth based on other wireless communication technologies. So far, FM (frequency modulation) radio technology like narrowband UHF (ultra high frequency) and VHF (very high frequency) are widely used for ship-to-shore and ship-to-ship communications, with satellites such as Inmarsat (international maritime satellite) [35] and maritime VSAT (very small aperture terminal) [36] systems used for long-range cases especially broadband services. Compared with terrestrial networks, these systems show limitations in terms of, e.g., transmission data-rate, transmission delay, bit error rate and communication cost. Although cellular networks and WiFi can be considered, they are only available for near port waters. Switching between these heterogeneous communication networks is not trivial either, because of multiple radio access technologies, diverse network protocols and administrative concerns, and it often results in unsatisfying communication quality with great variations. Moreover, a ship itself is a harsh environment for on-board communications.

Therefore, a current maritime communication scenario can be characterized with intermittent connectivity, long communication delays, a heterogeneous mix of networking elements and widely varying network conditions. Hence, the targeted maritime communication system needs to:

- Handle extreme connectivity challenges and leverage heterogeneous networks to provide basic connectivity services anytime, anywhere and on any devices,
- Resolve the network heterogeneity in order to take advantage of diverse maritime communication resources by enhanced mobility, quality and security provisioning.

4.2 Existing Communication Architecture Designs

In order to develop a communication system that satisfies the aforementioned requirements, a well-designed communication network architecture is the first key step. Existing approaches can be coarsely divided into three categories, one is IP-based solutions usually related to application-independent mechanisms, another is application-specific solutions represented by overlays, and the third is clean-slate designs. IP and overlay solutions are discussed in section 4.3 and 4.4, respectively. A clean-slate approach is to redesign the network from scratch to offer improved abstractions and/or performance, while providing similar functionality based on new core principles [37]. Compared with incremental approaches such as IP solutions and overlays, clean-slate designs are expected to solve the network architectural challenges, e.g., deficiencies of the current Internet architecture, fundamentally. However, in the maritime environment, it is considered relatively impractical to embrace a clean-slate redesign because of the difficulty of deployment on the large worldwide base of existing (IP) networks. Therefore, we focus on IP-based solutions and overlays. IP-based solutions often target at situations where connectivity problems can still be handled by incremental fixes and patches to the traditional TCP/IP architecture, and overlays often tend to solve extreme connectivity issues exhibited in resource-scarce networks such as sensor and satellite networks. We investigate various design options for the maritime communication scenario following these two directions, with special attention to their ability of ensuring connectivity among heterogeneous wireless networks.

4.3 IP-based Solutions

IP, as a common interconnection element to handle heterogeneity, resulting in fast service provisioning by dumb networks and intelligent end systems, has brought incredible success and rapid growth to the Internet. Nevertheless, IP alone is not enough to address all the challenges involved in the maritime communication scenarios. Cross-layer optimization mechanisms have to be introduced in order to address mobility, quality and security challenges.

We propose an integrated wireless Communication Architecture for Maritime Sector (CAMS) that optimally leverages the simple IP technology and sophisticated cross-layer optimization techniques. IP is used as the unifying technology to integrate diverse maritime wireless networks. Cross-layer optimization mechanisms, such as Media Independent Handover (MIH) layer, Authentication, Authorization, and Accounting (AAA) functions, Virtual Private Networking (VPN) technology, Host Identity Protocol (HIP) and specific application-level quality control, are applied so as to enhance quality and security in face of various maritime mobility scenarios.

The MIH mechanism is strongly suggested in CAMS because handover is the key enabling function for seamless mobility and service continuity among heterogeneous wireless communication networks. The MIH mechanism is first introduced in the IEEE 802.21 standard [38] existed as a Media-Independent Handover Function (MIHF) between link and network layers. The function can be exploited by the IP stack (or any other upper layer) to better interact with the diverse underlying technologies from mapping technology-specific primitives. To upper layers, it provides a media-independent interface in order to collect information from link layer and to control link behavior. Regarding the different link layer technologies, MIHF supports mapping between the common interface and a set of media-specific primitives. As the MIH mechanism is designed to enable interoperability mainly among IEEE 802, 3GPP, and 3GPP2 networks. A satellite extension is necessary for maritime cases, where the satellite independent service access point (SI-SAP) interface layer can be useful. SI-SAP is defined as part of the broadband satellite multimedia architecture [39] to provide a mechanism to carry IP-based protocols over different satellite networks.

Since clearly different network operators will provide coverage areas for the maritime customers, the AAA functions appear to be necessary to allow customers to perform authentication and authorization processes in visited networks based on subscription to a home network. The VPN technology can further help with secure data transfer between networked devices which are not on the same network and keeping the transferred data private from other devices or other intervening networks. As service continuity between different networks often relies on the maintenance of a permanent mobile terminal IP address, Mobile IP, or its variant Mobile IPv6 or another alternative technology like HIP is needed. At the same time, application-level quality control mechanisms can exploit limited maritime bandwidth smartly according to diverse application requirements, by 1) assigning applications with different priorities and 2) queuing their connections based on various network conditions.

Security is not particularly addressed by CAMS, but it can be enhanced by the aforementioned mobility handling and application-level traffic control solutions. Description of this CAMS architecture is detailed in Paper C in Part II.

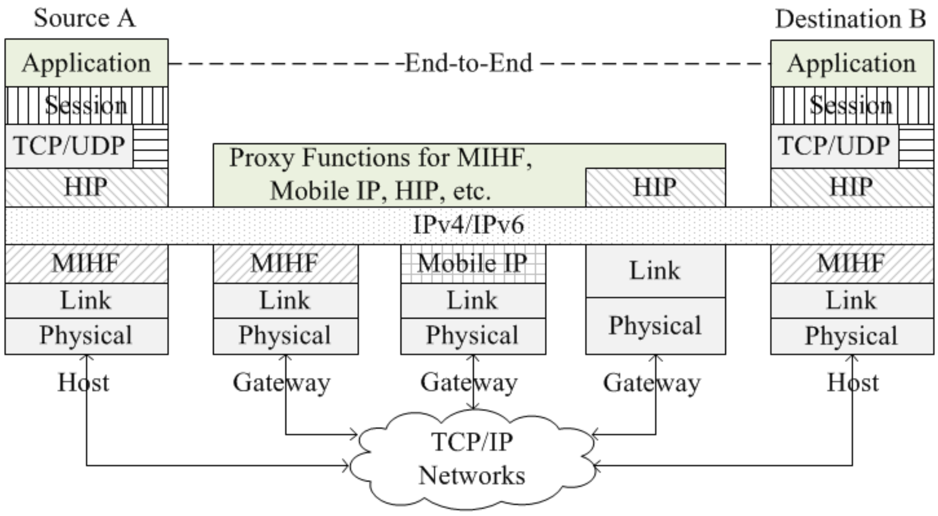

Figure 4.1: IP-based Solution with Optimization Mechanisms at Different Layers

All in all, IP-based solutions can be summarized as various cross-layer optimizations, i.e., session- and transport-layer enhancement mechanisms, HIP, Mobile IP, MIHF and other specific functions, employed around a common network layer, see figure 4.1. These optimization solutions utilize variants of incremental fixes and patches under the application layer to allow traditional (Internet) applications function properly, e.g., in wireless and mobile networking environments, and keep applications as less impacted as possible. Therefore, they have advantages such as backward compatibility, scalability, interactivity with the Internet, but they have limitations when it comes to situations like extreme connectivity challenges or interoperability with non-IP networks.

4.4 Overlay Solutions

IP-based cross-layer optimization mechanisms focus on mitigating the risk of connectivity loss and maximizing the connection quality from abstraction, hiding the particulars of the network from applications. This abstraction works well as long as, from a higher layer perspective, connectivity is not lost for too long and the necessary end-to-end communication can be established [40].

However, this assumption may not hold for many maritime communication conditions because of bandwidth limitation, coverage gap, user preferences, etc. The current primary means for broadband communication between ship and shore is via satellite networks. Satellite communication is expensive, usually with limited capacity, high propagation delay and high bit error rate. There are special problems related to lack of satellite coverage in places such as fjords, ports and at high latitudes. Moreover, most maritime customers today tend to use other means to communicate rather than satellites, especially when large amounts of data usage can be incurred.

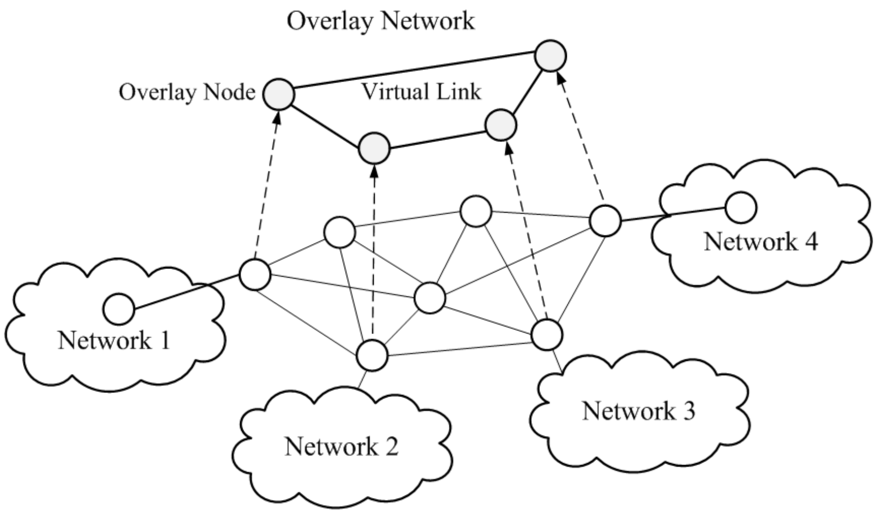

Figure 4.2: An Overlay Network for Connections between Different Networks

Therefore, in addition to IP-based mechanisms, alternative solutions like specific application protocols, or overlay networks, should be considered to further address maritime connectivity challenges. An overlay network is an application-specific network built on top of another network, which creates a virtual topology over the physical topology to provide an application or service not easily provided by traditional methods to end users, see figure 4.2. An overlay network, along with other similar ideas: virtual local area networks, virtual private networks and active/programmable networks, are the sources that triggered current network virtualization approaches [41] to coexisting heterogeneous network architectures facing with the inherent limitations of the Internet.

Compared with IP-based optimization solutions, an overlay approach can be incrementally deployed on end-hosts running the overlay protocol software, without cooperation from (Internet) service providers. It has been a very efficient way of embracing new networking technologies, e.g., the Internet was originally built as an overlay upon the telephone network, and today the telephone network is increasingly turning into an overlay network built on top of the Internet, especially as application-layer overlays [42] for content delivery, multicast, QoS, enhanced routing performance, security and so on.

Therefore, we choose the overlay approach to supplement IP-based solutions, for addressing both on-board and ship-to-shore communication challenges. Specifically, the Delay-Tolerant Networking (DTN) [43] architecture, see figure 4.3, is explored because of its ability of handling both heterogeneity and extreme connectivity problems by the transport-layer overlay implementation.

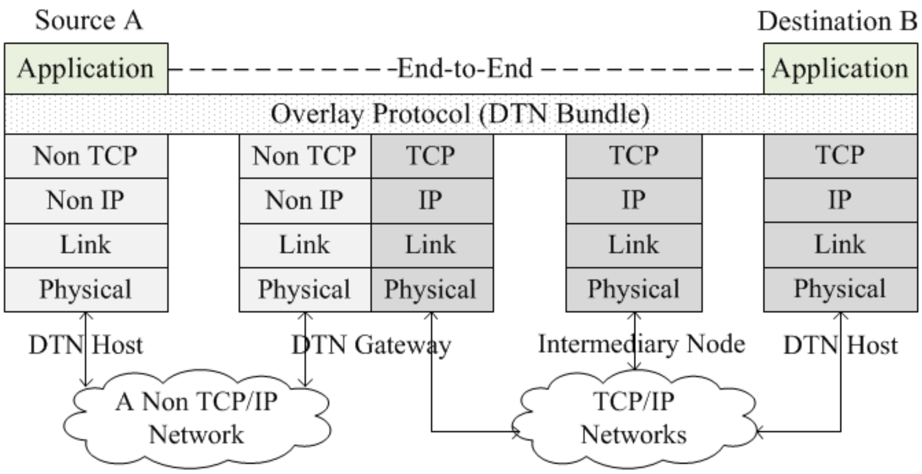

Figure 4.3: Protocol Stack for the DTN Transport-layer Overlay Network

4.4.1 The Ship-to-Shore Communication Case

The current primary means for broadband communication between ship and shore is via satellite networks. Due to the cost issue and some other problems with satellite communication, maritime customers still prefer traditional communication means especially for large amounts of data usage. As WiFi networks have been deployed in ports together with 3G/4G networks installed along the coasts and offshore areas, terrestrial networks started to contribute more to maritime communication. One solution to improve ship-to-shore broadband communication is to take advantage of different features of multiple wireless communication technologies from integrating different maritime networks. Mobility handling and seamless handover between these integrated networks have been discussed in the previous section with IP-based approaches, but the traffic migration between these networks is mainly based on network connectivity conditions whereas user preferences are ignored.

Therefore, we further propose a delay-oriented data traffic migration mechanism to allow user-preferred migration on delay-tolerant applications. It relies on a Delay-Tolerant Networking (DTN) architecture with DTN gateways deployed both on a ship and on shore. The on-board DTN gateway enables the connection between local networks and the Internet through either satellite link or other terrestrial networks. Apart from that, it also decides whether the application should be delayed, and handover to which candidate network, based on 1) the current ship location, 2) the history data and future schedule about travelling routes and 3) characteristics of the application.

The detailed description of this DTN-based traffic migration for maritime ship-to-shore communication can be found in Paper D in Part II. The proof that is originally reported in Paper D shows that a DTN overlay approach can provide maritime customers with not only reliable data transfer but also user preferred data migration, allow them to take advantage of heterogeneous communication systems and utilize preferred networks as much as possible. As the basic logic is simulated and verified before any real implementation, future deployment is justified in a way that can involve more maritime users to be interested in it.

4.4.2 The On-Board Communication Case

Maritime on-board communication provides basic data for the ship-to-ship and ship-to-shore communications. It is essential for maritime safety, security, integrated operations and for infotainment purposes. User groups involved in the local communication include on-board equipment, cargo elements, crew and passengers. Communication between them are supported by the on-board infrastructure and services, such as a sensor network, a personnel tracking system and a wireless local area network (WLAN). Due to the harsh environment of ships for wireless communication and the cost consideration, current communication solutions on board are not satisfying. A lot of recent work has concentrated on implementing ubiquitous technology for ship application such as employing Wireless Sensor Networks (WSNs) on board. However, real deployments still meet difficulties mainly because of the connectivity challenges. Therefore, we propose a hybrid network solution to maritime on-board communication where connectivity is extensively examined. At the same time, emergency situations are taken into consideration. Two key aspects of this proposal are: 1) to combine different on-board communication networks into a hybrid system, enabling the cooperation between them, 2) to smoothly integrate these networks with a consideration of backward compatibility, ease of deployment and connection to shore via Internet. The detailed description of this hybrid network solution can be found in Paper E in Part II.

In order to test that the integrated hybrid on-board network will increase communication efficiency and system reliability, we used computer simulation to evaluate some common scenarios, i.e., data collection in a monitoring sensor network, information dissemination for personnel tracking, and direct communication among people during emergencies. The simulation results based on these scenarios have proved that the hybrid network can provide better communication performance and increased reliability than separate systems, when faced with connectivity challenges or infrastructure damages.

In addition, we investigated different methods of integrating WSNs with external networks, especially TCP/IP networks as the TCP/IP protocol suite is the de-facto networking standard for both the global Internet and local-area networks. We suggest an integration roadmap for maritime scenarios based on a combination of different methods using the DTN networking architecture as a transition phase. Future on-board WSNs can be deployed based on the TCP/IP protocol stack, and they communicate with an on-board DTN gateway using local area networks. The DTN gateway will transmit the gathered information over the global Internet at places where it obtains preferred Internet access, e.g., WiFi at ports. This on-board DTN gateway is either a static server or a mobile device carried by a crew member. From the comprehensive analysis, we believe that this gradual network integration roadmap based on a combination of three well-known methods and use the DTN overlay architecture as a transitional step, can smoothly achieve the final seamless integration of diverse maritime communication systems.

4.5 Chapter Summary

In this chapter, various IP-based and overlay mechanisms are presented in order to tackle the communication challenges exhibited in the maritime environment. An integrated network architecture focusing on ship-to-shore communication is proposed which is originally reported in Paper C. This architecture leverages the widely used IP technology and its cross-layer optimizations for primarily mobility handling, and utilizes an adapted media independent handover mechanism tailored for the maritime cases. A DTN-based overlay approach is proposed to facilitate user-preferred data traffic migration reported in Paper D, and a DTN-based hybrid on-board network is studied in order to improve communication conditions on board, which is presented in Paper E. IP-based solutions are using specific optimization mechanisms from the network part to adapt the network as much as possible to challenging communication situations in order to support traditional applications. Overlay solutions such as the DTN architecture is to enable specific applications running in challenged areas through application-specific support on overlay nodes. Under certain circumstances, both solutions can work well and are useful for many maritime scenarios. However, as our IP-based optimization solutions focus on mobility handling and the DTN approach is specifically useful for addressing extreme connectivity challenges, quality and security requirements are not well covered. Besides, these specific IP-based optimizations and DTN solutions may work fine for current situations, what if use requirements change or communication technology evolves in future. Moreover, connection between these communication integration mechanisms and the service integration approach needs examination as well so as to form a complete solution to the task of integrated operations for ships. All these inspire a further research study that will be presented in the next chapter.

Chapter 5

Negotiation Between Service and Communication Integration Solutions

In order to handle complexity involved in moving towards Integrated Operations (IO) for ships, we applied our integration-oriented complex system development approach and divided the task into two sub-tasks, i.e, service integration and communication integration. The complexity-handling methodology was described in chapter 2, and the related service integration and communication integration sub-tasks were presented in chapter 3 and chapter 4 respectively. We pointed out in chapter 2 that: although system development complexity can be tackled by the exploration from multiple perspectives and using different corresponding methods, the connection between these perspectives should be well taken care of, and the whole development process should be treated as an integration journey among sub-processes. In our case, separate maritime service and communication integration solutions have been investigated individually in order to reduce the complexity of realizing integrated operations for ships. In what follows, the connection between these separate approaches will be examined, and negotiation will be carried out if inconsistencies among them are found, so as to form a complete solution to the complex IO task.

5.1 Connection Between Separate Solutions

From the application perspective, IO for ships requires the interaction among different applications or services. The challenge lies primarily in the interoperability issue caused by heterogeneous and isolated applications with monolithic functionalities and incompatible data formats, see the left part of figure 5.1.

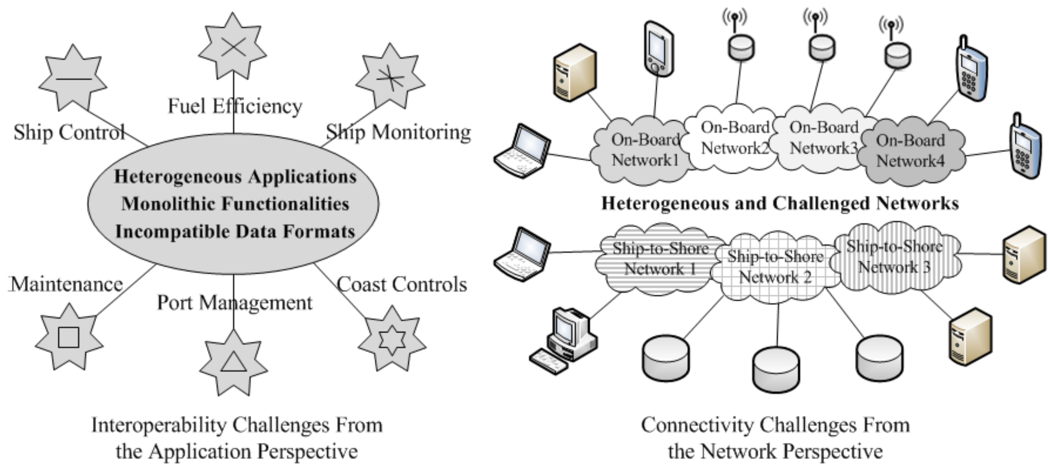

Figure 5.1: View System Development Challenges from Different Perspectives.

The integration of different applications can be achieved from either a data level or an application logic level. Our service integration approach consists of both solutions. Data integration focuses on data level interaction from the database community. It shows limitations when real-time interactions between operation processes are involved, but provides a common understanding of the maritime domain through ontologies. Application integration focuses on business logic level interaction, with also requiring integration at the data level to exchange information between services, non-service applications and other data sources.

As mentioned in chapter 3, an application integration environment based on SOA is becoming popular due to its independence of operating systems, computational platforms and programming languages. SOA can be implemented at different integration levels [22], such as SOAP Web services realization for business logic interaction and RESTful services for integrating heterogeneous data sources. Therefore, we expect that a service integration solution following SOA, which explores the Web services technology and a service bus mechanism, together with a common understanding provided by the maritime ontology, can handle interoperability challenges exhibited in IO for ships. This service integration approach is able to fulfill the system requirements as follows:

A) expose a standard mechanism through which applications can interact
B) shield services from different underlying technologies
C) allow for reuse of existing and newly developed services
D) offer backwards compatibility and migration to future solutions

However, there is an inherent assumption behind the Web services technology:

networks underneath must be well-connected, so that the traditional TCP/IP protocol suite can function properly based on ideal connectivity provided. Hence, we must examine IO for ships also from the underlying network perspective to check the assumed communication conditions.

From the network point of view, IO for ships is to connect devices or equipment located on board and on shore, where both on-board and ship-to-shore networks are included, shown in the right part of figure 5.1. These networks are heterogeneous regarding radio access technologies, network protocols and application scenarios. They also exhibit as "challenged networks" [43, 44], due to environmental restrictions, specific communication settings, system architectures, and the inherent characteristics of certain link layer technologies. Challenged networks are characterized as facing one or more of the following deficiencies: high error rates, long communication delays, unpredictable link availability, severe interferences, very low data rates, or non-existent end-to-end paths.

Heterogeneous and challenged networks introduce difficult communication situations where Web-services based integration requirements or the Internet design assumption cannot be met. In order to tackle that, the communication system is desired to satisfy the following connectivity, mobility, quality and security criteria adapted from [45]:

1) ensure robustness in presence of intermittent connectivity

2) allow fast detection and dynamic attachment to available resources

3) provide reliable transmission between network elements

4) support naming and late binding

5) enable interoperability among heterogeneous communication networks

6) provide various types of mobility support

7) support multiple classes of service

8) allow various communication patterns, unicast, multicast, anycast, etc.

9) facilitate content-, context- and network-awareness

10) provide effective trust&cost and enhanced security&privacy mechanisms

This list shows a clear picture of what to consider if the current communication system (Internet) architecture can be redesigned from scratch following a clean-slate approach [37], especially for challenged areas. However, due to the largely expanded existing communication systems, clean-slate approaches are not practical here. It is then unrealistic to expect all the aforementioned features to be fixed and covered by incremental changes, e.g., IP-based optimizations or an overlay approach that we proposed in chapter 4.

5.2 Conflicts Between Separate Solutions

We therefore notice that inconsistencies can arise by combining the separate service integration and communication integration solutions together in the maritime context.

On the one hand, for SOA-based service integration which relies on the Web technology, applications must be responsible for establishing all bindings necessary to perform communication, without awareness of the nature of underlying communication networks. As a result, applications have to assume the implicit design, conventions, and operating modes of the networking underneath, and it is difficult for them to adapt to different or new communication mechanisms [46]. For example, application integration which uses SOAP Web services technology expects continuous and end-to-end connectivity provided by the network based on the TCP/IP protocol suite. This is explained as “Please give me what I assumed” from the service integration solution in figure 5.2.

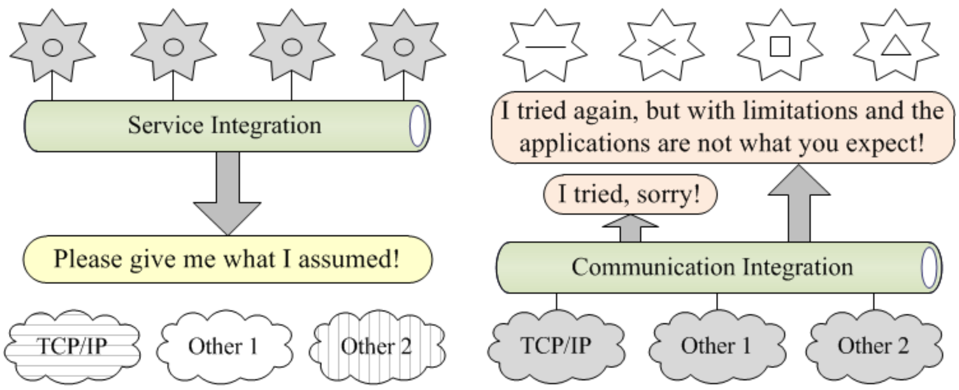

Figure 5.2: Inconsistencies Between Separate Solutions.

On the other hand, a IP-based networking solution tries to match the SOA trend by attempting to maintain continuous and end-to-end connectivity as much as possible from its optimization mechanisms. These mechanisms can provide better delay tolerance and higher location flexibility. Nevertheless, though the Internet (IP protocol) model maintains end-to-end reachability and supports heterogeneity, it does so by requiring every node to use a common network layer host identifier (IP address), packet format with universally-obeyed semantics, and routing methodology that assumes a connected routing graph [47]. These constraints are difficult to satisfy in harsh environments. IP-based optimization design will ultimately fail if the delay threshold is broken or no end-to-end path is available, i.e., the challenged maritime area. It can be explained as “I tried, sorry!” as shown in figure 5.2.

In order to address that, the DTN overlay was recommended. A DTN transport-overlay can ensure robustness in presence of intermittent connectivity, can provide reliable transmission between network elements, can support naming and late binding, can support multiple classes of service, etc. But it lacks sufficient security, mobility support, an end-to-end reliability mechanism and error detection at the bundle layer [48]. It is not yet widely implemented because of, e.g., lack of a killer application [49]. Other points of concern include time synchronization, fragmentation, and meta-data parsing complexity [50]. Furthermore, DTN implementation requires modification from traditional applications and existing application protocols, which often leads to specialized implementations and specific applications. Specialized implementations or specific applications are not in line with the SOA methodology, requiring adapters or gateways to be deployed for the interaction with other standard services. This is explained as "I tried again, but with limitations and the applications are not what you expect!" in figure 5.2.

5.3 Negotiation Applied to the Integration of Separate Solutions

Due to the aforementioned inconsistencies, a negotiation process between separate service and communication integration solutions is necessary, as shown in figure.5.3:

- First, it is necessary for applications to be able to adapt to changing network conditions, responding to different connectivity options, protocols, etc., in addition to relying on the network to adapt to existing or legacy applications through incremental fixes or patches.
- Second, instead of letting each application adapt to different communication characteristics individually or specifically, there should be a mediation mechanism in between applications and networks to negotiate the interaction between them, decoupling applications from the underneath networking details.
- Third, the mediation mechanism should be implemented in a way that newly developed or modified applications are interoperable with each other and with standard services, and future application requirements and emerging network technologies can be easily brought in.

The negotiation between separate solutions, is a trade-off process based on analysis of the merits and deficiencies of each solution in light of both the supporting

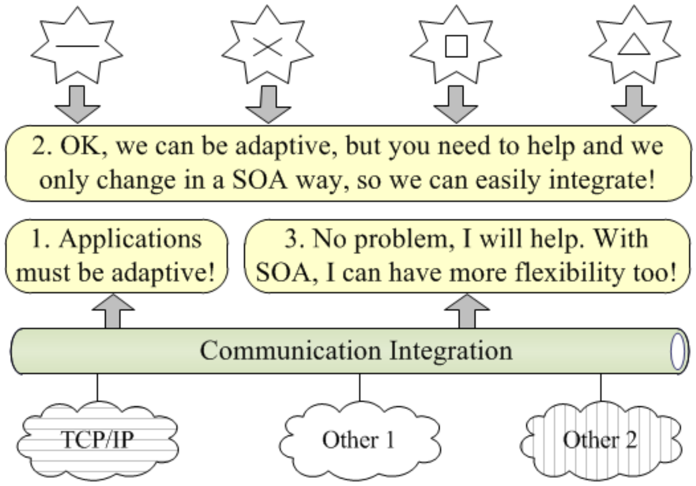

Figure 5.3: Negotiation Between Separate Solutions.

and contradictory evidence offered by the other solutions. For example, due to the specific maritime context, a IP-based solution should give way to other alternative approaches such as the DTN overlay through a Bundle mediation layer to allow applications adapt to various maritime networking environments. However, a DTN solution is not a final answer to evolving communication technologies or to constantly changing application requirements whereas the SOA methodology adopted in the service integration domain may play an important role. Therefore, a generic communication mediator which implements a similar DTN mechanism and obeys the SOA philosophy for embracing heterogeneity and change, can be a more appropriate answer here.

5.4 Implementing a Service-Oriented Communication Mediator

In order to implement the service-oriented mediator, it is necessary to look at related work especially from the following two aspects: communication mediation mechanisms for handling connectivity challenges and the evolution towards service-oriented network architectures.

5.4.1 Existing Network Mediation Mechanisms

The traditional TCP/IP based networking implies a tight temporal and spatial (location) coupling between applications and the underlying networks. That is, the application must know the destination of the transfer (spatial coupling) and be involved in the duration of the transfer (temporal coupling) [51]. These spatial and temporal bonds introduce difficulties for Internet applications running in harsh (mobile) networking situations. Therefore, in the networking literature, there is a lot of research work focusing on how to mediate the application to deal with a variety of communication conditions, e.g., node-centric approaches like DTN [43], data/information-centric solutions such as Haggle [46], DONA [52] and NetInf [53], service-centric approaches as Serval [54] and solutions focusing on network evolvability like XIA [55], FII [56] and OPAE [57].

DTN (Delay Tolerant Networking) aims at the interoperability among heterogeneous networks and the robustness in presence of intermittent connectivity by temporal decoupling. Many networks that are deployed in mobile and extreme environments, often do not utilize IP and have their own specialized protocols. DTN as a network architecture interconnects these networks via key services such as in-network data storage and retransmission, interoperable naming, authenticated forwarding and a coarse-grained class of service, with limited expectations of end-to-end connectivity and node resources.

Haggle, as an attempt to achieve both spatial and temporal decoupling, aims at allowing applications to take advantage of all types of data transfer (neighbourhood, infrastructure, mobility). Networking endpoints are specified by user-level naming schemes rather than node-specific network addresses, and limited resources can be used efficiently by taking into account user-level priorities for tasks.

DONA (Data Oriented Network Architecture), which involves a redesign of the Internet naming and name resolution, takes into account that the vast majority of today's Internet usage is data retrieval and service access. It provides a name-based anycast abstraction instead of host-oriented addressing to realize spatial decoupling between applications and networks.

NetInf (Network of Information architecture), which can run as an overlay on top of the existing network infrastructure during its migration phase, borrows DTN's convergence layer approach to extensibility with respect to different and new underlying network technologies. It aims at a global-scale information-centric network based on its hybrid name-based routing and name resolution scheme as well as efficient caching mechanisms.

Serval is an end-host stack for service-centric networking to access diverse services on the Internet, no matter they are large-scale, distributed, ad-hoc, or mobile. The centerpiece of the Serval architecture is a new service access layer that sits on an unmodified network layer to map service names in packets to service-table rules in hosts. The service access layer enables in-stack service-level policy, control, and routing to establish connections via diverse service-discovery techniques, while hiding the addresses and locations of services from applications.

XIA (eXpressive Internet Architecture) is to use eXpressive Internet Protocol (XIP) as a network-layer substrate to enable network innovation and support diverse sets of ideas from other clean-slate designs. Its flexible addressing and forwarding semantics can support multiple and yet unforeseen communication schemes in parallel (e.g., host-, data-/information- and service-centric). XIAs main approach to partial deployment is to allow transitions between architectures at network elements that can understand both, and this bridge between different architectures applies on all levels, from low-level routes to high-level services.

FII (Framework for Internet Innovation) is a "minimal architectural framework" in which comprehensive architectures can reside. It is derived from the simple observation that network interfaces should be extensible and abstract that allows for a diversity of architectures to coexist, communicate, and evolve. FII only defines three core interfaces (or primitives): the interface for communicating between domains (i.e., interdomain routing), the interface between applications and the network (i.e., the network API), and an interface hosts can use to protect themselves against denial-of-service attacks.

OPAE (for Ongoing and Pervasive Architectural Evolution) employs straightforward and well-known principles from systems design: layers of indirection for flexibility, system modularity to limit the impact of changes, and interface extensibility to reduce the pressure on architectural change. The OPAE design is very similar to the FII design, the main difference being the far more general approach to interdomain routing and security in OPAE. OPAE attempts to allow domains using different architectures to directly exchange low-level packets without any disruption in high-level services, so as to foster ongoing and pervasive architectural evolution.

As we can see that there are extensive research works going on to address the Internet's architectural deficiencies on, e.g., handling challenging communication situations, specifically in recent years focusing on "clean slate" redesigns of the network architecture. However, none of these clean-slate architectures have been deployed yet.

5.4.2 Service-Oriented Network Architectures

One key property of services within SOA (Service Oriented Architecture) is that they are *loosely coupled* which is defined as having "no tight transactional properties among the components" [58]. This essential SOA property has been driving the definition of important technologies in the telecommunication field for the last decade. Most of these technologies have focused on the definition of high-level network service interfaces and functionalities, exposing networking platform to upper layer applications for facilitating rapid development of value-added services. Though this exposure is independent of the underlying networking technologies, most of the current specifications, for example the Next Generation Network (NGN) [59], IP-based Multimedia Subsystem (IMS) [60], and the Service Delivery Platform (SDP) [61], still assume IP-based packet switching architecture for the physical network infrastructure [62]. Some works have proposed the adoption of SOA model in the lower network layers, but they have not proved to have the same impact yet [63] except to core network architecture from the Internet community. In the Internet core, SOA is widely used for implementing optical network infrastructure virtualization to enable heterogeneous network architectures (both IP- and non IP-based) to coexist and cooperate in the future Internet.

As it becomes harder and harder to let the Internet integrate new functionalities for fulfilling the demands of evolving (mobile) applications and new (wireless) transport technologies, many research efforts have started viewing SOA-based Internet designs that also involve edge networks and lower network layers, e.g., proposals presented in [64, 65, 66, 67, 68].

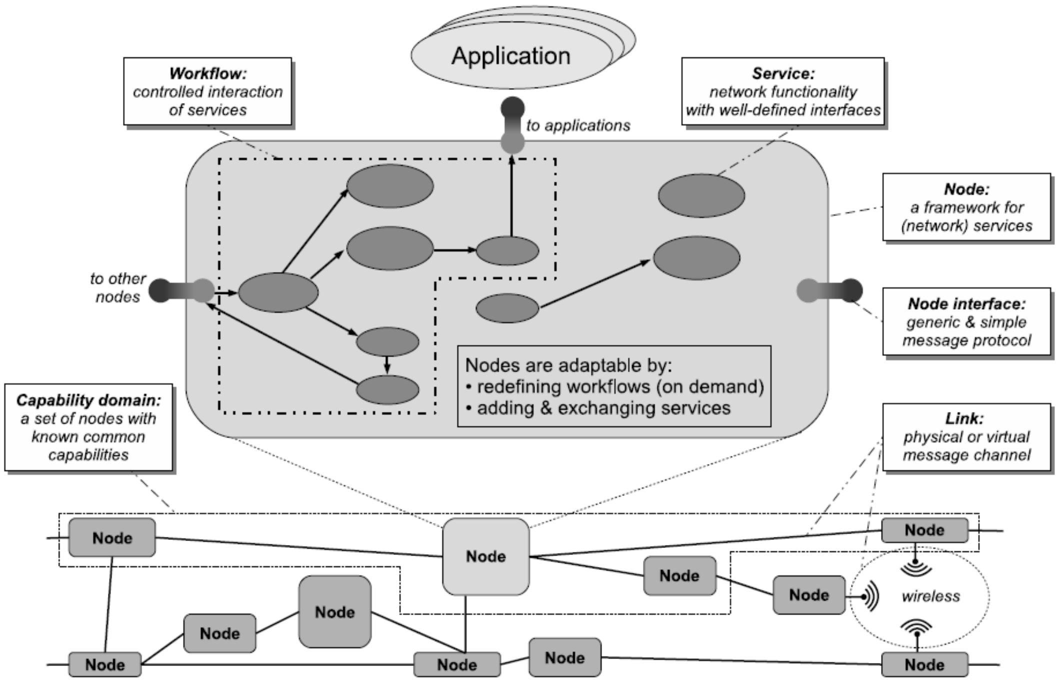

Figure 5.4: A Service-Oriented Approach for a Future Internet Architecture [65].

Figure 5.4 (reprinted from [65]) shows a service-oriented approach to the future Internet architecture, where the Internet is considered as a large, distributed (software) system. The goal is to encapsulate (micro-)protocols by services. Then services offer an abstract view on the functionality of these protocols, in contrast to hiding mechanisms by layers [69].

These Internet architecture approaches represent the direction of combining network functional composition with the widely-used software engineering methodology - SOA - to avoid layering, to pursue high flexibility and loose coupling. A list of research efforts on network functional composition can be found in [70] and figure 5.5 (reprinted from [67]) shows the example of TCP/IP protocol stack composition.

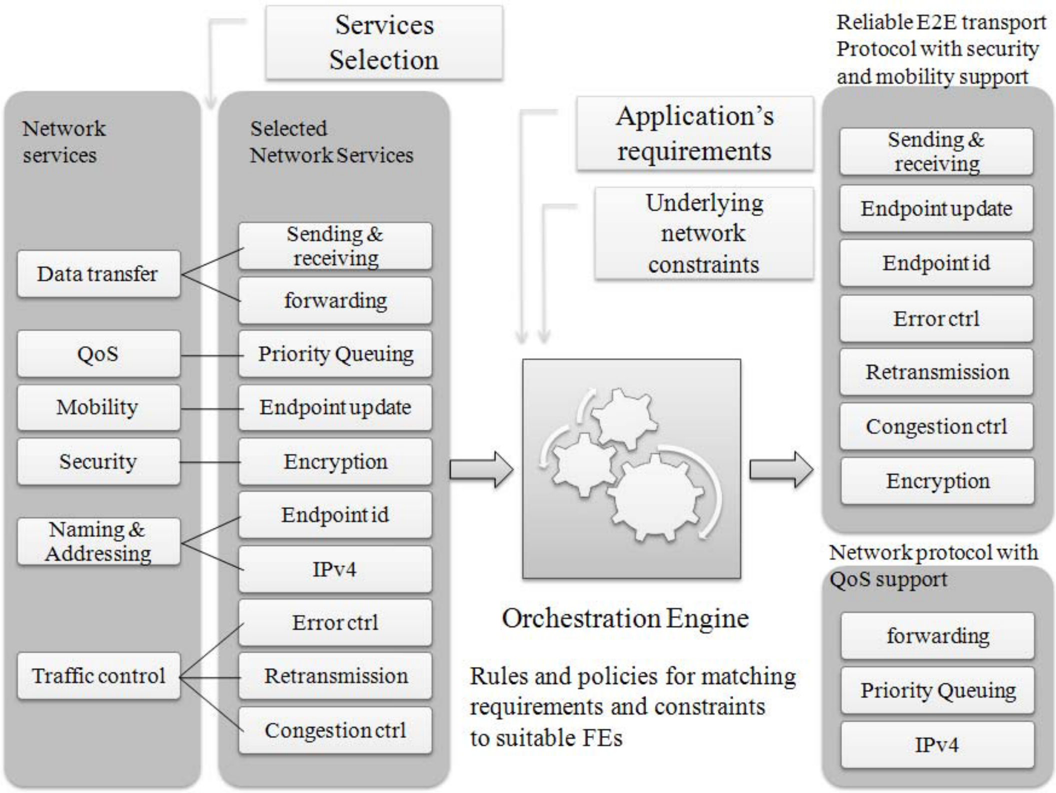

Figure 5.5: TCP/IP Protocol Stack Composition Example [67].

Also, there is a Relationship Oriented Service Architecture (ROSA) [71] proposed in the literature which can be applied to the next generation Internet and beyond. ROSA is standing on an import concept named domains. Domains can be conceived as a region (physical or virtual) characterized by a specific feature and restricted by boundaries, either business entities or physical devices. The relation between two domains (nodes, devices, services) can be characterized by a relationship metrics. The SOA principles are considered in all aspects of design including interfaces between modules and the relationship description, for ensuring the loose-coupling and flexibility.

5.4.3 Incrementally Deploy the Service-Oriented Mediator

Our idea of implementing a service-oriented network mediator consists of three steps as follows:

- Step I encapsulates existing mediation APIs as Web services and provides extra functionalities covering, e.g., security, resource and context management.
- Step II deploys this mediator on specific mobile devices and devices on ships first, applying independent and incremental deployment.
- Step III is to evolve with the trend of service-oriented network architecture.

Compared with the aforementioned related work, our proposal has the following differences:

We do not target at the general Internet architecture at the first place, but focus on solving communication problems in challenged areas such as the maritime environment. Therefore, we prefer incremental changes rather than clean-slate redesigns. Specifically we choose DTN to start with because of its ability to handle network heterogeneity and intermittent connectivity.

We start with DTN but without narrowing ourselves down to this particular solution. DTN has its own implementations and APIs which are quite possible to change in the future. Furthermore, it is likely that another architectural proposal targeting at challenged areas can supplement or replace DTN under some circumstances. Such scenarios dictate the need for an abstraction which can hide the architectural details from applications, a common standardized platform which can provide sufficient flexibility to incorporate diversity under widely accepted guidelines and standards. This observation of embracing diversity via abstraction has already been emphasized in the following proposals: NetAPI [72], FII [56], Tapa [73] and Juno [74]. However, our proposal goes a step further by making the abstraction service-oriented, e.g., to encapsulate existing APIs as Web services. This way it will enable easier development of applications, increase application interoperability and portability via seamless integration with the computing world. Practical examples of the real-world need for richer and more generic APIs can be found in the large number of applications running over HTTP, and the ongoing WebOS, Firefox OS implementations.

Similarly, we agree with the service-oriented Internet architecture proposal and we see the potential that it will thrive in future, while we do not limit our vision only to this solution. Instead, the service-oriented network interface abstraction can allow a variety of architectures to coexist and bridge the networking and service-oriented computing worlds together seamlessly.

It is worth mentioning that our SOA-based network mediator is different from what is described in [75] either. First, we base on existing DTN mechanisms and abstract the mediation functionalities as standard services through an overlay approach to allow incremental and independent deployment. The standard abstraction can also facilitate other networking technologies underneath and embrace application changes. However, the mediator mentioned in [75] limits its role to a functional composition network architecture which has not been deployed yet. Second, our mediator follows the SOA principles, which makes the mediation service no different from other application or network services except the real functionality. In principle, all services can be invoked by each other in a layer-less fashion to allow maximum flexibility, whereas the mediator in [75] is placed in between service and network layers following a three-layered architecture.

5.5 Chapter Summary

In this chapter, we analysed the connection between separate service and communication integration solutions in the context of integrated operations for ships, and identified the inconsistencies in between after we put them together in the maritime scenario. We pointed out that a negotiation or trade-off process can be used to address the inconsistencies and a SOA-based mediator is suggested for implementing the negotiation. We also suggest to deploy the SOA-based network mediator using overlay mechanisms incrementally and independently. This mediator was inspired by the current research work on Internet architecture designs, especially the service-oriented solution and network abstraction proposals. The service-oriented solution is in line with our idea that network resources should be interoperable with and be seamlessly integrated with computing applications where SOA is the main trend. The network abstraction proposals support our thoughts that: 1) it is almost impossible to expect the future Internet to have a single super network architecture which can satisfy all current and predicted requirements and enables innovation, 2) it is possible to grant the Internet with capability to embrace diversity and allow evolvability by generic abstractions. For the maritime area, specifically for realizing integrated operations of ships, a practical way of deploying this mediator is to base on existing technologies and infrastructures instead of expecting or waiting for a future better network architecture to fall into place. The approach should offer both backwards compatibility (e.g., incremental deployment of mediation services) and migration to future possibilities (e.g., standardized interface abstraction).

Chapter 6

Summary, Conclusion, and Outlook

6.1 Summary

This dissertation covers a range of topics on realizing Integrated Operations (IO) for ships, and explores modern information and communication technologies for handling the involved complexity, interoperability and connectivity challenges. Research questions raised in chapter 1 are answered one by one upon reaching the research goals in the following chapters.

- **Answer 1** to *"How to approach complex systems and especially how to solve the concrete task of integrated operations for ships?"*: In order to address complexity, an integration-oriented complex system development method is proposed and applied to IO for ships, where solutions to enhance interoperability and connectivity are investigated separately and the connection in between is examined carefully.

- **Answer 2** to *"How to ensure interoperability among heterogeneous operation applications with isolated, monolithic functionalities and incompatible data formats?"*: Maritime data integration is explored to improve information interoperability and the SOA-based integration solution is assumed to be able to achieve interoperability among heterogeneous applications, along with a maritime ontology to understand the specific domain.

- **Answer 3** to *"How to obtain required connectivity through multiple maritime wireless networks under harsh mobile communication conditions?"*: IP-based optimizations together with a DTN overlay architecture are studied and tailored for the maritime case so as to provide better connectivity in face of challenging communication conditions.

- **Answer 4** to *"If both interoperability and connectivity challenges are addressed independently, does it mean that the goal of integrated operations for ships is achieved? If not, what are the inherent reasons and what are the possible solutions?"*: Although separate interoperability and connectivity solutions can work well individually, the combination of them does not form a seamless solution to the integrated operations for ships because of the encountered conflicts or incompatibilities. Therefore, a negotiation process is proposed by incrementally implementing a SOA-based communication mediator between applications and networks to resolve the inconsistency issue. This proposal also gives hints on the evolution towards future Internet architectures.

The comprehensive study on integrated operations of ships offers an overall view of the topic and yields a whole-picture understanding. Although each subtopic was not studied as deep as many other related works, this overview approach provides us with the opportunity to look at the traditional problems in a systematic way and be able to gain some interesting observations. These observations are especially useful for solving integration problems in the computing and networking worlds when both fields need to interact closely with each other.

6.2 Conclusion

This dissertation is an important starting point for the implementation of integrated operations for ships. It has several main contributions summarized as follows:

- An integration-oriented complex system development approach is proposed for systems which are too complex to be viewed or modelled from one single perspective. This novel approach can be applied to the task of moving towards integrated operations for ships, to address the encountered complexity by approaching the system from multiple perspectives, which requires a careful examination on the connection between the corresponding separate solutions.

- When the integration-oriented approach is utilized, the challenge of dealing with IO for ships is mapped into an integration problem between separate solutions from different perspectives, i.e., maritime service integration and communication integration scenarios. Under each divided task, concrete information and communication technologies can be explored to address smaller problems individually, such as information interoperability, application integration, network connectivity in challenged areas. These individual proposals can be validated by various mechanisms separately, i.e., literature review, theoretical analysis, case studies and simulations, in order to make sure that each proposal is correct within its own context.

- Based on the separate solutions, connection in between them can be examined more clearly and by examining the connection, some inherent problems are interpreted through a different angle and novel ideas can be provided. Application integration based on SOA is recommended to enhance service interoperability for IO of ships and with the maritime ontology to deliver a common understanding of the domain. IP-based optimizations are exploited to improve network connectivity and the DTN architecture as a supplementing policy is suggested to handle extreme maritime communication challenges. When combining the separate interoperability and connectivity approaches together, it appears apparent that the SOA integration methodology should be followed and a similar DTN mechanism will be desired. Therefore, it becomes natural to argue for the SOA-based communication mediator which can also emerge as a promising solution to the future Internet architecture. In addition, the following advantages are achieved:

  - Separate approaches can be studied more comprehensively and deeply from investigating their connection and the usage under the specific maritime domain. An overall solution as a combination of the separate approaches can be derived before any real deployment.
  - By identifying and resolving inconsistencies between solutions that are utilized within different perspectives, the theory of our integration-oriented complex system development proposal was shown as useful at least for this case study on integrated operations of ships.
  - By viewing the issues of the current Internet architecture from looking at the connection between applications and networks, from measuring the connection between related separate solutions, the idea of service-oriented network abstraction can be validated more logically in terms of embracing architecture diversity, encouraging network evolvability and enabling seamless integration to the computing world. The argument for a service-oriented Internet architecture will be considered more convincing as well.

6.3 Further Outlook

This dissertation is an early effort towards integrated operations of ships, covering a vast research area and multiple information and communication technologies are touched. While comprehensiveness is ensured and diverse verification methods are utilized, limitations exist in the following aspects that remain to be studied later.

- Although a comprehensive solution including diverse proposals regarding different challenges is presented with theoretical analysis and computer simulations, prototype implementations are required in order to further evaluate the overall performance.

- Specifically, from the network part, Web services based abstractions need to be provided to cover the network heterogeneity and allow flexibility to handle communication challenges, e.g, intermittent connectivity exhibited in maritime scenarios.

- From the application perspective, new operating services and applications need to be developed based on the service-oriented network abstraction, and existing or legacy systems need to be adapted to the SOA environment via gateway mechanisms or wrappers.

REFERENCES

[1] "Center for Integrated Operations in the Petroleum Industry,", 2007, Available: http://www.iocenter.no/doku.php.

[2] "Integrated Operations in the High North - Design, implement and demonstrate the digital platform for next generation Integrated Operations,", 2008, Available: https://www.posccaesar.org/wiki/iohn.

[3] "Stortingsmelding nr. 38 (2003-2004) Om petroleumsvirksomheten,", 2004, Available: http://www.regjeringen.no.

[4] "Integrated operations,", Available: http://en.wikipedia.org/wiki/Integrated operations.

[5] "MarCom - Broadband at sea (Internet for coast, Polar Regions, offshore and sea farming),", 2007, Available: https://sites.google.com/site/marcomcommunity/about-marcom.

[6] Hind Benbya and Bill Mckelvey, "Toward a complexity theory of information systems development," *Information Technology & People*, vol. 19, no. 1, pp. 12–34, Jan. 2006.

[7] Patrik Haslum, "Reducing Accidental Complexity in Planning Problems," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, 2007, pp. 1898–1903.

[8] Kurt A. Richardson, Paul Cilliers, and Michael Lissack, "Complexity Science: A Gray Science for the Stuff in Between," *Emergence*, vol. 3, no. 2, pp. 6–18, 2001.

[9] Indranil Mukherjee, "The Complexity Paradigm: Implications for Information Systems and their Strategic Planning," *Journal of Computer Science*, vol. 4, no. 5, pp. 382–392, 2008.

[10] P. Cilliers, *Complexity and Postmodernism: Understanding Complex Systems*. Taylor & Francis, 2002.

[11] Douglas C. Schmidt, "Model-Driven Engineering," *IEEE Computer*, vol. 39, no. 2, Feb. 2006.

[12] "OMG Model Driven Architecture - The Architecture of Choice for a Changing World,", 2006, Available: http://www.omg.org/mda.

[13] Dr. Jeff Sutherland, “Agile Development: Lessons Learned From The First Scrum,”, Oct. 2004.

[14] Indranil Mukherjee, “Understanding Information System Failures from the Complexity Perspective,” *Journal of Social Sciences*, vol. 4, no. 4, pp. 308–319, 2008.

[15] Jr. Brooks and Frederick P., “No Silver Bullet Essence and Accidents of Software Engineering,” *Computer*, vol. 20, no. 4, pp. 10–19, Apr. 1987.

[16] Eric Evans, *Domain Driven Design: Tackling Complexity in the Heart of Business Software*. Addison-Wesley Professional, Aug. 2003.

[17] L. M. Haas, E. T. Lin, and M. A. Roth, “Data integration through database federation,” *IBM Syst. J.*, vol. 41, no. 4, pp. 578–596, Oct. 2002.

[18] Patrick Ziegler and Klaus Dittrich, “Data Integration - Problems, Approaches, and Perspectives,”, pp. 39–58, 2007.

[19] Patrick Ziegler and Klaus R. Dittrich, “Three Decades of Data Integration - All Problems Solved?,” in *Proceedings of the 18th IFIP World Computer Congress (WCC 2004)*, 2004, vol. 12, pp. 3–12.

[20] Maurizio Lenzerini, “Data integration: a theoretical perspective,” in *Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2002, pp. 233–246.

[21] Wache H., Vögele T., Visser U., Stuckenschmidt H., Schuster G., Neumann H., and Hübner S., “Ontology -Based Integration of Information - A Survey of Existing Approaches,” in *Workshop on Ontologies and Information Sharing*, 2001, pp. 108–117.

[22] Boualem Benatallah and Hamid R. Motahari Nezhad, “Advances in Software Engineering,”, chapter Service Oriented Architecture: Overview and Directions, pp. 116–130, 2008.

[23] Mike P. Papazoglou and Willem-Jan Heuvel, “Service oriented architectures: approaches, technologies and research issues,” *The VLDB Journal*, vol. 16, no. 3, pp. 389–415, Jul. 2007.

[24] Ralf Klischewski, “Information Integration or Process Integration? How to Achieve Interoperability in Administration,”, in *Electronic Government*, vol. 3183, pp. 57–65, 2004.

[25] M.N. Huhns and M.P. Singh, "Service-Oriented Computing: Key Concepts and Principles," *Internet Computing, IEEE*, vol. 9, no. 1, pp. 75–81, 2005.

[26] Gustavo Alonso, Fabio Casati, Harumi Kuno, and Vijay Machiraju, *Web Services - Concepts, Architectures and Applications*, Oct. 2003.

[27] Michael Carey, "Data delivery in a service-oriented world: the BEA aquaLogic data services platform," in *Proceedings of the ACM SIGMOD international conference on Management of data*, 2006, pp. 695–705.

[28] Roy Thomas Fielding, *Architectural styles and the design of network-based software architectures*, PhD thesis, 2000.

[29] W3C, "Web Services Architecture - W3C Working Group Note,", 2004, Available: http://www.w3.org/TR/ws-arch.

[30] Udi Dahan, "Event-Driven Architecture: SOA Through the Looking Glass," *Microsoft Architecture Journal*, vol. 21, Sept. 2009.

[31] David S. Linthicum, *Next Generation Application Integration: From Simple Information to Web Services*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2003.

[32] David Chappell, *Enterprise Service Bus*. O'Reilly Media, Inc., 2004.

[33] M. Keen, A. Acharya, S. Bishop, A. Hopkins, S. Milinski, C. Nott, C. Robinson, J. Adams, and P. Verschueren, "Patterns: Implementing an SOA Using an Enterprise Service Bus,", Technical report, Jul. 2004.

[34] Mark Endrei, Jenny Ang, Ali Arsanjani, Sook Chua, Philippe Comte, Pœl Krogdahl, Min Luo, and Tony Newling, *Patterns: Service-Oriented Architecture and Web Services*. IBM Redbooks, Jul. 2004.

[35] "International Mobile Satellite Organization,", 2009, Available: http://www.inmarsat.com.

[36] "Very-small-aperture terminal,", 2009, Available: http://www.vsat-systems.com/.

[37] Anja Feldmann, "Internet clean-slate design: what and why?," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 3, pp. 59–64, Jul. 2007.

[38] "IEEE Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services,", 2008.

[39] "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Services and architectures,", 2007.

[40] Jörg Ott, "Delay Tolerance and The Future Internet," in *Proceedings of the 11th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2008.

[41] N.M.M.K. Chowdhury and R. Boutaba, "Network virtualization: state of the art and research challenges," *Communications Magazine, IEEE*, vol. 47, no. 7, pp. 20–26, 2009.

[42] Jinu Kurian and Kamil Sarac, "A survey on the design, applications, and enhancements of application-layer overlay networks," *ACM Comput. Surv.*, vol. 43, no. 1, pp. 5:1–5:34, Dec. 2010.

[43] Kevin Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," in *Proceedings of conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 2003, pp. 27–34.

[44] Khaled A. Harras, Mike P. Wittie, Kevin C. Almeroth, and Elizabeth M. Belding, "ParaNets: A Parallel Network Architecture for Challenged Networks," in *Proceedings of the 8th IEEE Workshop on Mobile Computing Systems and Applications*, 2007, pp. 73–78.

[45] Dipankar Raychaudhuri and Mario Gerla, *Emerging Wireless Technologies and the Future Mobile Internet*. Cambridge University Press, 2011.

[46] Jing Su, James Scott, Pan Hui, Jon Crowcroft, Eyal De Lara, Christophe Diot, Ashvin Goel, Meng How Lim, and Eben Upton, "Haggle: Seamless Networking for Mobile Applications," in *Proceedings of the 9th international conference on Ubiquitous Computing*, 2007, pp. 391–408.

[47] K. Fall and S. Farrell, "DTN: an architectural retrospective," *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 5, pp. 828–836, May 2008.

[48] L Wood, W M Eddy, and P Holliday, "A bundle of problems," *2009 IEEE Aerospace conference*, vol. 40, no. 23, pp. 1–17, 2009.

[49] Anders Lindgren and Pan Hui, "The quest for a killer app for opportunistic and delay tolerant networks: (invited paper)," in *Proceedings of the 4th ACM workshop on Challenged networks*, 2009, pp. 59–66.

[50] Artemios G. Voyiatzis, "A Survey of Delay- and Disruption-Tolerant Networking Applications," *Journal of Internet Engineering*, vol. 5, no. 1, 2012.

[51] Michael Demmer, Kevin Fall, Teemu Koponen, and Scott Shenker, "Towards a Modern Communications API," in *Proceedings of the 6th Workshop on Hot Topics in Networks (HotNets-VI)*, 2007.

[52] Teemu Koponen, Mohit Chawla, Byung-Gon Chun, Andrey Ermolinskiy, Kye Hyun Kim, Scott Shenker, and Ion Stoica, "A data-oriented (and beyond) network architecture," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 181–192, ACM, Aug. 2007.

[53] Christian Dannewitz, Dirk Kutscher, Börje Ohlman, Stephen Farrell, Bengt Ahlgren, and Holger Karl, "Network of Information (NetInf) An information-centric networking architecture," *Computer Communications*, Jan. 2013.

[54] Erik Nordstrom, David Shue, Prem Gopalan, Rob Kiefer, Matvey Arye, Steven Ko, Jennifer Rexford, and Michael J. Freedman, "Serval: An End-Host Stack for Service-Centric Networking,", Technical report, Oct. 2011.

[55] Dongsu Han, Ashok Anand, Fahad Dogar, Boyan Li, Hyeontaek Lim, Michel Machado, Arvind Mukundan, Wenfei Wu, Aditya Akella, David G. Andersen, John W. Byers, Srinivasan Seshan, and Peter Steenkiste, "XIA: Efficient Support for Evolvable Internetworking," in *Proceedings of the 9th USENIX NSDI*, Apr. 2012.

[56] Teemu Koponen, Scott Shenker, Hari Balakrishnan, Nick Feamster, Igor Ganichev, Ali Ghodsi, P. Brighten Godfrey, Nick McKeown, Guru Parulkar, Barath Raghavan, Jennifer Rexford, Somaya Arianfar, and Dmitriy Kuptsov, "Architecting for innovation," *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 3, pp. 24–36, ACM, Jul. 2011.

[57] Ali Ghodsi, Scott Shenker, Teemu Koponen, Ankit Singla, Barath Raghavan, and James Wilcox, "Intelligent design enables architectural evolution," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, 2011, pp. 3:1–3:6.

[58] Munindar P. Singh and Michael N. Huhns, *Service-Oriented Computing: Semantics, Processes, Agents*. John Wiley & Sons Ltd, 2005.

[59] K. Knightson, N. Morita, and T. Towle, "NGN architecture: generic principles, functional architecture, and implementation," *Communications Magazine, IEEE*, vol. 43, no. 10, pp. 49–56, Oct. 2005.

[60] Antonio Cuevas, J.I. Moreno, P. Vidales, and H. Einsiedler, "The IMS service platform: a solution for next-generation network operators to be more than bit pipes," *Communications Magazine, IEEE*, vol. 44, no. 8, pp. 75–81, Aug. 2006.

[61] H. Ohnishi, Y. Yamato, M. Kaneko, T. Moriya, M. Hirano, and H. Sunaga, "Service Delivery Platform for Telecom-Enterprise-Internet Combined Services," in *Proceedings of IEEE Global Telecommunications Conference*, Nov. 2007, pp. 108–112.

[62] Qiang Duan, Yuhong Yan, and A.V. Vasilakos, "A Survey on Service-Oriented Network Virtualization Toward Convergence of Networking and Cloud Computing," *IEEE Transactions on Network and Service Management*, vol. 9, no. 4, pp. 373–392, 2012.

[63] G. Branca and L. Atzori, "A Survey of SOA Technologies in NGN Network Architectures," *Communications Surveys Tutorials, IEEE*, vol. 14, no. 3, pp. 644–661, 2012.

[64] Bernd Reuther and Joachim Götze, "An Approach for an Evolvable Future Internet Architecture," in *Proceeding of Workshop on New Trends in Service & Networking Architectures*, 2008.

[65] Reuther Bernd and Müller Paul, "Future Internet Architecture - A Service Oriented Approach," *it - Information Technology*, vol. 50, no. 6, pp. 383–389, 2008.

[66] Dennis Schwerdel, Abbas Siddiqui, Bernd Reuther, and Paul Müller, "Composition of Self Descriptive Protocols for Future Network Architectures," in *Proceedings of the 35th Euromicro Conference on Software Engineering and Advanced Applications*, 2009, pp. 571–577.

[67] A. Karouia, R. Langar, Thi-Mai-Trang Nguyen, and G. Pujolle, "SOA-Based Approach for the Design of the Future Internet," in *Proceedings of the 8th Annual Communication Networks and Services Research Conference (CNSR)*, 2010, pp. 361–368.

[68] Alberto J. Gonzalez, Ramon Martin de Pozuelo, Martin German, JesusAlcober, and Francesc Pinyol, "New Framework and Mechanisms of Context-Aware Service Composition in the Future Internet,", vol. 35, no. 1, pp. 7–17, Feb. 2013.

[69] John Day, *Patterns in Network Architecture: A Return to Fundamentals*. Prentice Hall, 2008.

[70] Christian Henke, Abbas Siddiqui, and Rahamatullah Khondoker, "Network functional composition: State of the art," in *Proceedings of Telecommunication Networks and Applications Conference (ATNAC)*, 2010, pp. 43–48.

[71] Kumar Ram, *A service based approach for future internet architectures*, PhD thesis, 2011.

[72] Ganesh Ananthanarayanan, Kurtis Heimerl, Matei Zaharia, Michael Demmer, Teemu Koponen, Arsalan Tavakoli, Scott Shenker, and Ion Stoica, "Enabling Innovation Below the Communication API,", Technical Report UCB/EECS-2009-141, Oct. 2009.

[73] Fahad R. Dogar and Peter Steenkiste, "Architecting for edge diversity: supporting rich services over an unbundled transport," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, 2012, pp. 13–24.

[74] Gareth Tyson, Andreas Mauthe, Sebastian Kaune, Paul Grace, Adel Taweel, and Thomas Plagemann, "Juno: A Middleware Platform for Supporting Delivery-Centric Applications," *ACM Trans. Internet Technol.*, vol. 12, no. 2, pp. 4:1–4:28, Dec. 2012.

[75] Abbas Ali Siddiqui, Christian Henke, Daniel Günther, Paul Müller, and Thomas Magedanz, "Mediation between Service and Network Composition,", Workshop on IP: Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks", 2010.

Part II

Appendix A

Paper A - Complex System Development

Title:	Towards Integration-Oriented Complex System Development
Authors:	Liping Mu, Andreas Prinz and Carl Erik Moe
Affiliation:	University of Agder, Norway
Conference:	*Norsk konferanse for organisasjoners bruk av informasjonsteknologi (NOKOBIT)*, Tromsø, Norway, 21-23 November 2011
Copyright ©:	Academy Publisher

Towards Integration-Oriented Complex Systems Development

Liping Mu, Andreas Prinz and Carl Erik Moe
University of Agder (UiA), Norway
E-mails: {liping.mu, andreas.prinz, carl.e.moe}@uia.no

Abstract — **Modern information systems tend to be large-scale and complex, and it has been observed that traditional information system development approaches have problems handling very large systems. Efforts have been concentrated on addressing these problems from the solution domain, but less on the complexity of the problem itself. This paper attempts to uncover the reasons for these problems by looking closer at system complexity, the causes of complexity and the approaches that have been used to handle complexity. Based on the analysis and a literature review, we suggest that some major problems of very large complex system development can be solved by treating the development process as an integration task from a multi-perspective angle.**

Keywords—Information System (IS), Information System Development (ISD), Complexity, Integration.

I. INTRODUCTION

An Information System (IS) is a system that holds data and transforms them into information and knowledge for supporting processes and communication within an organization. It is usually a combination of Information Technologies (IT), people, processes, data and their interactions. Modern information systems are mostly software-intensive, in which software represents a significant segment in system functionality and development. Information System Development (ISD) is viewed as an organizational change project that aims at improving an organization through the adoption of IT applications. Information systems play important roles in modern society. Most of the organizations today are information driven and information sharing is often done by the interconnection of various types of information systems. According to [1], organizations are increasingly developing information ecosystems where the information systems, databases, workflows, people and infrastructure are integrated to build, maintain and dynamically update the myriad views from different organizational perspectives. Both the size and complexity of modern information systems have significantly increased.

As the size and complexity of an IS keep increasing, so does the complexity of the ISD activity. Benbya et al. have shown that the complexity of ISD comes from two disciplinary areas [2]: 1) the area of the problem being solved (the problem domain) and 2) the area of constructing a software solution (the solution or implementation domain). Therefore, in this paper, we consider two kinds of complexity: the complexity in the problem domain focusing on the essential complexity, and the complexity in the solution domain focusing on the accidental complexity.

Traditional system development approaches focused on addressing the complexity that arises from the solution domain - accidental complexity. However, as modern information systems are becoming highly essentially complex in the problem domain, the traditional way of addressing accidental complexity is not enough anymore and the complexity of ISD deserves further investigation as also pointed out by Lee et al. [3]. It has been recognized that essential complexity of ISD needs to be handled as well, but most current solutions only tackle essential complexity partially.

In this paper, we use the methodology of literature review to show that traditional system development approaches have mainly focused on addressing accidental complexity that arises from the solution domain, which is not enough for developing modern complex software systems with high essential complexity. To develop complex systems like a modern IS in a traditional way often introduces "bootstrap" and "adaptability" problems. We show that current solutions have started tackling essential complexity but concentrate their efforts on handling adaptability, while the bootstrap problem remains unsolved. The bootstrap problem is introduced by the traditional way of developing an incompressible complex system. After analyzing the reasons for the problems of developing a modern IS, we propose our idea on complex systems development: to treat the whole development process as an integration journey and use ontology to guide this process, namely the integration-oriented complex systems development.

II. Accidental Complexity in IS

Traditional system development approaches have focused on addressing accidental complexity in the solution domain. We explain this from the technical and the social aspects. By technical aspect, we mean the technologies that have been used for constructing the target system. By social aspect, we consider it as the management factor connected to the development activity/project being performed by people. It is worth mentioning here that our way of categorizing the accidental complexity of systems development in the solution domain is different from the view of seeing an ISD activity from the technical, organizational and societal perspectives as for example in [4]. We focus on differentiating between the computer technology part and the human management part in the process of constructing a software solution for illustration purposes, see figure A.1:

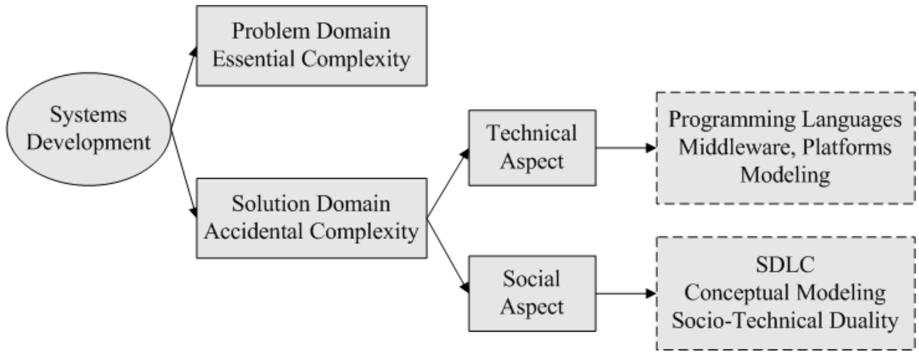

Figure A.1: Traditional System Development Approaches for Handling Accidental Complexity.

A. Technical Aspect

Many papers have mentioned that a significant factor behind the difficulty of developing software systems is the wide conceptual gap between the problem and the implementation domains [5, 6], the difficulty of representing problems in programming languages and mapping them to the target platform. In this sense, accidental complexity of software systems development has been well recognized, and it was usually addressed by lifting the implementation abstraction level. For example, early programming languages and early operating system platforms have shielded software developers from the complexity of the underlying computing environment dramatically. Further advances in languages and platforms during the past two decades have raised the abstraction to a higher level, with object-oriented languages like C++ and Java, reusable libraries and middleware platforms.

The introduction of technologies that effectively raise the implementation abstraction level can significantly improve productivity and quality of the targeted software system, and open the door to new software opportunities that are acted upon. However, the result is often a new generation of more complex systems and associated software development concerns [5].

In addition, modeling has been widely used for addressing accidental complexity. In the modeling world, a model is a formal specification of the structure, function or behavior of a system. Models can be used for system specification, enhancing understanding, system validation and driving implementations. A model of a system that rests upon the problem domain is a simple version of the target system that has abstracted away all its accidental complexity of implementation. In this sense, modeling handles accidental complexity by avoiding it in the first place.

B. Social Aspect

Technology has effectively alleviated the accidental complexity of software systems development, but it leads to larger and more complex systems with more difficult development and management work. There is a famous Chinese proverb saying: one monk will shoulder two buckets of water, two monks will share the load, but add a third and no one will want to fetch water. It tells us that the human or social factor is often the major reason for increased accidental complexity of group activities like large software system development projects. Therefore, methodologies and processes need to be used in order to manage this kind of accidental complexity.

An engineering focused approach such as Systems Development Life Cycle (SDLC) is a systematic procedure to develop a software system through sequential stages. SDLC has the objective of ensuring that high quality systems are delivered by providing strong management controls over the projects and maximizing the productivity of the systems staff.

Likewise, conceptual modeling, the activity of formally describing some aspects of the physical and social world, is done to help with the common understanding and communication between users and developers during an ISD project. The major reasons for the failure of ISD projects related to requirements specification described in [7] include the lack of common understanding between people involved in the development. Therefore, conceptual modeling can help decrease the failure rate of ISD projects by enhancing the common understanding among developers.

ISD, which is created and embedded in the collective actions found in different communities of practice, can be viewed as a process of social construction of knowledge. The macro-phenomena in ISD are the result of the micro-interactions among participants and artifacts used in the social process. The exploration of possible causes of ISD failures has ranged from mere technological explanations toward a rich and complex human organizational analysis, leading to more socio-technical based ISD approaches. Vidgen et al. have pointed out that socio-technical approaches of ISD seek to engage the "user" of the information in genuine participation to achieve an acceptable fit between people and technology rather than forcing one or both to change and adapt to the other [8].

III. Essential Complexity in IS

A. Complex Systems

There are various descriptions of a complex system from the literature, e.g., the one in [9]. Building on these many definitions, we define a complex system as a constantly evolving system that is made up of a large number of nonlinearly interacting parts which will make the overview behavior changeable and unpredictable. Given the systems that we have developed and which are continuing to evolve in all cases of life, it is clear that modern information systems are becoming complex systems. They are large-scale, distributed and complex. Their complexity can be further divided into incompressibility and changeability.

Large-Scale: Modern systems have increasing scale in some or all of these dimensions: the number of subsystems and components, the amount of data to be stored, accessed, manipulated, and refined, the number of connections and interdependencies, the number of system purposes, and users perceptions of these purposes (the number of disparate stakeholders).

Distributed: As communication and Internet technologies keep evolving, systems are becoming much more decentralized and geographically distributed than before. When we deploy systems today, the development effort concentrates more on integrating existing components or systems than on building new ones. These existing and new components or systems do not need to be at the same place nor have the same underlying infrastructures or operating systems.

Complex:

- Incompressibility - as a whole, the structure and behavior of a modern complex system cannot be easily described by one single rule, nor may they be inferred from the structure and behavior of its parts. The system and its parts sometimes have emergent behavior which cannot be predicted from their current states and specifications. These characteristics introduce the incompressibility of a complex system. Cilliers [10] has explained that incompressibility means that there is no accurate (or rather, perfect) representation of the system which is simpler than the system itself.

- Changeability - The developing process of a complex system often involves the integration of existing systems or components and adding multiple functions or new capabilities that have not been anticipated before. Therefore, the system as a whole will have dynamic behaviors and functions. Meanwhile, the heterogeneous elements of a complex system evolve and change individually which influences the whole system. These features explain the changeability of a complex system.

B. Modern Complex Information Systems

1) Incompressibility - Bootstrap Problem

An IS is both a technological system and a social one, and the interaction between them. It often spreads to all aspects and parts of an organization or among organizations of a company. A modern company has usually more than one organization: corporate management, sales, marketing, engineering, product development, IT infrastructure, etc. They are connected by the company's IS to share knowledge and information. Each organization has its own view of the IS and its own use of the data stored in the databases. All these different views of the same IS make the development of it a very complex process that often requires a reference discipline. Seeing ISD as a reference discipline demands different theories, including economics, mathematics, linguistics, semiotics, ethics, political science, psychology, sociology and statistics, along with computer science [11].

Even though modern information systems are becoming more and more complex and the ISD process should be considered as a reference discipline touching multiple theories, practical large information system development is still very often a top-down process following the sequential SDLC mode, attempting to define a model of the target system covering all views of it during system conceptualization. After the conceptualization step, a complete abstract version of the system should be developed and all assumptions are validated. It is worth mentioning here that the non-sequential characteristic and the flexibility of Agile Software Development [12] have made it quite popular for modern software systems development, but as Agile methods meet many impediments for large complex systems development, they are very often only efficient for small projects. Therefore, given that modern information systems are complex systems which are incompressible [13], and modeling/abstracting a complex system is to compress it, it is not difficult to see that finishing the first conceptualization step of ISD is nearly impossible if one follows the top-down approach. This defines the bootstrap problem of modern ISD.

2) Changeability - Adaptability Problem

Apart from incompressibility, a complex system exhibits inherent changeability as well, which necessitates the developed system to be adaptable. A modern IS - as a large socio-technical entity has a high degree of complexity arising out of the intricate combination of technical, human and organizational characteristics. The relationships among those different factors are non-linear, which gives rise to emergent behaviors of the system as a whole, making the behavior prediction of the whole system very difficult. Also these different factors are changing over time, exacerbating the essential changeability of the entire IS and the interoperability problem between different components of the IS. The changeable behaviors are difficult or even impossible to predict, which explains the adaptability problem of modern ISD.

IV. Current Approaches to Address Essential Complexity

A. Concentrated on changeability feature - adaptability problem

Current system development approaches have started to pay attention to handling essential complexity, but concentrate their efforts mainly on addressing the changeability feature in order to solve the adaptability problem. We explain this from the technical and social aspects as well, see figure A.2:

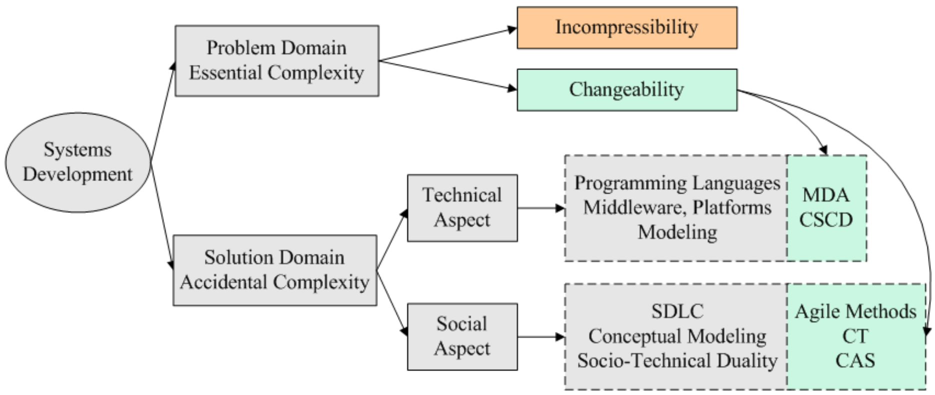

Figure A.2: Current System Development Approaches for Handling Changeability.

1) Technical aspect

Software systems development based on the modeling technique is evolving from model-based approaches into model-driven ones. Stahl et al. [14] have explained that in the model-driven development field, models do not only constitute documentation, but are considered equal to code, as their implementation is automated. The automation of the implementation process can push accidental complexity handling a step further by liberating human labor and eliminating human uncertainty factors.

OMG's Model Driven Architecture (MDA) [15] is a famous instance of the model-driven development methodology. It uses models to describe complex systems at multiple levels of abstraction from a variety of perspectives, and uses standard tools to automatically transform and analyze these models. It follows the traditional focus of addressing accidental complexity by elevating the implementation abstraction level, but with its significant vision of lifting the abstraction level up to the problem domain. Therefore, accidental complexity can be handled more efficiently or even be avoided.

Furthermore, MDA provides an open, vendor-neutral way to deal with the challenge of business and technology change. MDA separates business and application logic from underlying platform technology by three important concepts: Computation-Independent Models (CIM), Platform-Independent Models (PIM) and Platform-Specific Models (PSM). CIM captures the business view from the stakeholders' perspective. PIM is the description of the target system without implementation details for a specific computing system. PSM defines the necessary details for the specific computing platform in which the target system will run. By decoupling with each other, business and technical aspects of a software system can evolve at their own pace. Therefore, MDA can be considered as well designed for handling changeability from both problem and solution domains.

Similarly, the Conceptual Schema-Centric Development (CSCD) approach proposed in [16] is designed to tackle the fast evolution problem in the ISD field. CSCD treats conceptual schema - the general knowledge about an IS's domain and about the functions it has to perform - as the center of the development process. The conceptual schema of CSCD is like a PIM in MDA. When using UML as the conceptual modeling language, CSCD can be considered as the MDA version for ISD, with a broader vision of including not only the initial development of an IS, but also its evolution.

2) Social aspect

Because of the changeability of modern complex systems, SDLC has evolved from the initial well-known waterfall linear mode to non-linear ones, like the iterative mode. The iterative mode attempts to provide more ease-of-change during the development process, e.g. rapid prototyping - after a quick requirements gathering phase, a prototype application is built and presented to the application users. Feedback from the user provides a loop to improve or add functionality to the application. Another example is the incremental mode which combines both linear and iterative modes. It breaks a project into smaller segments and provides more ease-of-change during the development process. Agile software development is a group of software development methodologies that are based on both iterative and incremental modes. There are different agile methods focused on different aspects of the software development life-cycle, in which, Scrum is widely used for managing the software projects [17]. Existing Agile methods are quite efficient to address the changeability of both requirements and solutions but show limitations for large complex system development as explained in [18].

Jacucci et al. [4] have pointed out that modern ISD as a complex activity has to deal with changes from a number of perspectives: 1) from a technical perspective, the evolution of computing and communication technologies, the increased number of systems and applications, and the increased inter-connection and dependencies among them, 2) from an organizational perspective, the adaptation of structural and operational systems to the rapidly changing markets and workforce diversity, and 3) from a societal perspective, the increased interdependent organizational activities and social relations across geographical and organizational boundaries.

Therefore, Benbya et al. [2] have proposed to use Complexity theory (CT) as a frame of reference to analyze its implications on ISD. CT is a new way of thinking about systems and systems' behaviors based on the idea that order emerges through the interactions of organisms or agents. Having the same theoretical background as CT, the Complex Adaptive Systems (CAS) theory is more often used in ISD projects to cope with complexity and changes. Mukherjee has defined in [19]: a complex adaptive system denotes an open ended system of many heterogeneous agents which interact nonlinearly over time with each other and their environment and which are capable of adapting their behaviors based on experience. If ISD projects can evolve like CAS, such as the co-evolutionary software development process (see [20], [2], [21]), then they should be able to evolve themselves to adapt to the changing environments. Even though, CT and CAS have not been prominently applied in the ISD context, they changed the way in which we view ISD and introduced complexity thinking in IS.

B. The bootstrap problem remains unsolved

On one hand, modern information systems are becoming complex systems which are incompressible. It means that if a to-be-constructed model of a complex IS wants to capture all the possible behaviors contained by the system being represented, then that model must be at least as complex as the system itself. In other words, the best representation of a complex IS is the system itself, and any representation or abstraction of the system will be incomplete and can lead to an incomplete or even wrong understanding. On the other hand, system development approaches have focused on addressing accidental complexity and the changeability, while less attention has been paid to the initiation of large complex ISD. For large complex information systems, the development still follows the traditional top-down approach with gradually using CT/CAS theory and Agile methods to cope with non-linear behaviors and changes, therefore the bootstrap problem remains unsolved.

V. Integration-Oriented Complex Systems Development

Although modern information systems are incompressible, this is not an excuse for not bothering [22]. Although there is no single representation that can completely capture the inherent complexity of a complex system and multiple representations may lead to conflicts, that does not mean that incomplete representations are useless. These obstacles of complex systems development encourage us to explore. If not any single abstract version of the target system can capture the complete essence of it, we can capture the system by different abstractions and from a number of perspectives using multiple methods: pluralism. The system development process therefore becomes an integration journey. If inconsistency and conflicts among these different abstractions appear, a negotiation or trade-off is needed. A common ontology among them is highly encouraged which can be used to guide the whole integration journey. We explain this integration-oriented complex systems development from the system perspective and the system development activity perspective.

A. From the system development activity perspective - Pluralism

Pluralism for complex systems development means to approach them from many different directions using multiple methods. Multiple methods can lead to the incommensurability problem between them. Apart from that, different models of the target system based on different perspectives can have inconsistencies or conflicts. Despite these difficulties, the attractiveness of pluralism and multi-method research, in terms of the richness and increased validity of the results, is expected to grow in the long run [23]. Besides, if both the negotiation between different methods and the connection between different models can be carefully taken care of, pluralism for large complex system development will become easier. In [23], it was also claimed that applying various systems development methods at different administrative levels of the business organization can accelerate the ISD process and change management during the systems maintenance. The proposed approach is to view an IS as a fractal enterprise that organizes itself towards well negotiated goals allowing suitable local development approaches in different parts of the system.

B. From the system perspective - Ontology-based Integration

The fractal approach for ISD proposed in [24] is related to the idea of decomposing the system into different subsystems, namely the “divide and conquer” methodology. However, the incompressibility of complex systems impedes us to decompose the target system into subsystems or parts, and even if we manage to do so and can use pluralism, the inconsistency problem of these different parts still increases the total difficulty.

There is an example in the Telecom industry: a communication system is expected to have three main functions in terms of service delivery: quality, security and mobility. Therefore, the development group was divided into three subgroups according to these three objectives: one took care of quality, another was responsible for the security and the third was to handle mobility. At the end, each group had addressed its own problem perfectly, but after integrating these three parts together, the whole system worked terribly poor. Therefore, dependencies and negotiation among components have to be particularly considered very early and be well taken care of during the whole integration process.

In the fractal information systems development approach, this is done by the fractal enterprise idea: the enterprise develops and maintains a core information architecture ontologically shared by all fractal entities. Applied to the service-oriented paradigm, the core architecture may serve as "information bus", where all fractal entities may "plug in" for negotiation and realization of the common system goals.

According to [25], ontology is an explicit formal sharing conceptualization of one domain, and it can advance the understanding, management and integration of a system development process. Therefore, ontology is expected to be a chosen negotiation tool. Particularly for ISD, ontology can act as an information integrator, as it is more concerned with the relationships among entities than with the entities themselves, and the semantics of these relationships can be consistently observed in the ontology [26]. Therefore, ontology tends be an ideal integrator for complex systems development.

C. A practical example: ontology-based data integration

A practical example of the ontology-based approach for data integration problem has been described in [27]. This example is in line with our suggested integration-oriented complex systems development methodology. It focuses on solving the problem of vast human interaction involved in exchange of data and facilitating the information interoperability between multi domains in the oil & gas industry. Modern oil & gas industry is to a large extent a knowledge and information domain and it relies heavily on a large amount of data residing in multiple heterogeneous data sources (may belong to different information systems). In order to present these data in a unified view to users to achieve the transparent manipulation of information, it is important to develop an integrated IS that can benefit much from the ontology-based data integration approach. Once we have translated the complex system development problem into an integration task, the assessment of different methodologies of complex system development can be as well achieved by comparison of different models of the integration solution. There are three well-known models of data integration: warehousing, federation and mediation, see figure A.3.

In warehousing, data must be extracted, transformed and loaded from remote sources to a local central repository named "data warehouse". The central repository provides a single access point to a collection of data copied from heterogeneous

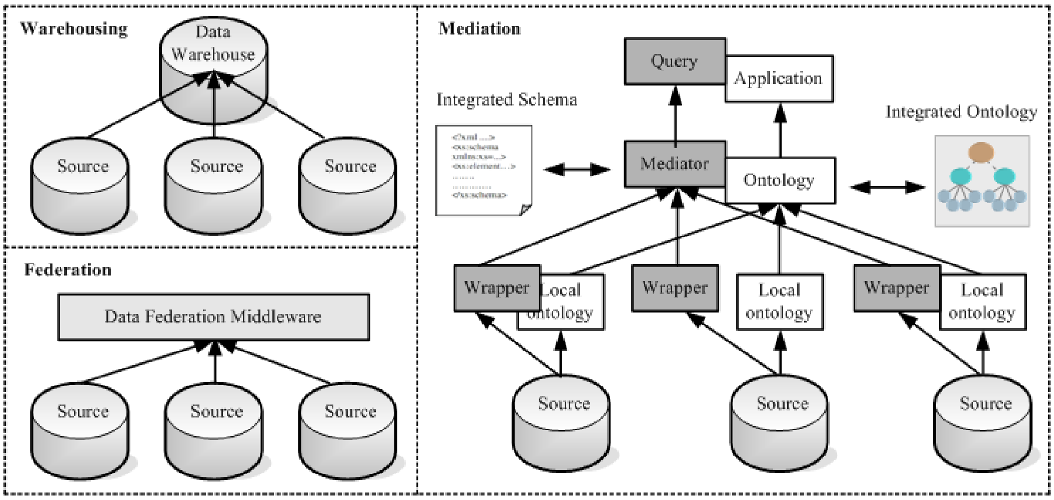

Figure A.3: Three Models of Data Integration.

sources. This model is very popular for achieving data integration so far, but it is not appropriate for large complex ISD, because it tries to design a complete global system model before any implementation processes, and this introduces the bootstrap problem.

In federation, database systems are distributed and independent, preserving database autonomy. Database systems can communicate with each other directly and data can be retrieved via a middleware component. This approach complies with the "divide and conquer" policy, with less attention to the integration goal which may lead to the inconsistency problem for large complex ISD. Again, conceptually a common global data model is required in federation which introduces the bootstrap problem.

In mediation, there is a mediator who does not store any data on its own, but rather provides a virtual view of the integrated sources. Wrappers are often used to translate data access and manipulation requests between the mediator and data sources. The mediator splits a user query into sub-queries, sends the sub-queries to appropriate wrappers and integrates the query results locally. Therefore, the mediation model, which accents both decomposition and integration tasks, is usually more flexible than the other two for large systems development. It can deal better with autonomous and frequently changing data sources. Also, different local methods can be used to develop and maintain local data models in different data sources.

VI. Conclusions

In this work we have pointed out that the traditional system development approaches focused on addressing accidental complexity, whereas modern systems have increasing essential complexity. Although work has been done to address the essential complexity, most of them concentrated on addressing changeability, and incompressibility has been neglected. This is critical for large ISD, when they are developed in a top-down manner. We have therefore argued that complex systems development should be treated as an integration task from multiple perspectives and the common ontology has to be taken care of to guide the whole integration process. We believe that if we change the way of looking at a modern complex software system and its development and pay more attention to integration, the difficulties of large complex system development can be alleviated. However, future work is necessary 1) to justify our argument line from enough practical examples and 2) to evaluate our proposal for large complex systems development.

REFERENCES

[1] M.L. Brodie, "Data Integration at Scale: From Relational Data Integration to Information Ecosystems," In *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications,* 2010, pp. 2-3.

[2] H. Benbya, and B. McKelvey, "Toward a complexity theory of information systems development", *Information Technology & People,* Vol. 19, Issue 1, pp.12-34, 2006.

[3] G. Lee, and W. Xia, "Development of a Measure to Assess the Complexity of Information Systems Development Projects," In *Proceedings of the 23rd International Conference on Information Systems,* 2002.

[4] E. Jacucci, O. Hanseth and K. Lyytinen, "Introduction: Taking complexity seriously in IS research," *Information Technology & People,* Vol. 19, Issue 1, pp. 5-11, 2006.

[5] R. France and B. Rumpe, "Model-driven Development of Complex Software: A Research Roadmap," In *Proceedings of Future of Software Engineering (FOSE),* 2007, pp. 37-54.

[6] D. C. Schmidt. "Model-Driven Engineering,". *IEEE Computer,* Vol. 39, Issue 2, pp. 25-31, 2006.

[7] A. Gemino and Y. Wand, "A framework for empirical evaluation of conceptual modeling techniques," *Requirements Engineering,* Vol. 9, Issue 4, pp. 248-260, 2004.

[8] R. Vidgen and S. Madsen, "Exploring the Socio-Technical Dimension of Information System Development: Use Cases and Job Satisfaction," In *Proceedings of the European Conference on Information Systems,* 2003.

[9] I. Mukherjee "The Complexity Paradigm: Implications for Information Systems and their Strategic Planning," *Journal of Computer Science,* Vol 4, Issue 5, pp. 382-392, 2008.

[10] P. Cilliers, Complexity and Postmodernism: Understanding Complex Systems, London: Routledge, 1998.

[11] D. Avison and S. Elliot, "Scoping the discipline of information systems" *Information Systems: The State of the Field,* John Wiley and Sons, 2006.

[12] Manifesto for Agile Software Development, http://agilemanifesto.org/.

[13] K. A. Richardson, P. Cilliers and M. Lissack, "Complexity Science: A "Gray" science for the "stuff in between"," *Emergence,* Vol. 3, Issue 2, pp. 6-18, 2001.

[14] T. Stahl, M. V´olter and K. Czarnecki, Model-Driven Software Development: technology, engineering, management, John Wiley, 2006.

[15] OMG Model Driven Architecture, http://www.omg.org/mda/.

[16] A. Olivé, "Conceptual Schema-Centric Development: A Grand Challenge for Information Systems Research," *Computer Science,* Vol. 3520, pp.1-15, 2005.

[17] K. Schwaber and M. Beedle, Agile software development with SCRUM, Prentice Hall, 2002.

[18] R. Crocker, Large Scale Agile Software Development, Addison-Wesley, 2004.

[19] I, Mukherjee, "Understanding Information System Failures from the Complexity Perspective," *Journal of Social Sciences,* Vol. 4, Issue 4, pp. 308-319, 2008.

[20] J. A. Highsmith, Adaptive Software Development - A Collaborative Approach to Managing Complex Systems, Dorset House, 2000.

[21] R. M. Kim, S. M. Kaplan, "Interpreting socio-technical co-evolution: Applying complex adaptive systems to IS engagement," *Information Technology & People,* Vol. 19, Issue 1, pp.35-54, 2006.

[22] K. A. Richardson, "Managing complex organizations: Complexity thinking and the science and art of management," *Computer and Information Science,* Vol. 10, Issue 2, pp. 13-26, 2008.

[23] J. Mingers, "The paucity of multimethod research: a review of the information systems literature," *Information Systems Journal,* Vol. 13, Issue 3, pp. 233-249, 2003.

[24] M. Kirikova, "Towards Flexible Information Architecture for Fractal Information Systems," In *Proceedings of the International Conference on Information, Process, and Knowledge Management,* 2009, pp. 135-140.

[25] T. R. Gruber, "A translation approach to portable ontology specifications," *knowledge acquisition,* Vol. 5, pp. 199-220, 1993.

[26] I.E.A. Abugessaisa and A. Sivertun, "Ontological approach to modeling information systems," In *Proceedings of the 4th International Conference on Computer and Information Technology,* 2004, pp. 1122-1127.

[27] L. Mu, K. Li and A. Prinz, "Maritime data integration using standard ISO 15926," In *Proceedings of the 20th International Offshore and Polar Engineering Conference,* 2010.

Appendix B

Paper B - Maritime Data Integration

Title:	Maritime Data Integration Using Standard ISO 15926
Authors:	Liping Mu, Li Kuang and Andreas Prinz
Affiliation:	Department of Information and Communication Technology, University of Agder, N-4898 Grimstad, Norway
Conference:	*The 20th International Offshore and Polar Engineering Conference (ISOPE)*, Beijing, China, 20-26 June 2010
Copyright ©:	Academy Publisher

Maritime Data Integration Using Standard ISO 15926

Liping Mu, Li Kuang and Andreas Prinz
Department of Information and Communication Technology,
University of Agder (UiA), N-4898 Grimstad, Norway
E-mails: {liping.mu, andreas.prinz}@uia.no

Abstract — **The Oil & Gas industry is moving forward with Integrated Operations (IO). This includes a strong focus on data integration issues. There are different ways to achieve data integration, and ontology-based approaches have drawn much attention. For maritime data integration, the international standard ISO 15926 has been developed to integrate and represent the information in and across process plants. In this article, we follow the ontology-based approach and use the ISO 15926 standard as well as Semantic Web technologies to implement maritime data integration, to see how well they fit together and prove the benefits they can bring for the next generation of Integrated Operations.**

Keywords—Data Integration, Ontology-Based Integration, ISO 15926, Semantic Web, Oil & Gas Ontology, Integrated Operations.

I. INTRODUCTION

Modern Oil & Gas industry is to a large extent a knowledge and information industry. It is moving forward with Integrated Operations (IO). The Norwegian Oil Industry Association (OLF) has defined the term IO as “real time data onshore from offshore fields and new integrated work processes. It was mentioned in [1] that IO consists of collaborative efforts in the Oil & Gas industry to support operational decisions about offshore installations by onshore control centers, developing common standards, integrated solutions, and new technologies. The first generation of Integrated Operations (IO G1) has integrated processes and people offshore and onshore, improving the ability to support offshore operations from onshore centers. The second generation of Integrated Operations (IO G2) aims for heavily instrumented facilities, heavy automation and multi-domain optimization of processes. Thus, to solve the problem of vast human interaction involved in exchange of data and facilitating the information interoperability between multi domains will be difficult. In other words, data integration will still be a significant topic during the second generation of Integrated Operations.

Within the Oil & Gas industry, data integration, also known as information integration, involves gathering and combining data residing in multiple heterogeneous data sources (like different offshore databases) and presenting these data in a unified view to users (like onshore control centers) in order to achieve the transparent manipulation of information. How to tackle the data integration problem in the Oil & Gas industry is our main topic in this paper.

There are different approaches to realise data integration. Ziegler and Dittrich separated the data integration approaches into different abstraction levels [2]: (1) manual level, (2) user interface level, (3) application level, (4) middleware level, (5) data access level, (6) data storage level. For manual data integration, all the integration work is done by the end users. With a common user interface, end users could use an interface like the World Wide Web to make a query. Application level approaches rely on applications to do the integration job. Middleware provides reusable functionality used to solve dedicated aspects of the integration problem. For the uniform data access approach, a unified global view of distributed data is provided. The common data storage method has to transfer data to a new data storage with local ones being abandoned or remaining operational.

There are three well-known models of data integration: federation, warehousing and mediation ([3], [4], [5], [6]). In federation, database systems are distributed and independent, preserving database autonomy. Database systems communicate to each other directly and data can be retrieved via a middleware component. Federation is considered as a middleware level data integration approach. In warehousing, data must be extracted, transformed and loaded from remote sources to a local central repository named "data warehouse. The central repository provides a single access point to a collection of data obtained from heterogeneous sources. Warehousing is considered as a data storage level integration approach. In mediation, a mediator does not store any data on its own. It rather provides a virtual view of the integrated sources. Wrappers are often used to translate data access and manipulation requests between the mediator and data sources. The mediator splits a user query into sub-queries, sends the sub-queries to appropriate wrappers and integrates the query results locally. Mediation is considered as a data access level integration approach. As pointed out in [7], the mediation model is usually more flexible than the other two. It can deal better with autonomous and frequently changing data sources.

Another way to categorize data integration approaches is according to heterogeneity handling. There are mainly three kinds of data heterogeneities: syntactic heterogeneity, schematic (structural) heterogeneity, semantic heterogeneity [8]. Existing integration solutions are typically using shared information models in formats such as XML, developing common XML interfaces to facilitate multiple data manipulation, handling syntactic and schematic heterogeneities. With semantic data integration becoming popular; it is not a big challenge anymore to eliminate syntactic, schematic and also semantic heterogeneities. Semantic integration is based on the conceptual representation of the data and their relationships. Sowa defined this representation as ontology [9]. Ontology-based approaches are widely used to implement semantic data integration.

In this paper, we use the ontology-based integration approach together with ISO 15926 standard to address the maritime data integration problem. The paper is organized as follows. Section 2 describes ontology-based data integration approaches. Section 3 is concerned with the ISO 15926 standard and Semantic Web technologies. Section 4 presents a case study of maritime data integration based on our approach. Section 5 indicates the future work. Finally, Section 6 summarizes the paper.

II. ONTOLOGY-BASED DATA INTEGRATION

Gagnon indicated that ontology-based data integration is to use ontology(s) to semantically integrate data and/or information from multiple heterogeneous sources [10]. In computer and information sciences, Gruber has pointed out that "an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse [11]. An ontology typically contains the following components: a vocabulary of concepts or classes, often arranged in a taxonomic, tree-like structure; relationships between concepts; concept attributes or properties; and a set of logical axioms that define the assumptions about the domain.

Wache et al. have concluded that there are generally three ways to employ ontologies to data integration [12]: single ontology approaches, multiple ontologies and hybrid approaches. To understand these approaches easily, it is important to know the concept of mappings. So they defined the term "mapping as referring to the connection of an ontology to other parts of the application system, such as mappings between ontologies and the information they describe and mappings between different ontologies.

Single ontology approaches: All information sources are directly related to one global ontology which provides a shared vocabulary for the specification of the semantics. This approach requires that all information sources have nearly the same view on a domain, with the same level of granularity. This approach is vulnerable to changes in the information sources.

Multiple ontologies: Each information source is described by its own ontology separately and mapped to each other. This ontology architecture is compatible with change, but an additional representation formalism defining the inter-ontology mappings is necessary.

Hybrid approaches: A combination of the two preceding approaches is used to overcome the drawbacks of them. A local ontology is built for each information source and mapped to a global shared vocabulary. Sometimes the shared vocabulary is also an ontology. New sources can be easily added with no need for modifying existing mappings. The acquisition and evolution of ontologies are also supported.

Figure B.1 presents a typical ontology-based data integration architecture. In this architecture, the hybrid approach has been taken and the system is based on a mediation model. When implementing this architecture in the integration systems, it is very important to define and develop the integrated ontology and to achieve mappings.

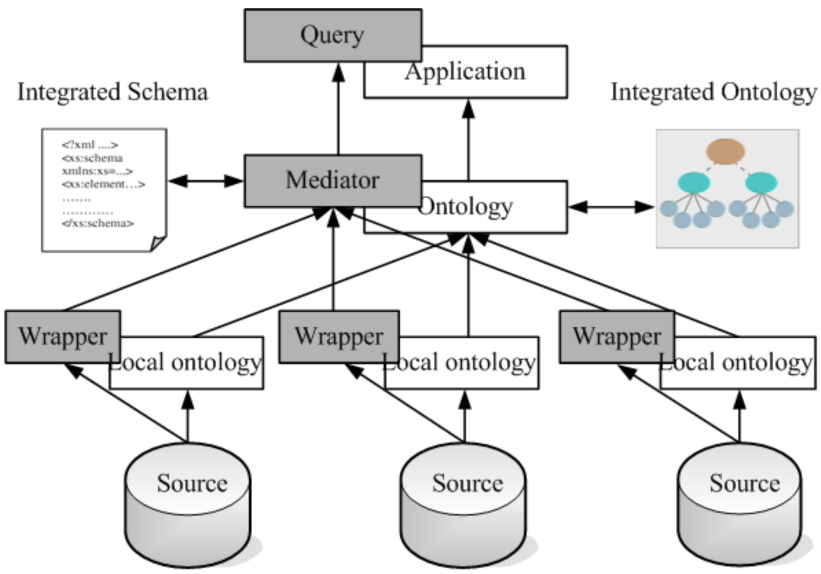

Figure B.1: Ontology-based data integration architecture.

III. THE ISO 15926 STANDARD AND SEMANTIC WEB TECHNOLOGIES

It was discovered that large traditional industries, such as the Oil & Gas industry, are dependent on internationally agreed standards to maintain a semantically equal understanding of shared domains in and across organizations [13].

The life span of an Oil & Gas industry plant is typically more than 50 years. During the plants lifetime, the information that describes it changes little compared to the turnover of computer systems and data formats. Ideally, information about an industrial facility should be treated as independent of concrete choices of data storage, use, or representation. The information standard ISO 15926 has been defined with the aim of providing formats and methodology to support this need.

PCA (POSC Caesar Association) [14], the leading global standardization organization for the process industry including Oil & Gas, has developed a methodology for data integration across disciplines and phases, and this work has been documented in ISO15926. It is an international standard with the title: "Industrial automation systems and integration - Integration of life-cycle data for process plants including Oil & Gas production facilities. It is used for data integration, sharing, exchange, and hand-over between computer systems. The goal of ISO15926 is to enable data integration for process plants, in order to reduce redundant and inconsistent information in sharing data between different companies or organizations. ISO15926 is introduced because of the requirement for a common terminology for a huge number of heterogeneous data sources. The scope of ISO15926 nearly covers the whole process plant industry, including the Oil & Gas one.

ISO15926 contains 7 parts. Part 1 is an introduction document to the ISO15926 standard which gives an overview and describes the fundamental principles. Part 2 defines a generic, conceptual data model for the representation of the life-cycle of a process plant. Part 3 defines an ontology for geometry and topology based on concepts of ISO 10303-42 and ISO 10303-104. Part 4 specifies reference data that represent information in a certain domain. The reference data library is common for all users. Part 5 specifies the procedures to be followed by a registration authority for reference data. Part 6 is the methodology for the development and validation of reference data. It defines the abstract syntax of the reference data. Part 7 provides an implementation methodology for the integration of distributed systems.

The methodology of ISO 15926 has been used in several research projects funded by Norwegian Research Council (PETROMAKS) and important stakeholders in the Norwegian offshore industry for developing an Oil & Gas ontology. The Oil & Gas ontology might be expressed in several technologies and is standardized to be a Part of ISO 15926. So the Oil & Gas ontology could be considered as standard ISO 15926, but ISO 15926 is more than an Oil & Gas ontology.

It was pointed out in [13] that ISO 15926 has defined syntax and graphical representation yet not formal semantics. It is built on EXPRESS (ISO 10303-11) to specify its data model yet facilitates more accurate distributed specification. In this case, Semantic Web with its main technologies like RDF (Resource Description Framework), RDF Schema (RDFS), OWL (Web Ontology Language) could play an important part. These technologies allow the definition of logical relationships used for reasoning, which sometimes EXPRESS cannot fully support. The authors concluded that a transformation from ISO 15926 to an ontology language is anticipated to facilitate reasoning. OWL was chosen here as the ontology language, because OWL facilitates greater machine interpretability of Web content than that supported by XML.

Therefore, we use the ISO 15926 standard as the methodology to implement maritime data integration, with the Oil & Gas ontology being the integrated ontology, and using the Semantic Web technologies to express the ontologies.

IV. A CASE STUDY IN THE OIL & GAS INDUSTRY

A. Data Integration Requirement

For companies that provide safety systems for the customers, Safety Instrument System (SIS) is often needed. SIS is an automation system whose task is to detect and prevent potential dangerous situations that can escalate into larger accidents. It often contains the following subsystems: IMS (Information Management System), PCS (Process Control System), PSD (Process shutdown), F&G (Fire & Gas), and ESD (Emergency shutdown system). Different subsystems are independent, and together they form a chain of barriers to prevent accidents.

SIS is passive during normal operation and it has to be verified regularly that it actually will work on demand. This could be done by explicit full-scale tests. However, full-scale test is time-consuming. Alternatively, logged data from unplanned shut-downs could be used to verify activated functions. Each system in the SIS has a real-time database that stores logged data. In order to verify the functions, it is necessary to collect data from various sources. These data sources are mostly heterogeneous. Therefore, data integration is needed to get better functions.

We have developed a prototype of a tool for online analysis of SIS in operation. This tool collects data from various sources, analyzes them and reports the wellness of the SIS in operation. However, to achieve better state of data integration, we need to break down the barriers between isolated information domains within the Oil & Gas industry. Data integration is heavily dependent on the capability to semantically recognize and the ability to reason upon the logical content in information, which was also emphasized in [1]. Hence, semantic data integration for the SIS system is required.

B. Data Integration Hierarchy

Using the conceptual layering hierarchy described in [15], combined with the typical ontology-based data integration architecture we pointed out before (Figure

B.1), a practical data integration hierarchy has been designed. The conceptual laying hierarchy has been widely used in ontology-based data integration scenarios. It is arranged in four layers: data source, data source ontology, domain ontology, and view. This hierarchy has two main advantages: flexibility (changes of one layer will not affect other layers) and extensibility (adding a new data source with new schema is easy). Figure B.2 shows the conceptual layering hierarchy and the practical layering used in this case study.

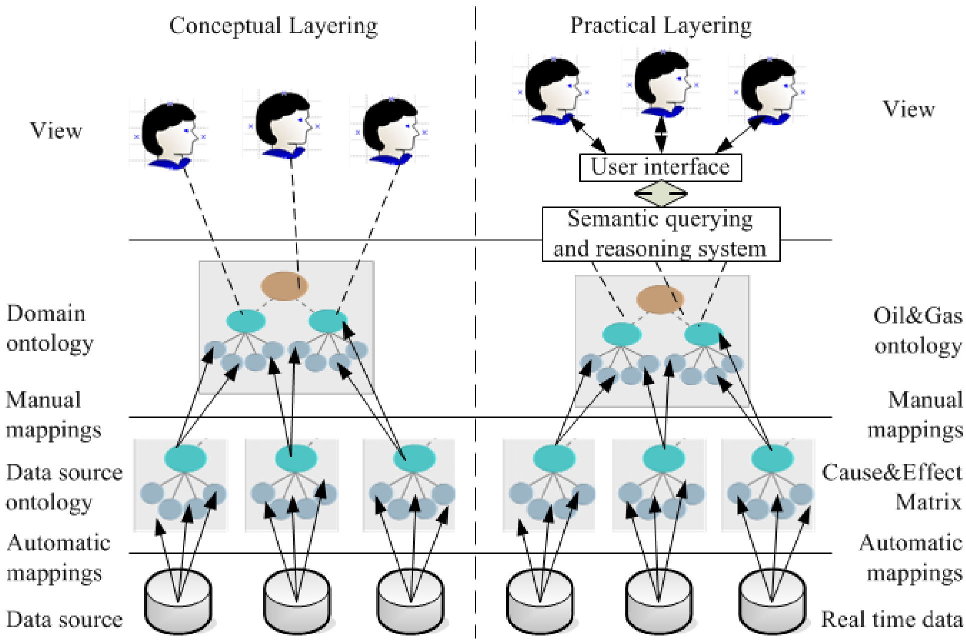

Figure B.2: Conceptual layering and practical layering of the ontology.

- The "data sources" layer stores the raw data. Usually the data are stored in a relational database, such as MySQL and Oracle. For this case study, the data source is the real time data from the SIS system.
- The "data source ontology" here is not a real ontology, since it does not represent a shared conceptualization of a domain which was explained in [15]. It is the schema of the data source, such as the Cause & Effect matrix in our case study. The Cause & Effect matrix describes the logic of a SIS in a detailed way.
- The "domain ontology" is the real ontology, which provides the terminology and taxonomy for the domain. Here we use the Oil & Gas ontology as the domain ontology.

- The “view” layer could use the common user interface to query for the information. The semantic querying and reasoning system is defined by the software engineer who is familiar with the domain ontology and the developer tools of the ontology.

When designing such ontology-based data integration systems, it should be noted that semi-automatic or manual mappings at different abstraction levels may be necessary with the help of domain experts. Real life challenges such as heterogeneity, dynamics, distribution and limitations on data representation technologies sometimes impose much more complexity than expected.

C. Implementation Methodology for Data Integration

ISO 15926 part 7 specifies the implementation methodology of data integration. It answers the following questions: How is the ISO 15926 standard related to RDF and OWL? How does the data integration work in a distributed system? ISO 15926 part 7 specifies generic templates and object information models (OIMs).

A template is a standard format for a kind of data sheet to enable a common look for every user. It is a lower level model built upon ISO15926 part 2. A template can be considered as a Lego block that you can use to build anything you like. If we regard part 2 as “grammar” and part 4 as “words”, than the template is the “phrase” of a sentence. The template is a generic model, and there are specified templates for each field of industry. For example, there are templates for “pumps” and “piping”. In the first step of data integration, you have to find out which templates fit for the domain.

The OIM ontology specifies the template. It is defined by domain experts, which means that each company that wants to use the ISO15926 standard has to participate in the development of OIMs. The models in OIM give more specific information of things than the template.

D. Overview of Data Integration Implementation

According to the user requirements, the data integration hierarchy and the implementation methodology defined in ISO 15926 Part 7, we have designed a data integration system as shown in figure B.3.

The following steps are necessary to build the data integration system:

- Mapping the Cause & Effect matrix to an OWL ontology based on the ISO15-926 standard. The OWL ontology is stored in the facade. A facade is an RDF quad store, set up to a standard schema and API (Application Programming Interface), as defined in ISO 15926 Part 7. We have tried two approaches to implement this part.

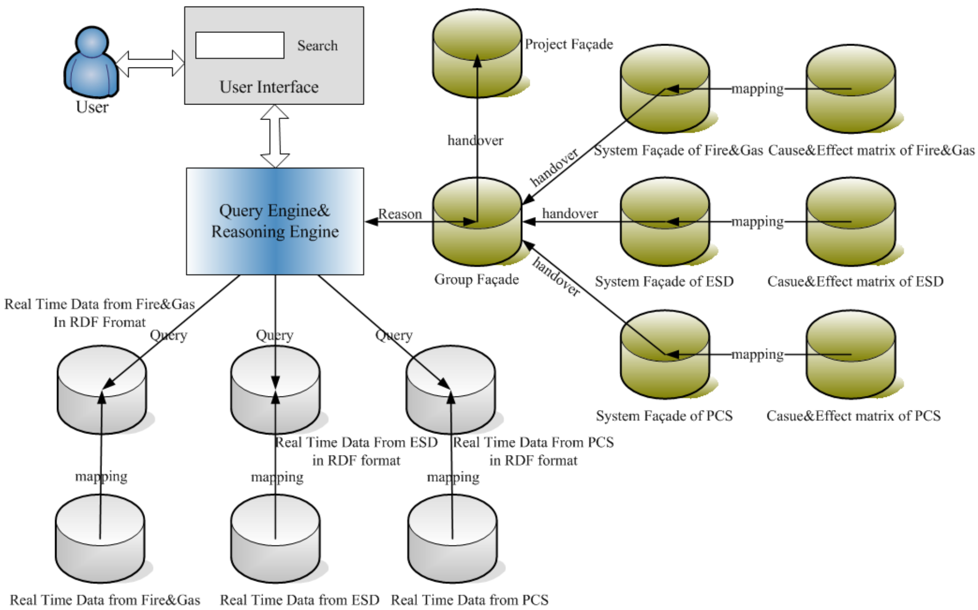

Figure B.3: Data integration system architecture.

1. Manual mapping approach: use the ontology creation and modification tool Protégé [16] to map the Cause & Effect matrix to the OWL ontology based on the ISO15926 standard.
2. Automatic mapping approach: Use the transformation software JXML2-OWL [17] to map automatically. (This approach is still in progress.)

- Mapping the real-time data to the data source ontology; depending on the format of the real-time data, two approaches can be used:

  1. Mapping from a relational database to an OWL instance: use the Jena API [18] for implementation.
  2. Mapping from XML to OWL: use JXML2OWL for implementation.

- The semantic query engine and reasoning engine implementation contains two parts:

  1. The query engine receives the query information from the user interface, queries the real time databases and returns the query results. The query engine is developed based on Jena API; and SPARQL (Query Language for RDF) is used to query data.
  2. The reasoning engine reasons the queries of the user, gets the semantic information according to the ontology stored in facade. The reasoning

engine is also developed based on Jena API; and the reasoning tool Pellet [19] is used as a reasoner.

- User interface: The user interface is user friendly. It supports keyword searching and advanced searching with different conditions. JSP (JavaServer Pages technology) is used to implement the user interface.

Mappings: The automatic mapping roadmap is shown in figure B.4. This mapping approach attempts to use the JXML2OWL mapping tool to lift the XML to OWL. As we see from the figure, in theory, by defining the mapping from Cause & Effect XML schemas to Oil & Gas ontology OWL, the Cause & Effect XML instance can be transformed to the Oil & Gas ontology automatically. Similarly, by defining the mapping from real time data XML scheme to Oil & Gas ontology OWL, the real time data XML instance can be transformed to a real time data OWL instance automatically. After the transformation the real time data XML instance based on the Cause & Effect matrix will be based on the Oil & Gas ontology OWL instance. However, the real transformations are much more challenging, especially when real time instances are involved. One of the major challenges today is the mapping in order to have the consistence with the ontology. There are many "dialects" in even the same XML standards, e.g. WITSML (Wellsite Information Transfer Standard Markup Language) [20]. Hence most of the mappings have to involve domain experts for the correct interpretations, which means the system must involve consistence and correctness checking before moving on to the next step.

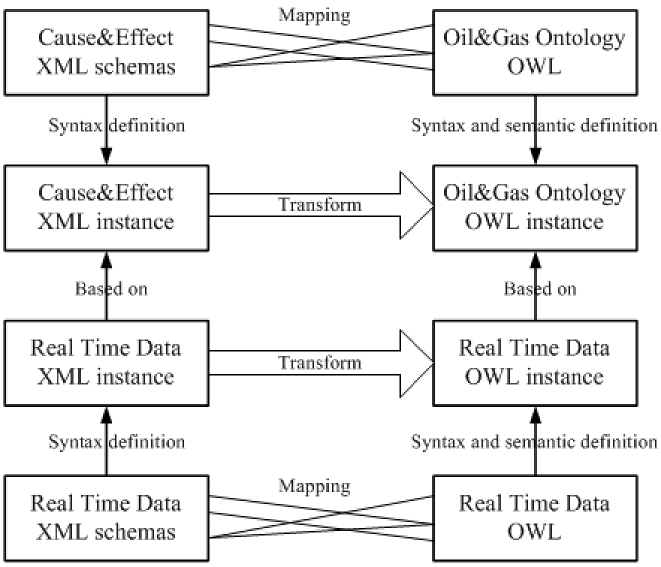

Figure B.4: Automatic mapping roadmap.

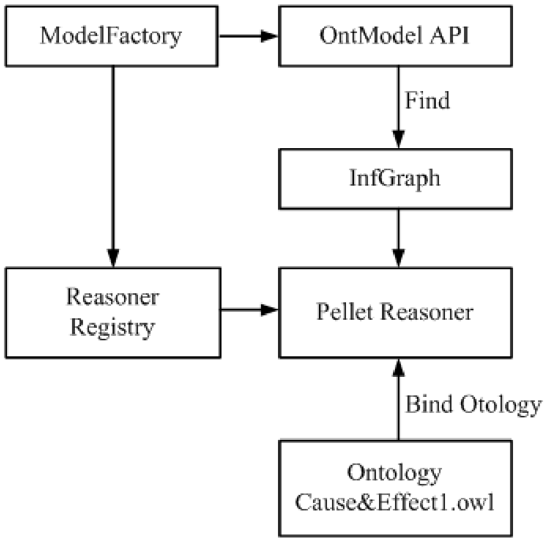

Figure B.5: Jena inference structure.

Query and reasoning: The Jena API is used as programmatic environment, and Pellet is used as a reasoner in the semantic reasoning implementation. Reynolds has described a Jena inference structure [21] shown in figure B.5. We adopted this structure in our case study. The ModelFactory is used to associate a model to a reasoner in order to get a new model. The new model has the inference data as inferred by the Pellet Reasoner. The OntModel API provides methods to find the graph in the InfGraph. The semantic query system here supports querying of

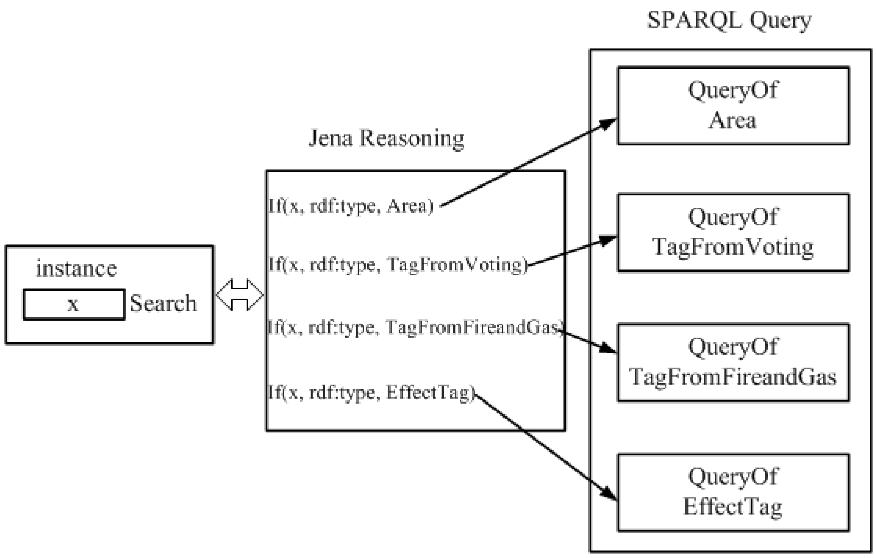

Figure B.6: Query of instance.

instance, class and instance with timestamp. Right now only the instance and class query are fully implemented. SPARQL is mainly used as query language. The Jena API is used for realization of the instance, and manipulating of the ontology model. As shown in figure B.6, when the user queries the instance on the user interface, the Jena Reasoning system finds out the class, to which the instance belongs; then matches the class to the predefined SPARQL query method of the class.

Figure B.7 shows the query of the class. It uses the Pellet reasoner to provide the inferred model of the given class, and also infers the status of the tags. The query will list all the instances of the given class. Use the method in the Jena API to get the description and local name of each instance. If the instance is a kind of tag, it will also give the status of the tag.

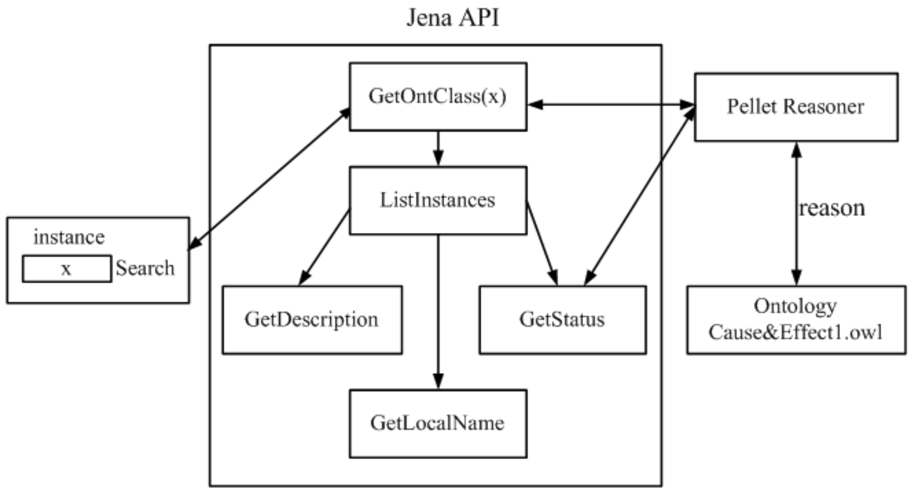

Figure B.7: Query of class.

E. Testing Results

The users of all the integrated systems share the same user interface. The query engine and reasoning engine separate the users from the data layer.

A prototype for the querying and reasoning system has been developed. This prototype proves the concept of reasoning, and shows the improvement of data quality and accessibility by querying the information. However, the main challenge of this designed system is the efficiency issue. There are still difficulties for the reasoner to be able to efficiently deal with real time data.

Automatic mappings from real-time data to the Cause & Effect matrix have been done as well as the manual mappings from the Cause & Effect matrix to the Oil & Gas ontology. The JXML2OWL approach has showed some limitations when it comes to heterogeneous data sources. It simply supports the match between the XML tree view and the OWL graph view. It is not possible to add any restrictions in the newly created OWL file. So it is better to manually map the local ontology to the domain ontology with participation of the domain experts.

System facades store the data source ontology. Each system facades can handover its data to a group facades. The group facades stores the integrated ontology of the whole group. The reasoning engine reasons through the group facades to get the integrated meaning of the query user input. Then the reasoning engine sends the result back to the query engine. Therefore, the query engine can use the result from the reasoning engine to implement semantic query handling of the real-time data.

Based on the analysis and the implementation of data integration, this case study has proved that the ISO15926 standard is very useful in the SIS system to realize data integration.

V. FUTURE WORK

First we have to improve the query mechanism to support search through timestamps, so that the safety person could be able to get the information from offshore of any time or time range. Then we need to improve the mapping mechanism to support automatic mappings from the Cause & Effect matrix to the Oil & Gas ontology to reduce the vast human interaction work.

Another task is to fulfill application integration. Data integration concerns integration on the data and information level, and application integration considers integration on the level of application logic or integration of functionalities. For the maritime industry nowadays such as in the "Maritime Communications - broadband at sea (MarCom)" project (2007-2010), integration of applications is necessary.

VI. CONCLUSIONS

In this paper, we introduced data integration and the integration requirement in the maritime industry. Some general integration approaches are categorized and compared, focusing on ontology-based ones. We gave an overview of the ISO 15926 standard and the Semantic Web technologies. A case study of maritime data integration has been presented. This case study has followed the ontology-based approach with standard ISO 15926 as the methodology; Semantic Web technologies have been adopted for implementation. The result has proved the benefits they can bring for the maritime information sharing and further the Integrated Operations applications, and also showed the real life challenges that have to be fairly solved in the future work. We would like to draw a conclusion that the ISO15926 standard and the Semantic Web technologies are suitable to use in the SIS system. They together could greatly improve the current system with the real life challenges being handled step by step.

References

[1] K. A. H. Thorsen and C. Rong, "Data Integration in Oil and Gas at Norwegian Continental Shelf," in *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications - Workshops,* 2008, pp 1597-1602.

[2] P. Ziegler and K. R. Dittrich, "Three Decades of Data Integration All Problems Solved," *IFIP 18th World Computer Congress,* Vol 12, pp 3-12, 2004.

[3] A. P. Sheth and J. A. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous and Autonomous Database," *ACM Computer Surveys,* Vol 22, No 3, pp 183-235, 1990.

[4] M.P. Reddy, B. E. Prasad and P. G. Reddy, "A Methodology for Integration of Heterogeneous Databases," *IEEE Transactions on Knowledge and Data Engineering,* Vol 6, No 6, pp 920-933, 1994.

[5] T. Hernandez and S. Kambhampati, "Integration of Biological Sources: Current Systems and Challenges Ahead," *ACM Sigmod Record,* Vol 33, No 3, pp 51-60, 2004.

[6] J. F. Aldana, M. Roldán, I. Navas, A. J. Pérez and O. Trelles, "Integrating Biological Data Sources and Data Analysis Tools through Mediators," in *Proceedings of the ACM symposium on Applied computing,* 2004, pp 127.

[7] L. Q. Hieu, "Integration of Web Data Sources: A Survey of Existing Problems," in *Proceedings of the 17th GI-Workshop on the Foundations of Databases,* 2005, pp 78-82.

[8] A. R. Beck, A. G. Cohn, B. Bennett, G. Fu, S. Ramage, M. Sanderson, J. G. Stell and C. Tagg, "UK Utility Data Integration: Overcoming Schematic Heterogeneity," in *Proceedings of SPIE,* Vol 7143, 2008, pp 71431Z-71431Z-11.

[9] J. F. Sowa, "Knowledge Representation: Logical, Philosophical, and Computational Foundations," 2000. [Cited on 18 January 2010]; Available from: http://www.jfsowa.com/ontology/index.htm.

[10] M. Gagnon, "Ontology-Based Integration of Data Sources," in *Proceedings of the 10th International Conference on Information Fusion,* 2007, pp 1-8.

[11] T. Gruber, "Ontology," in *the Encyclopedia of Database Systems,* 2009. [Cited on 18 January 2010]; Available from: http://tomgruber.org/writing/ontology-definition-2007.htm.

[12] H. Wache, T. V´ogele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann and S. H´ubner, "Ontology-Based Integration of Information A Survey of Existing Approaches," *Ontologies and Information Sharing,* pp 108-117, 2001.

[13] S. Hakkarainen, L. Hella, D. Strasunskas and S. Tuxen, "A Semantic Transformation Approach for ISO 15926," *Advances in Conceptual Modeling - Theory and Practice,* Vol 4231, pp 281-290, 2006.

[14] POSC Caesar Association. [Cited on 18 January 2010]; Available from: https://www.posccaesar.org/

[15] J. Angele and M. Gesmann, "Data Integration using Semantic Technology: A use case," in *Proceedings of the 2nd International Conference on Rules and Rule Markup Languages for the Semantic Web,* 2006, pp 58-66.

[16] M. Horridge, H. Knublauch, A. Rector, R. Stevens and C. Wroe, "A Practical Guide to Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools," University of Manchester, 2004.

[17] T. Rodrigues, P. Costa, J. Cardoso and J. Fernandes, "JXML2OWL Mapper," 2006. [Cited on 18 January 2010]; Available from: http://jxml2owl.projects.semwebcentral.org/jxml2owlmapper/.

[18] B. McBride, D. Boothby and C. Dollin, "An Introduction to RDF and the Jena RDF API," 2009. [Cited on 18 January 2010]; Available from: http://jena.sourceforge.net/tutorial/RDF API/.

[19] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur and Y. Katz, "Pellet: A Practical OWL-DL Reasoner," *Web Semantics: Science, Services and Agents on the World Wide Web In Software Engineering and the Semantic Web,* Vol 5, No 2, pp 51-53, 2007.

[20] R. Yau and W. Sellick, "Wellsite Information Transfer Standard Markup Language," 2006. [Cited on 18 January 2010]; Available from: http://www.idsdatanet.com/witsml-wellsite-information-transfer-standard-markup-language.html

[21] D. Reynolds, “Jena 2 Inference support,” 2009. [Cited on 18 January 2010]; Available from: http://jena.sourceforge.net/inference/

Appendix C

Paper C - IP-based Ship-to-Shore Communication

Title:	An Integrated Wireless Communication Architecture for Maritime Sector
Authors:	Liping Mu, Ram Kumar and Andreas Prinz
Affiliation:	Department of Information and Communication Technology, University of Agder, N-4898 Grimstad, Norway
Conference:	*The 4th International Workshop on Multiple Access Communications (MACOM)*, Trento, Italy, 11-13 September 2010
Copyright ©:	Academy Publisher

An Integrated Wireless Communication Architecture for Maritime Sector

Liping Mu, Ram Kumar and Andreas Prinz
Department of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway
E-mails: {liping.mu, ram.kumar, andreas.prinz}@uia.no

Abstract — **The rapid evolution of terrestrial wireless systems has brought mobile users more and more desired communication services. Maritime customers are asking for the same, such as the concepts of "Broadband at Sea" and "Maritime Internet". Quite a lot of research work has focused on the development of new and better maritime communication technologies, but less attention has been paid on interworking of multiple maritime wireless networks or on satisfying service provisioning. To address this, an integrated wireless Communication Architecture for Maritime Sector (CAMS) has been introduced in this article. CAMS is aimed at 1) granting maritime customers uninterrupted connectivity through the best available network and 2) providing them with the best-provisioned communication services in terms of mobility, security and Quality of Experience (QoE). To address mobility challenge, the IEEE 802.21 standard is recommended to be used in CAMS in order to achieve seamless handover. CAMS provides application-level QoE support attending to the limited communication resources (e.g. bandwidth) at sea. Certain security considerations have also been proposed to supplement this architecture.**

I. INTRODUCTION

Due to the development of new applications and the fast evolution of wireless communication technologies, maritime customers are demanding better communication solutions to satisfy the increasing user requirements. In this context, concepts like "Broadband at Sea" and "Maritime Internet" have become popular [1].

Newer security and transport related applications such as video surveillance for piracy prevention and real-time updates of navigational data are increasingly being used. Besides, the usage of personal and business purpose applications like telephony and email are also considered while implementing communication systems for ship's management.

Some of these newly envisaged applications demand a strict network Quality of Service (QoS) such as guaranteed bandwidth and lower delays, and some require uninterrupted Internet connectivity. On the other hand, the fast evolution of wireless communication technologies provides maritime customers opportunity to achieve better and faster ship-to-ship and ship-to-shore communications. For example, maritime mesh networks based on long range wireless technology (WiMAX) [2] is a promising solution, and satellite broadband such as VSAT (Very Small Aperture Terminals) service is changing maritime communications dramatically. At the same time, last-mile wireless access technologies, such as IEEE 802.11, IEEE 802.16, 3GPP standards for cellular access networks, keep contributing to the near-shore communications. In order to efficiently use these wireless communication systems and take advantage of the various available features, procedures to integrate these networks and to automatically select the best underlying network are desired. To satisfy the different maritime communication requirements, network resources have to be utilized reasonably and communication services have to be provisioned and tailored to user requirements. Furthermore, mobility handling mechanisms are necessary so as to achieve a seamless mobility experience when switching between different underlying networks.

In this article, an integrated wireless Communication Architecture for Maritime Sector (CAMS) is introduced to address both application requirements and rapid technology evolution. CAMS is aimed at satisfying always-best-connected requirement and better services-provisioning in terms of mobility, security and QoE.

The rest of the article is organized as follows. In Section II, maritime customer communication requirements are identified. Section III extracts the key system requirements for maritime communication. Then, in Section IV, the integrated maritime communication architecture is presented. Finally, Section V concludes this paper and points out future work.

II. Maritime Customer Communication Requirements

Maritime communication is becoming more important in both commercial and research fields, especially in countries like Norway, which has economic dependency on an ocean area about six times the size of its mainland. After having contacted many maritime customers [1], we have acquired a detailed list of user requirements for maritime communication as given below.

Make use of available bandwidths as much as possible

Customers on ship are willing to keep in touch with shore centers and to use Internet anytime, anywhere on any device, and they prefer to have the possibility of being best connected to the available network in terms of bandwidth, quality and cost. For example, when the ship is moving to an area covered by terrestrial communication networks, services provided by these systems are mostly desirable.

Classify data traffic to optimize the usage of bandwidth

Bandwidth is a limited resource especially in the maritime scenario that drastically changes with geography. For example, in harbors WiFi is available to support high bandwidth with very low price, whereas far out into the sea (far northern area for Norway), only satellites can provide low bandwidth connectivity characterized with high cost and long propagation delays. Therefore, maritime communication resources have to be utilized reasonably and intelligently by classifying and prioritizing the communication traffic.

Service continuity at different locations and via different devices

Continuous land-based assistance and navigation are always in high demand. Service continuity becomes an important topic especially during the switching between different maritime wireless networks. For instance, a customer on-board who fills out an important on-line report to the shore center while the ship moves from communicating via satellite to WiMAX in a port, would want to keep the session uninterrupted during the transition.

More secured information exchange and Internet connectivity

It has become a security problem for shipping companies that the crew, while surfing the Internet and often unintentionally, exposes the on-board systems to viruses and hacking attacks. Security is a critical factor in the "Maritime Internet" context. Authentication and authorization mechanisms are needed for preventing attacks to the system. Also, traffic control to some extent is necessary for preventing less important data traffic from clogging the channels so as to enable the critical data to get through.

III. System Requirements for Maritime Communication

If we translate these maritime user communication requirements into system requirements, the target communication system is expected to have the following capabilities: provide optimum connectivity, mobility handling, QoE support and security.

Connectivity

With respect to maritime communications, almost all of them are based on wireless communication technologies. Compared to terrestrial wireless communication, it is challenging to deploy cellular systems at sea to achieve high data-rate transmission because of the geographic restrictions. So far, Frequency Modulation (FM) radio technology like narrowband Ultra High Frequency (UHF) and Very High Frequency (VHF) are widely used for ship-to-shore communication, with cellular systems used for near port waters. Satellites such as International Maritime Satellite (INMARSAT) are often used for long-range ship-to-ship and ship-to-shore communications. However, due to the fact that FM radio transmission has a low data-rate characteristic and satellite communication is quite expensive, considerable effort has been devoted to the development of new maritime wireless communication technologies and cheaper satellite services. Maritime mobile WiMAX networks have drawn much attention [2]. Furthermore, advances in antenna technology and satellite coverage have combined to make VSAT Ku Band satellite services very attractive, as they can provide higher data-rate transmission, good Quality of Service, compatibility with IP networks and flat-rate charging.

All in all, the target maritime communication system needs to use these existing or future maritime wireless networks to provide customers basic connectivity services.

Mobility

There are four types of mobility defined in [3] mainly from the user's point of view: *terminal*, *personal*, *session* and *service mobility*. In [4], four levels of network interworking for mobility handling are distinguished from an operator's perspective:

- *Level A would allow a user to get access to a set of services available in a visited network while relying on his/her home network credentials;*
- *Level B would allow users to be able to get access to specific services located in their home network when connected through a visited network;*
- *Level C does not require users to re-establish active session(s) when moving between networks;*
- *Level D provides seamless service continuity to satisfy service requirements also during mobility.*

An intrinsic characteristic in maritime wireless communication scenarios is heterogeneity, which refers to the coexistence of multiple and diverse wireless networks with their corresponding radio access technologies [4] and network protocols. Therefore, integrating heterogeneous wireless networks in the maritime communication scenario is required in order to take advantage of the different features of each one of them, and all four levels of interworking are desired in the target maritime communication system for mobility handling.

QoE

Bandwidth at sea is a very limited resource due to the geographical restrictions, which frequently exhibits great variations with the high mobility of maritime communication entities and the switching between different underlying wireless networks. The QoS for an application session is determined by a number of factors, such as the maximum bandwidth that can be allocated to it and the current state of the network. It mainly focuses on the network perspective and attempts to objectively measure the service delivered by the operator: bit rate, delay, jitter, bit error rate and so on. Whereas in the maritime communication scenario, customers have the possibility of choosing from multiple underlying networks; applications on board often have different capacity, integrity and security requirements related to different traffic types (e.g. distress calls, alert messages transmission, remote navigation assisting, confidential business data transmission and multimedia entertainment applications). Therefore, subjective factors regarding quality of service should be also considered in the target communication system. ITU-T has defined the QoE concept as "*overall acceptability of an application or service, as perceived subjectively by the end-user*" [5]. By considering both QoS and QoE when delivering communication services to maritime customers, application context and user expectations will be fairly treated besides objective QoS provided by the network operator.

Security

End-to-end security for ship-to-shore communication is vital, as ship-to-shore communications are mainly related to remote operation, navigation and safe shipping in which the integrity of exchanged information is vital. Additionally, business information traveling among maritime partners has to be kept as confidential; individual information for crew use is often sensitive. These confidential or sensitive information cannot be exposed or subjected to malicious intent. Hence, security mechanisms are highly desirable in the target maritime communication system.

IV. An Integrated Wireless Communication Architecture for Maritime Sector

Existing maritime wireless networks are often independent systems without interworking between them. Maritime communication service provisioning therefore has to be supported by means of specialized service platforms that could deal with quality, security and mobility simultaneously. In order to accomplish that, the first key step is to design an efficient maritime communication platform architecture.

The Internet architecture was designed to push the intelligence to the end systems with dumb networks to provide fast service provision, but it only works very well when the network qualities are stable. Telecom network architectures are designed to have complex networks to benefit simple terminals and relevantly guaranteed service provisioning, but the services they satisfy are often simple and flat. Nevertheless, it is not difficult to identify the key technologies and marketing strategies within the Internet and Telecom network architectures that have made them so successful. For example, IP technology - a common interconnection element to address heterogeneity - in the Internet paradigm has brought incredible success and rapid growth. Similarly, the combination of mobility handling and QoS provisioning in the Telecom world has attracted ubiquitous users.

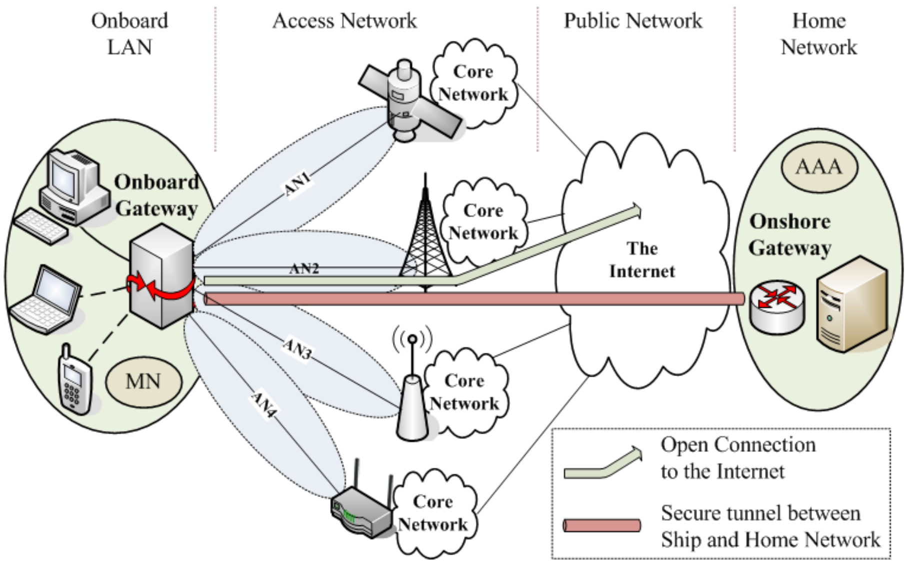

Figure C.1: An Integrated Communication Architecture for Maritime Sector.

Although none of these two architectures apply directly to maritime scenarios, a tailored communication architecture - CAMS - that optimally leverages these two paradigms can best satisfy the maritime communication requirements based on relevantly harsh conditions. The architecture is shown in figure C.1. In this architecture, IP is used as 1) the unifying technology to integrate different access networks 2) to follow the all-IP principle direction in communication evolution. The on-board gateway as a mobile node is equipped with multiple interfaces corresponding to different access technologies (e.g., AN1: satellite networks, AN2: WiMAX networks, AN3: cellular networks, AN4: WiFi networks). It cooperates with the onshore network which behaves like its home network in order to fulfill the mobility handling, QoE support and security enhancement tasks, which will be explained in detail in the following sections.

It is worth mentioning that the selection of this architecture model is not only based on performance criteria, but on its cost and feasibility as well. Any candidate architecture has to be able to backwardly integrate existing infrastructures while at the same time be easily evolved. Hence, two characteristics of our maritime communication architecture in terms of integration ability and scalability are central:

(1): In CAMS, the onshore network behaves like a home network for maritime customers. Therefore, separate subscriptions between customers and any network operator are not required. Customers have direct agreements with our home network, and our home network has separate service level agreements with each network operator.

(2): In CAMS, direct links between different networks are not necessary. Networks are connected with each other via the Internet, which is considered as loose-coupling architecture [6] for network integration. Compared with the tight-coupling model, it allows the independent deployment of each wireless network system.

Mobility Handling and Security Enhancement

Before finalizing any mobility handling solution, future trends for mobility handling must be considered. Given the mobility management tendencies described in [7], we feel three of them are most important in a maritime scenario:

(1): Different network operators will clearly provide coverage areas for the maritime customers. Hence, it is important for the communication architecture to be independent of administrative concerns.

(2): Existing mobility studies focus on solving issues between two specific technologies and many mobility mechanisms are within specific network architectures, e.g., mobility handling in IP Multimedia Subsystem (IMS) and Ambient Networks. Therefore, it is desirable to have a more general or intelligent mobility handling mechanism that could be used in all heterogeneous maritime wireless networks.

(3): Mobility management is tending towards a cross-layer approach and favoring both user and network requirements. In other words, it will become common to gather an assortment of information from several sources: link to application layer taking into account QoE factors.

These mobility handling tendencies need to be taken care of in our maritime communication architecture. Since handover is the key enabling function for seamless mobility and service continuity, it is necessary to explain handover concept first. *Handover* indicates the process by which the mobile node obtains facilities and preserves traffic flows upon the change from one point of the network attachment to another, and according to [8], there are three primary characteristics of the networks that can serve to categorize handover: subnets, administrative domains, and access technologies. Therefore, six types of handover have been defined: *intradomain*, *interdomain*, *intrasubnet*, *intersubnet*, *intratechnology* and *intertech-*

nology handover. We will discuss mobility handling for *interdomain*, *intersubnet*, *intertechnology handover* based on interworking-level concept which has been introduced in Section III. Inter-entity handovers are relevantly more common in the maritime environment and considered more difficult than intra-entity ones.

Interdomain Service Access - Level A and Level B

An interdomain handover involves the switching between different administrative domains, and requires authorization for acquisition or modification of resources assigned to the mobile. In CAMS, the onshore network behaves like a “home network” for maritime customers so as to let them be independent of administrative concerns. Therefore, Level A interworking is required to allow them to get access to services available in all “visited networks”. Authentication, authorization, and accounting (AAA) functions need to be implemented in target system (see AAA Server Service and AAA Client Service in figure C.1). AAA functions allow customers to perform authentication and authorization processes in a visited network based on subscription profiles and security credentials. AAA services are known to cause significant overall handover delay. To address this, media-independent pre-authentication interdomain handover optimization [8] can be applied in CAMS for mitigating the total delay.

In order to get access to specific services provided by networks other than the serving one - Level B interworking - requires a data transfer mechanism. Virtual Private Networking (VPN) technology uses data encapsulation to achieve secure data transfer between two or more networked devices which are not on the same private network and to keep the transferred data private from other devices or other intervening networks. There are different VPN approaches when it comes to wireless VPN. Columbitech has proposed a session-layer solution: using Wireless Transport Layer Security (WTLS) standard [9]. The WTLS solution enables secure and convenient remote access to the corporate network in an environment with multiple wireless access networks. Wireless VPN technology over WTLS standard is desired to be used in CAMS in order to achieve three aims:

- *Enable the transfer of user data between networks in order to give access to specific services provided in a network other than the serving one.*
- *Allow initialized incoming connections when using access networks with Network Address Translation (NAT) function.*
- *Enhance security for ship-to-shore communications based on tunneling technology (e.g., remote assistant and remote maintenance applications which demand high security).*

On-board LAN, onshore home network and onshore head office can constitute a virtual private network, in which AAA mechanism and tunneling technology are both applied. Therefore, security could be enhanced in two aspects. Primarily, only authorized users are allowed to access the ongoing information. Then, encryption can help achieve data integrity by protecting message contents from being modified under transit along the communication path.

Intersubnet Service Continuity - Level C

Service continuity during intersubnet handover often relies on the maintenance of a permanent mobile terminal IP address which can be addressed by Mobile IP or its variants. In Mobile IPv4, a foreign agent which works together with the home agent is needed on the visited network, while in Mobile IPv6, there is no need to deploy special routers as "foreign agents". Also, IEEE 802.21 standard which we will introduce later defines a set of handover enabling functions (for MobileIP) with required functionality to perform enhanced handovers. Therefore, MIPv6 is preferable in CAMS. However, considering that 1) MIPv4 works with IPv4 and MIPv6 was designed for IPv6 2) the slow adoption and migration from IPv4 to IPv6 3) the handover performance comparison between Host Identity Protocol (HIP) and MIPv6 in [10] and 4) HIP supports mobility between different IP address realms and easier NAT traversal [11], it is difficult to say which mobility management policy is better in the maritime context: stick to the current MIPv4 solution and move to MIPv6 when IPv6 is available or embrace HIP-based mobility handling directly. From the literature[10, 11, 12], we could expect that HIP is better than Mobile IP solutions in CAMS, while future testing and evaluation is needed.

Seamless Intertechnology Handover - Level D

Intertechnology handover is also referred to as vertical handover which can be further classified into two types [13]: *downward vertical handover* and *upward vertical handover*. Downward vertical handover is to switch between two networks that are both available. Hence, it often happens for convenience reasons (e.g., user's preference, higher bandwidth, lower delay, etc.), and the communication is still alive if the handover does not happen. Upward vertical handover to another available network is mandatory in order to keep the communication active, because the mobile customer is moving out of the coverage of the current serving network. In this sense, decision making for downward vertical handover will be much more complex and deserves more effort than the upward one. It is more important because of customers' desire, e.g., when the ship approaches the shore, customers are willing to use WiFi connection. It is more complex because it needs more information for feeding handover decision maker from all involved parts - networks, terminal and user - which is often difficult to get.

In [13], vertical handover process has been divided into three phases: network discovery, handover decision and handover implementation. Handover implementation phase usually involves link establishment, higher layer mobility management and AAA functions. Higher layer mobility performing and AAA functions introduce significant delays during handover because of the difficulty of information collection and the lack of smooth cooperation between link and higher layer functions.

In order to address these deficiencies and help with handover decision making, IEEE 802.21 standard [14] has been introduced. IEEE 802.21 defines an abstraction layer between link and network layer which can be exploited by the IP stack (or any other upper layer) to better interact with the heterogeneous underlying technologies by mapping technology-specific primitives. A new link layer entity called Media-Independent Handover Function (MIHF) is specified in the standard. This MIHF entity mainly aims at exchanging of information and commands between upper and lower layers. The main function of MIHF is to coordinate the exchange of information and commands between the different devices involved in making handover decisions and executing handovers [15]. To upper layers, it provides a media-independent interface in order to collect information from link layer and to control link behavior. Regarding the different link layer technologies, it supports mapping between the common interface and a set of media-specific primitives. MIHF is designed both for terminals and networks; therefore, remote interfaces such as terminal-network and network-network interfaces will work together with local interfaces to aid the interactions among all devices involved in the handover. These interactions are provided by a set of services: event, command and information services[15].

Since the MIHF entities within terminals and networks can talk to each other, handover could be initiated from both sides. In the maritime communication scenario, the initiation is preferred to be done by the terminal (e.g. the on-board gateway equipped with multiple interfaces) for flexibility and prioritizing user's preference. While served by a given access network, the MIHF entity of the mobile terminal can interact with the MIHF entity in the serving network in order to get the information from other available networks, making it possible to initiate an intertechnology handover with desired pre-configuration for the target network [16] to reduce the handover delay. However, it is often necessary to have a list of candidate access networks in the mobile node, and the MIHF entities need to be added within all devices involved in the handover, together with the relevant protocols. Figure

C.2 below shows the protocols stack in the client side on board and the server side on shore of our maritime communication architecture.

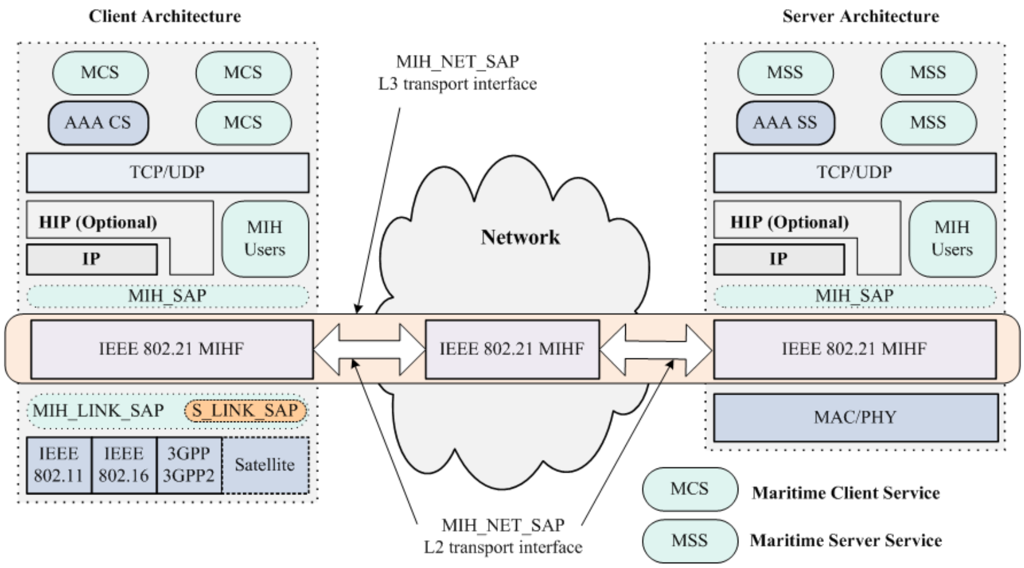

Figure C.2: Maritime Communication Architecture Protocols Stack.

IEEE 802.21 framework does not standardize the actual handover execution mechanism: handover decision-making or mobility management procedure. It recommends applying the Signal-to-Noise Ratio (SNR) metric for the handover decision-making. However, in the maritime scenario, using only SNR for handover decision-making is not enough since 1) there are different communication applications with different QoE requirements and 2) there are heterogeneous wireless access networks with different QoS characteristics. Therefore, several metrics could be combined together intelligently and dynamically so as to achieve more reasonable handover decision-making: SNR (or received signal strength (RSS)), QoS (e.g., bandwidth, data rate, access delay, losses), QoE (e.g., context information, price, user preferences, power consumption). Furthermore, a back-and-forth (ping-pong) effect should be avoided either by a more robust handover decision-making algorithm or by post-handover mechanisms.

IEEE 802.21 is designed to enable interoperability mainly among IEEE 802, 3GPP, and 3GPP2 networks. Similarly, ETSI has defined a broadband satellite multimedia (BSM) architecture [17] to provide a mechanism to carry IP-based protocols over different satellite networks by adding a satellite independent service access point (SI-SAP) interface layer, aiming to achieve interoperability among these satellite networks with different link layer technologies. BSM does not specify mobility management mechanisms. However, the methodologies of heterogeneity handling between BSM architecture and IEEE 802.21 framework are similar, hiding the differences by adding a common abstraction layer. Therefore, we could integrate SI-SAP within the IEEE 802.21 MIH framework to enable the handover between satellite networks and non-satellite networks in the maritime communication scenario, which is also recommended in [18].

QoE Support and Security Enhancement

Maritime communications are mainly based on wireless networks which often provide limited bandwidth with different QoS provisioning. Furthermore, maritime customers expect to have applications on board with differentiated parameters in terms of capacity, integrity and security. To address this requirement based on the restricted resources, application-level QoE support could be a good alternative. Application-level QoE support can be done by 1) differentiating applications with different priorities and 2) queuing their connections based on network conditions. The priorities are assigned according to customers preferences and the connection control takes place at the egress of the on-board gateway.

In CAMS, at first, different servers with different IP addresses can be used to separate applications. For example, there are basically two categories of applications: one for administrative system and the other for welfare. Under each category, there are several sub-categories. Within administrative system, there are emergency messaging sending, safety and monitoring data transmission, reporting information exchanging and so on. They could be assigned with secondary priorities. Different traffic types (data, voice, video) can be separated as well, according to different port numbers and protocols, such as real-time and non real-time traffic. They could be assigned with third-level priorities. Therefore, the priority map is chaining different queuing "disciplines" together nicely where ongoing packets are sorted by filtering them on their protocols, ports, sources and destinations. The application-level QoE support mechanism is shown in figure C.3.

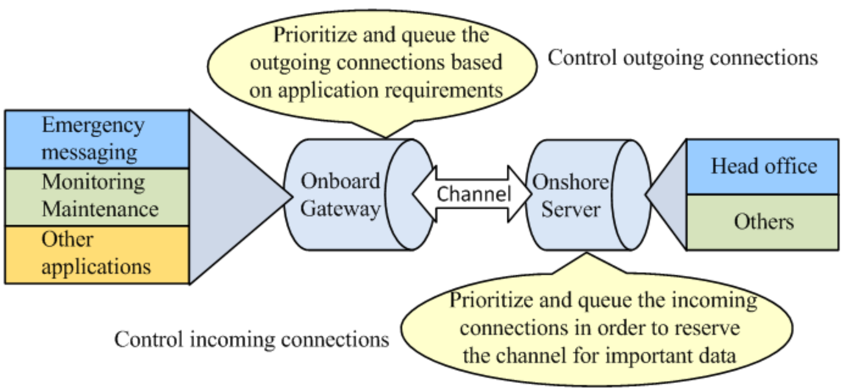

Figure C.3: Application-level QoE Support Mechanism.

By adding graphical user interface to the Linux QoS configuration technique, the on-board gateway is able to intelligently allocate limited resources in accordance with prioritized egress connection demands based on customers preferences. However, it has to be carefully implemented to be available only for authorized users. The application-level QoE support is mainly for shaping outgoing traffic. It is difficult to shape incoming traffic from user side, because QoS policy decisions for ingress traffic are controlled outside the on-board network infrastructure. However, the onshore gateway can be used as an ingress connection "controller" by queuing the incoming connections in order to reserve the channel for important data.

QoE support mechanism allows the customer to configure the system in order to make sure that more important data gets sent first, and various connections are given more fair treatment than usual. Together with our proposed VPN solution including a secure tunnel between the on-board gateway and the home network to carry sensitive information related to, e.g. ship's navigation and management, the on-board gateway has the capability to route certain packets through the encrypted tunnel, while separately forwarding unencrypted packets to the open Internet (see figure C.1). The unencrypted packets belong to value-added services provided to on-board customers who require such connectivity like browsing or multimedia. This two-prong approach helps the architecture to have a fine grained control over the data whilst avoiding home network with unnecessary data and routing information. The secure VPN tunnel connects the two trusted networks (on-board and home) through untrusted networks (access core and the Internet). By combining separation of traffic and VPN technology, security can be further enhanced. However, more detailed security mechanisms will be left for future work.

V. Conclusion and Future Work

In this work we have introduced an integrated wireless communication architecture that tries to provide maritime customers ubiquitous services by integrating heterogeneous underlying wireless networks. Solutions for addressing key issues such as quality, security and mobility are covered in this architecture with more detailed discussion of seamless handover. We believe that future maritime communication will benefit much from integration of existing networks, and quality, security and mobility have to be carefully addressed simultaneously considering user preferences. However, future work is required to demonstrate the performance of our proposed architecture:

- A new maritime handover decision-making algorithm will be designed and tested in order to intelligently switch among heterogeneous maritime wireless networks and handover between satellite networks and non-satellite networks will be further studied.

- Application-level QoE support on both on-board and onshore gateways will be tested to prove the efficiency of reasonable utilization of limited resources according to different application requirements.
- Wireless VPN technology and AAA functions will be applied to the maritime scenario for measuring the security improvement.

References

[1] MARCOM: Broadband at Sea, Internet for coast, polar regions, offshore and sea farming, available at: http://www.marcom.no/

[2] Pathmasuntharam, J.S., Jurianto, J., Kong, P.Y., Ge, Y., Zhou, M., Miura, R.: High Speed Maritime Ship-to-Ship/Shore Mesh Networks. 7th International Conference on ITS Telecommunications. 1–6 (2007)

[3] Schulzrinne, H., Wedlund, E.: Application-layer mobility using SIP. IEEE Service Portability and Virtual Customer Environments. 29–36 (2000)

[4] Ferrus, R., Sallent, O., Agusti, R.: Interworking in heterogeneous wireless networks: Comprehensive framework and future trends. IEEE Wireless Communications. 17, 22–31 (2010)

[5] ITU-T Recommendation Std: Vocabulary and effects of transmission parameters on customer opinion of transmission quality. ITU-T, P.10/G.100. (2008)

[6] Buddhikot, M., Chandranmenon, G., Han, S., Lee, Y.W., Miller, S., Salgarelli, L.: Integration of 802.11 and third-generation wireless data networks. IEEE 22nd Annual Joint Conference of Computer and Communications. 1, 503–512 (2003)

[7] Fernandes, S., Karmouch, A.: Vertical Mobility Management Architectures in Wireless Networks: A Comprehensive Survey and Future Directions. IEEE Communications Surveys & Tutorials. 1–19 (2010)

[8] Dutta, A., Famolari, D., Das, S., Ohba, Y., Fajardo, V., Taniuchi, K., Lopez, R., Schulzrinne, H.: Media-independent pre-authentication supporting secure interdomain handover optimization. IEEE Wireless Communications. 15, 55–64 (2008)

[9] Columbitech: Columbitech Wireless VPN - Technical Description. (2007)

[10] Jokela P., Rinta-aho T., Jokikyyny T., Wall J., Kuparinen M., Mahkonen H., Melén J., Kauppinen T., Korhonen J.: Handover performance with HIP and MIPv6. 1st International Symposium of Wireless Communication Systems. 324–28 (2005)

[11] Nikander P., Gurtov A., Henderson T.R.: Host Identity Protocol HIP: Connectivity, Mobility, Multi-Homing, Security, and Privacy over IPv4 and IPv6 Networks. IEEE Communications Surveys & Tutorials. 12, 186–204 (2010)

[12] M. Ratola, Which Layer for Mobility? - Comparing Mobile IPv5, HIP and SCTP. Seminar on Internetworking in the spring 2004. Espoo, Finland, T-110.551. (2004)

[13] Chen, W.T., Liu, J.C., Huang, H.K.: An adaptive scheme for vertical handoff in wireless overlay networks. 10th International Conference on Parallel and Distributed Systems. 541–548 (2004)

[14] IEEE Draft Std: IEEE Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services. P802.21/D14. (2008)

[15] A. De La Oliva, A. Banchs, I. Soto, T. Melia, and A. Vidal, An overview of IEEE 802.21: media-independent handover services, IEEE Wireless Communications. 15, 96–103 (2008)

[16] Dutta, A., Chakravarty, S., Taniuchi, K., Fajardo, V., Ohba, Y., Famolari, D., Schulzrinne, H.: An Experimental Study of Location Assisted Proactive Handover. Global Telecommunications Conference. 2037–2042 (2007)

[17] ETSI Std: Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Services and architectures. European Telecommunications Standards Institute, TR 101 984. (2007)

[18] Hu, Y.F., Berioli, M., Pillai, P., Cruickshank, H., Giambene, G., Kotsopoulos, K., Guo, W., Chan, P.M.L.: Broadband satellite multimedia. Communications. 4, 1519–1531 (2010)

Appendix D

Paper D - Overlay-based Ship-to-Shore Data Migration

Title:	Delay-Oriented Data Traffic Migration in Maritime Mobile Communication Environments
Authors:	Liping Mu and Andreas Prinz
Affiliation:	Department of Information and Communication Technology, University of Agder, N-4898 Grimstad, Norway
Conference:	*The 4th Fourth International Conference on Ubiquitous and Future Networks (ICUFN)*, Phuket, Thailand, 04-06 July 2012
Copyright ©:	Academy Publisher

Delay-Oriented Data Traffic Migration in Maritime Mobile Communication Environments

Liping Mu and Andreas Prinz
Department of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway
E-mails: {liping.mu, andreas.prinz}@uia.no

Abstract — **Maritime ship-to-shore communication has to satisfy different user requirements while suffering dynamic communication circumstances. Satellite networks are the primary means to communicate between ship and shore. The idea of compensating the deficiency of satellite communication with other terrestrial networks is not new. However, focus was often on how to keep users always best connected, without considering respective application needs or the potential disconnections. In this paper, we propose a delay-oriented satellite data traffic migration solution, exploiting the route repetitiveness and predictability of ships, leveraging the delay tolerance of many communication applications, with the support of delay tolerant networking architecture. We use model-checking techniques to support our claim that a delay-oriented data traffic migration approach can provide users with not only more reliable delivery but also preferred services based on some reasonable assumptions. Issues and suggestions related to implementing the DTN-based traffic migration are discussed additionally.**

I. Introduction

The current primary means for broadband communication between ship and shore is via satellite networks. Satellite communication is usually expensive, with limited capacity, high propagation delay, high bit error rate, and there are special problems related to the lack of coverage in places such as fjords, ports and at high latitudes. Therefore, maritime customers are reluctant to use satellites to communicate, especially when large amount of data usage can be incurred, like satellite Internet. Solutions to address these problems mainly focus on two aspects, one is to improve the transmission technology itself, and the other is to exploit existing communication technologies from an architectural perspective such as integrating different maritime wireless communication systems. Our methodology is to follow the latter, exploring different features of multiple wireless communication technologies. The integration of different maritime networks and seamless handover between them have been discussed in our previous work [1], but the traffic migration was mainly based on the network connectivity. In this paper, we explore further on user preferred satellite traffic migration.

Basically, there are two types of application categorized from the users' perspective in the maritime communication area: one is for operation and administrative usage and the other is the individual (passengers, crew, etc.) type. When we look at applications from this user point of view, the intuitive way of confronting limited communication resources would be prioritizing the different types. If we inspect applications from a network perspective such as the transmission latency, we can coarsely separate them into delay-tolerant and real-time applications. This way, solutions for addressing limited maritime communication resources would be differentiated data delivery based on different delay thresholds. One potential implementation is user-preferred traffic migration on delay-tolerant applications, which we name delay-oriented data traffic migration.

In order to implement a delay-oriented migration strategy, network architectural support is necessary. To this end, we explore a store-and-forward approach adopted by the Delay-Tolerant Networking (DTN) architecture [2]. DTN is a promising networking practice for challenged environments including the satellite communication. It was argued that the future Internet architecture should inherently consider challenged networking conditions as a regular case rather than treating them as errors, and one prominent example of achieving so is a DTN solution [3]. Before implementing DTN-based data migration for maritime customers, a first-step testing can be done by means of model checking techniques. First, we test whether a DTN approach is able to cope with long and frequent disruptions in challenged satellite networks compared to a traditional TCP implementation. This decides whether or not the DTN-based networking architecture can facilitate data traffic migration based on the delay thresholds. Then, we use model checking to prove that delay-oriented data traffic migration can fulfil users' requirements better compared with a connectivity-based handover policy.

Regarding the real-life implementation of delay-oriented data migration, we propose to deploy one DTN gateway on ship and another on shore. The on-board DTN gateway enables the connection between local on-board networks and the Internet via either satellite link or other terrestrial networks. Apart from that, it will also decide whether to delay the application and handover to which candidate network based on 1) the current ship location, 2) the history data and schedule about travelling routes and 3) the characteristics of the application.

The rest of the paper is organized as follows. In Section II, related work is discussed. Section III describes some application scenarios for data traffic migration in the maritime environment and the system support. Section IV presents our delay-oriented migration solution, and the process to check the correctness of this solution by using model checking techniques. The simulation and verification results are presented after. Then, in Section V, issues related to real implementing delay tolerant architecture in the target scenario are discussed. Finally, Section VI concludes this paper and points out future work.

II. Related Work

Maritime broadband communication is important in both commercial and research fields, especially for countries that economically depend on ocean areas very much. Most of maritime communications have to rely on the wireless communication technology. Compared with terrestrial wireless communication, it is hard to deploy cellular systems at sea due to the geographic restrictions. In this case, satellite networks became the primary means for maritime broadband communication. Although the satellite system provides almost a global coverage for connectivity, it is still expensive for maritime customers to use it regularly to access Internet. Ideas of complementing satellite with other terrestrial communication technologies have been proposed in the literature. For example, in [1], an integrated maritime communication architecture has been investigated. Seamless handover between different networks was discussed in detail together with the prioritization of different applications. However, intentionally delaying an application to a user preferred communication opportunity was not covered, which has recently drawn much attention in the terrestrial cellular field.

Migrating data transfer from a terrestrial cellular network to other communication methods was mainly caused by the cellular overload problem. The way of offloading has been influenced by the idea of opportunistic networking [4] and the DTN paradigm. In addition to offloading cellular traffic to WiFi networks and femtocells, opportunistic offloading [5], leveraging peer-to-peer opportunistic communication between mobile users, has recently become popular [6]. In paper [7], a system called Wiffler for augmenting mobile 3G capacity using WiFi was proposed. In this system, two important factors of individual mobile users have been considered: 1) the delay tolerance and 2) the offloading potential. The aim is to reduce 3G usage by solving the trade-off issue between these two factors.

We share a common idea with the Wiffler system about leveraging delay tolerance of many communication applications such as email and file transfer. However, our targeted network for data to be migrated to is not limited to WiFi but other terrestrial networks as well, and the migration aim is different. Our motivation is to help maritime customers with more opportunities to use cheaper and better communication services. In order to do so, we exploit the DTN networking architecture, which can also provide maritime customers with guaranteed data transmission in default satellite networks when facing long interruptions. Our ultimate goal is to encourage maritime customers to use modern wireless communication technologies more often, instead of traditional transportation methods like post for data transmission. In this sense, although we recommend and enable customers to exploit terrestrial networks as much as possible for saving cost, satellite use in the long run is not decreasing but increasing as the whole data demand increases tremendously, in particular the large amount of data transfer for ubiquitous Internet access.

III. Maritime Satellite Data Traffic Migration

Delay tolerant applications are the main target of data migration in the maritime mobile environment. The migration will focus on ship-to-shore communications, where communication originated from the ship is examined. Communication which originates from the shore such as remotely updating software on board will use the same migration process, but it needs the satellite link to behave as a control channel to exchange dynamic information between the two places. We do not consider emergency messages transfer, because we assume that emergency information should not be delayed and it is often transmitted via specific channels such as the marine VHF (Very High Frequency) radio or the AIS (Automatic Identification System) link. If emergency messages need to be sent out by all possible means, we can set the delay threshold of emergency data as zero in our model. Figure D.1 shows a simple

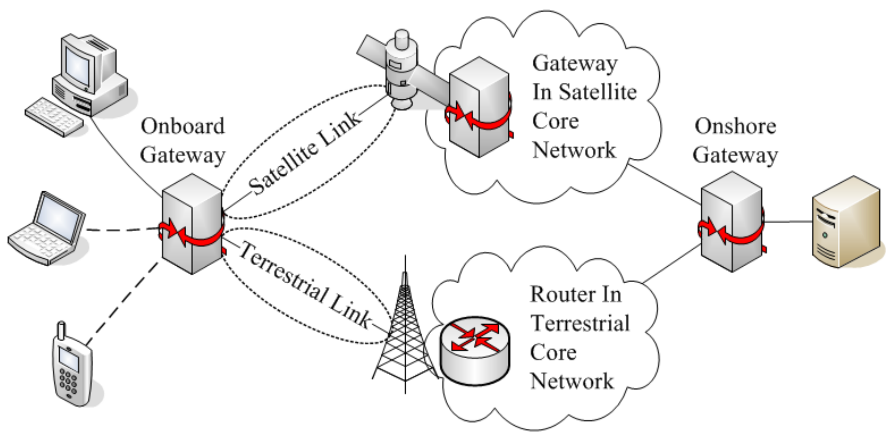

Figure D.1: System Structure for Maritime Data Traffic Migration.

system structure for maritime satellite data traffic migration scenarios.

Application Scenarios

Maritime navigation and operation systems often need to be updated frequently with new and revised forms and requirements and fed with new information. Currently, administrative data communication between ship and shore is dominated by emails with attachments sent via a satellite link or marine VHF radio. Since the transmission capacity of VHF is limited and the satellite transmission is expensive, larger amounts of data are still often transferred to physical storage media and transported by post or other delivery methods. This is slow, cumbersome and expensive in the long term with high risk for errors and high delays.

Relatively, satellite communication has the potential to grow in ferries and cruise ships with a large amount of passengers on board. These passengers are willing to get good Internet services, such as web browsing, connecting to well-known online services (e.g., Youtube) and downloading desired videos, and receiving subscribed information from scheduled websites regularly. At the same time, passengers would like to use mobile phones to take pictures and record videos, and upload them to the cloud storage (e.g., Facebook) to share with family and friends. However, the current high cost of using satellite Internet on board has made a lot of passengers hesitate to try this service.

Therefore, making use of cheaper and higher-capacity channels is desirable, which can be implemented by migrating data traffic from satellite to these channels based on the users' delay tolerances on many applications:

- Delay large administrative files and bulk data transfer to terrestrial networks based on applications' delay thresholds and the future connectivity to these networks.
- Migrating large amount of individual data downloading/uploading from satellite to cheaper communication services within users' delay-tolerance limits. Peer-to-peer opportunistic data sharing among mobile users can be further explored.

System Support

Maritime Wireless Communication Networks Communication networks that are able to support large amounts of data transfer in the maritime area are satellite networks and coastal communication systems. Satellite communication systems are utilizing GEO, LEO and HEO orbits, in which, GEO systems are the best for broadband service because they can offer higher data speeds. However, the signals of a GEO satellite cannot reach some polar regions and the propagation delays are high. Following are some factors that can impair the performance of GEO based communication:

- High propagation delay with RTT (Round Trip Time) around 600 ms in a bidirectional GEO system [8]
- High transmission error rate with a residual PER (Packet Error Rate) ranging from 10^{-3} to 10^{-1} for GEO mobile terminals [9]
- Lacking coverage at relatively high latitudes, such as Europe and the US, where the small elevation angle may result in frequent link obstructions

As WiFi networks have been deployed in ports together with 3G/4G networks installed along the coasts and offshore areas, terrestrial networks started to contribute more to maritime communication. It has become possible to take advantage of different features of multiple maritime wireless communication technologies, and users can benefit from data migration to a preferred communication system.

Delay Tolerant Networking We already know that there are many factors that can impair a GEO based data transmission. These factors and the combination of them can contribute to long durations of disruptions and high disruption frequency, especially for mobile GEO terminals. This will severely degrade the performance of traditional TCP and other standard Internet transport protocols. Therefore, many approaches have been proposed in the literature to handle this problem [10, 11, 12, 13], where delay tolerant networking exhibits as a promising candidate. We hereby

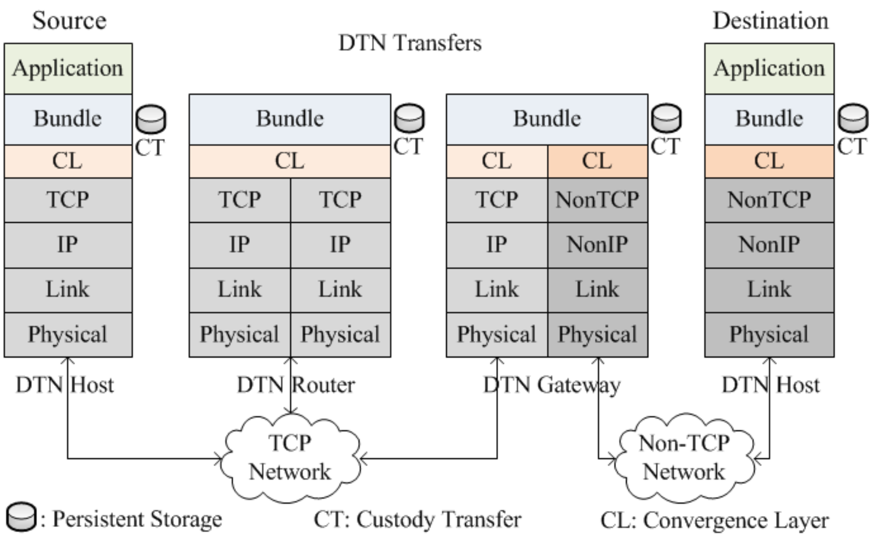

Figure D.2: DTN Architecture and the Protocol Stack.

choose it in the maritime data migration context because of reasons as follows: first, it can improve the reliability of satellite communication; second, it smooths the integration process of heterogeneous maritime communication networks, and most importantly, it enables delay-oriented data traffic migration. We list the detailed tasks that a DTN architecture can bring to facilitate data migration based on delay thresholds. The architecture itself and the protocol stack is presented in figure D.2.

- *Management of interruptions over the default satellite access network, especially for mobile terminals*
- *Functioning as an overlay to handle interoperability among heterogeneous maritime communication systems*
- *Dealing with dynamics during a handover transition*
- *Information storage at intermediate nodes to help enable intentional delay of many communication applications*

IV. CORRECTNESS CHECKING

A Delay Oriented Migration Strategy

A simple way to do satellite traffic migration would be: using the preferred networks whenever they are available and switching back to satellite whenever they become unavailable, such as the connectivity-based handover strategy described in [1]. But this connectivity-based migration will not work well in the real maritime environment because:

(1) Most of the time, ships are within the satellite coverage but not a terrestrial network range, and the simple policy will severely limit the fraction of data that can be migrated from satellite to other alternatives.

(2) Sometimes, connectivity via terrestrial networks lasts short with a poor quality because of ships' mobility, and the simple migration policy will cause undesired performance degradation for some applications.

Therefore, we should both make use of the delay tolerance of many maritime communication applications and explore the predictability of future maritime communication opportunities, in order to make a best migration strategy. The aim is to maximize the amount of data that can be migrated from satellite to other alternative networks without degrading the user experience from intolerable delays. This migration strategy can be supported by a DTN approach and the arguments are as follows.

- *Statement one*: Exploiting a DTN approach to enable delay-oriented data traffic migration can help maritime customers get better guaranteed data delivery between source and destination, compared with a traditional TCP based solution.

- *Statement two*: Introducing the delay threshold of a communication application and using this to migrate data transfer to a preferred network can help maritime customers get cheaper communication services without degrading user experience, compared with a connectivity-based handover policy.

Simulation

We use model checking techniques to prove the aforementioned two statements. Model checking can be used to formally verify concurrent finite state systems for ensuring reliability. SPIN model checker [14] [15] is a commonly used model-checking tool. SPIN accepts design specifications written in PROMELA (Process or Protocol Meta Language), a verification modelling language. Correctness claims can be specified in standard Linear Temporal Logic (LTL) that can express, e.g., safety properties (something bad never happens) and liveness properties (something good keeps happening).

Before we use PROMELA to model our system and to prove the statements, we abstract the difference between a TCP implementation and the DTN solution in figure D.3 below. A TCP virtual link is like a direct link between the source and destination whereas the DTN virtual link has a relay node in between to help.

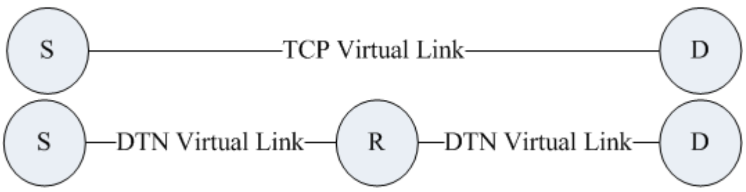

Figure D.3: TCP vs DTN.

We therefore model the system by using three processes in PROMELA: the source *S*, a DTN intermediate node *R* and the destination *D*. Regarding a TCP implementation, if the channel between source and destination is available, it means:

- a) For a transmission to start up, the real link is available
- b) For an on-going transmission, the disruption duration is shorter than the maximum tolerable disruption length of a typical TCP implementation (around 20 minutes).

In this case, the main task is to do a performance comparison (retransmission mechanisms for disruptions and congestion control for random losses) between the TCP and DTN implementations which was detailed in [16] and [12]. The correctness checking in this paper will handle another case where the channel between source and destination is unavailable for a TCP implementation (no connectivity support for a transmission to start up or the disruption is longer than the maximum tolerable disruption length of an on-going transmission). Therefore, in the simulation, the direct data transmission between source and destination via traditional TCP will fail. It can be seen from the upper part of figure D.4 that the data transfer between *S* and *D* is time-out. Then, we see from the lower part of the figure that

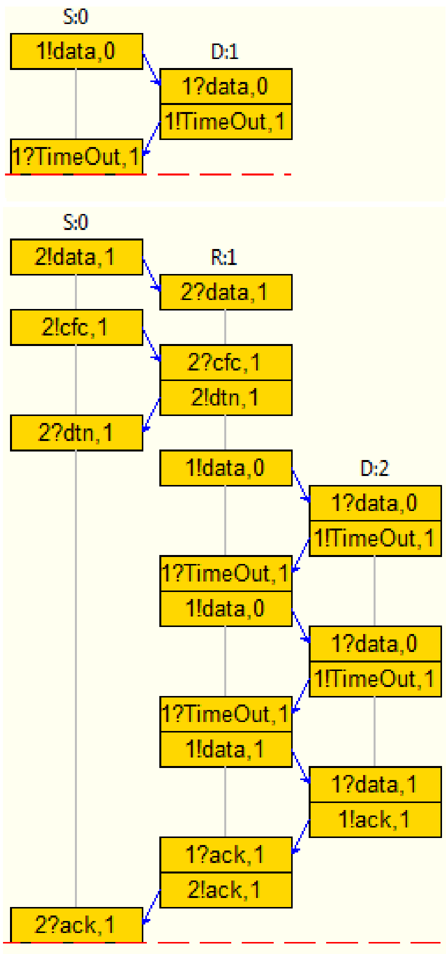

Figure D.4: Comparison between TCP and DTN.

the sender is trying the DTN approach by sending the data to the intermediate DTN node *R*, and at the same time, a message of *call for custody transfer*(cfc) has also been delivered to *R*. If *R* accepts the custody transfer, then the sender does not need to worry any more, the DTN node will take over the task of reliable transfer and implement its retransmission mechanisms. In our simulation result, we can see that the DTN node has accepted to be a custodian and sent a (dtn,1) message back to the source. Then it tried retransmission several times and finally got the data delivered.

We model the second statement by using similarly three processes in PROMELA: the source *S*, the DTN gateway *G* which performs the migration function, and the preferred link *PreferredLink*. The successful direct data transfer between the source and the preferred link means the source is currently within the coverage range of the preferred network. The failure means that it is not in the preferred range and it uses a connectivity-based data migration strategy. However, if the application in

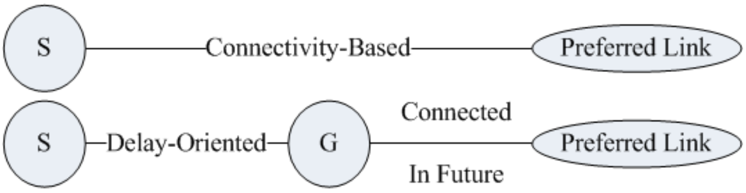

Figure D.5: Connectivity-Based vs Delay-Oriented.

use can tolerate some delay and it prefers an alternative network other than satellite, although the source is only within a satellite footprint, it can still use the preferred connection as long as it adopts a delay-oriented migration solution and the time to obtain the better connectivity is within the application's delay threshold. In this sense, with a delay-oriented data migration strategy, it seems that there is always an alternative virtual link available other than the default satellite link, see figure D.5.

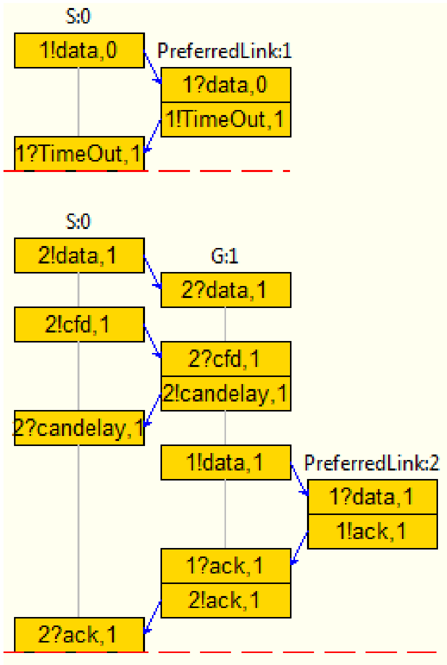

Figure D.6: Comparison between Connectivity-Based and Delay-Oriented.

The main difference between the two models for the two statements is the aim. In addition to reliable data transfer from source to destination, the aim of the second model is to use the better channel as much as possible. At the same time, the DTN gateway in the second model is more than a regular DTN node; it runs a daemon program to a) read the data's delay threshold b) predict the future connectivity of preferred networks and c) intentional delay the matched data. Also, there is an implicit difference: for the first model, the DTN node will run its retransmission mechanisms till the data finally got delivered, but for the second model, if the data cannot be successfully transmitted via the preferred link before its delay threshold, then it will be transferred through the default channel instead of retransmission on the better channel. As we can see from the simulation result in figure D.6, there is no retransmission.

Verification

In order to verify our first argument, we make the following assumptions, which are reasonable in real life.

- The transmission between source and the DTN node is successful because it is often a stable link. For example, in figure D.1, if the source is the on-board terminal, then the intermediate DTN node is the on-board gateway; if the source is the onshore center, then the intermediate DTN node can be the onshore gateway or the DTN gateway within the access core network.

- The DTN intermediate node is able to do custody transfer (retransmission responsibility).

- In the future, there will be a connection between the DTN node and the destination. This assumption is reasonable because of the mobility of the ship.

Based on these assumptions, we define the requirement of our first statement verification as: *If a DTN node accepts the custody transfer and there will be a future connectivity between the DTN node and the destination, then the data should be always successfully delivered to the destination and the ack packet should be returned to the source.* We formulate the LTL logic for this requirement using *CRCDR, CUSTODY, STATE* as global variables (*CRCDR* means the connectivity between a DTN node and the destination; *CUSTODY* represents the custody transfer and *STATE* describes the status of the data delivery from source to destination). Then we get the following LTL formula for never claim in SPIN:

```
p -> [] q
#define p (CRCDR && CUSTODY)
#define q (STATE)
```

Similarly, we get the never claim formula for proving the second statement as following:

```
p -> [] q
#define p (CANDELAY && CRCDR)
#define q (STATE)
```

Here, *CANDELAY* means that the data can be delayed by prediction and *CRCDR* means that in future, a better channel between the DTN gateway and the destination is available within the delay threshold. As long as these two assumptions hold, the data will be always delivered via the better connection.

```
 (Spin Version 5.2.5 -- 17 April 2010)
        + Partial Order Reduction
Full statespace search for:
        never claim             +
        assertion violations    +
        acceptance   cycles     +
        invalid end states      -
State-vector 64 byte, depth reached 0,
errors: 0
        1 states, stored
        0 states, matched
        1 transitions (= stored+matched)
        0 atomic steps
hash conflicts:         0 (resolved)
```

We used the verification function of SPIN to run the aforementioned two never claims and we have achieved valid results, which means that our two statements hold under some reasonable assumptions. Part of the verification result is shown in the above verbatim table.

V. Discussion

So far, we explored DTN mainly on its data transport capability, namely, the reliable data transfer by custody and hop-by-hop delivery. Since the whole Internet has not become delay-and disruption-tolerant, specific application gateways are still needed to translate between packets and bundles, such as the gateway on board and the gateway in the onshore office. The DTN design was chosen also because of its ability of integrating heterogeneous wireless networks, e.g., satellite and terrestrial networks, by deploying a common bundle layer into their protocol stacks. The drawback is that the deployment of a bundle layer into existing protocols is a non-trivial job. However, as long as we have the DTN gateway on board and on shore respectively, and we assume that a DTN gateway is available or will become available in the foreseeable future in the challenged satellite network, DTN-based data management will be deployable, for both reliable transfer and user preferred delivery in the maritime mobile environment. Following is a flow chart of the data migration process for the on-board DTN gateway. We can see that the final data delivery is either bundle delivery in the satellite network or packet delivery in a non-satellite network. It means that the bundle delivery in a satellite network will guarantee reliable data transfer and the packet delivery in non-satellite networks does not require DTN support in these networks.

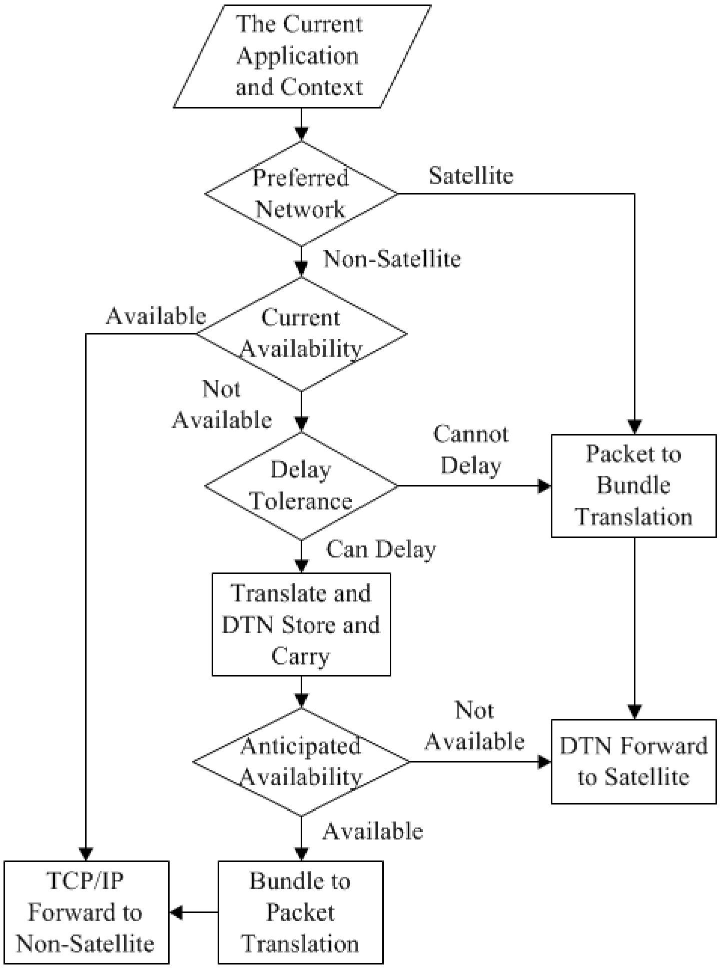

Figure D.7: Flow of the DTN-Based Data Migration Process On Board.

VI. Conclusion and Future Work

In this paper, we introduced a delay-oriented data traffic migration strategy based on the DTN architecture to leverage the predictability of maritime communication opportunities and the delay tolerance of many communication applications. We presented proof, through model checking, that a DTN approach can provide not only reliable data transfer but also user preferred data migration to the maritime customers. As the basic logic is simulated and verified before any real implementation, further deployment is justified in a way that can involve more maritime customers to be interested in it. The technical proposal will enable them take advantage of heterogeneous communication systems and utilize preferred networks as much as possible. However, further testing and implementation are needed:

- Conduct an investigation about the coverage range and other characteristics of satellite and WiFi networks at ports in maritime countries like Norway.
- Implement the DTN-based data migration function above some test-bed wireless networks. Main sub-functions are the context-aware data migration decision making, the protocols translation and the migration process execution.

References

[1] L. Mu, R. Kumar, and A. Prinz, “An integrated wireless communication architecture for maritime sector. in MACOM, 2011, pp. 193-205.

[2] K. Fall, “A delay-tolerant network architecture for challenged internets, in *Proceedings of conference on Applications, technologies, architectures, and protocols for computer communications,* 2003, pp. 27-34.

[3] J. Ott, “Delay tolerance and the future internet, in *Proceedings of the 11th International Symposium on Wireless Personal Multimedia Communications (WPMC),* vol. 26, 2008.

[4] L. Pelusi, A. Passarella, and M. Conti, “Opportunistic networking: data forwarding in disconnected mobile ad hoc networks, *IEEE Communications Magazine,* vol. 44, no. 11, pp. 134-141, 2006.

[5] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, G. Pei, and A. Srinivasan, “Cellular traffic offloading through opportunistic communications: a case study, in *Proceedings of the 5th ACM workshop on Challenged networks,* 2010, pp. 31-38.

[6] S. Trifunovic, B. Distl, D. Schatzmann, and F. Legendre, “Wifi-opp: ad-hocless opportunistic networking, in *Proceedings of the 6th ACM workshop on Challenged networks,* 2011, pp. 37-42.

[7] A. Balasubramanian, R. Mahajan, and A. Venkataramani, “Augmenting mobile 3G using WiFi, in *Proceedings of the 8th international conference on Mobile systems, applications, and services,* 2010, pp. 209-222.

[8] C. Caini, R. Firrincieli, M. Marchese, T. Cola, M. Luglio, C. Roseti, N. Celandroni, and F. Potort I, “Transport layer protocols and architectures for satellite networks, *International Journal of Satellite Communications and Networking,* vol. 25, no. 1, pp. 1-26, 2007.

[9] G. Giambene and D. Miorandi, “A simulation study of scalable TCP and High-Speed TCP in geostationary satellite networks, *Telecommunication Systems,* vol. 30, no. 4, pp. 297-320, 2005.

[10] J. Border, J. Griner, G. Montenegro, Z. Shelby, and M. Kojo, “Performance enhancing proxies intended to mitigate Link-Related degradations, 2001.

[11] C. Caini and R. Firrincieli, “Tcp hybla: a tcp enhancement for heterogeneous networks, *International Journal of Satellite Communications and Networking,* vol. 22, 2004.

[12] C. Caini, P. Cornice, R. Firrincieli, and D. Lacamera, “A DTN approach to satellite communications, *IEEE Journal on Selected Areas in Communications,* vol. 26, no. 5, pp. 820-827, 2008.

[13] G. Giambene and S. Hadzic, “A cross-layer PEP for DVB-RCS networks. in *PSATS,* vol. 15, 2009, pp. 12-19.

[14] G. J. Holzmann, “The model checker SPIN, *IEEE Transactions on Software Engineering,* vol. 23, no. 5, pp. 279-295, 1997.

[15] “On-the-fly, ltl model checking with spin. [Online]. Available: http://www.spinroot.com

[16] C. Caini, R. Firrincieli, and M. Livini, “DTN bundle layer over TCP: retransmission algorithms in the presence of channel disruptions, *Journal of Communications,* vol. 5, no. 2, 2010.

Appendix E

Paper E - Overlay-based On-Board Communication

Title:	A Hybrid Network for Maritime on-board Communications
Authors:	Liping Mu
Affiliation:	Department of Information and Communication Technology, University of Agder, N-4898 Grimstad, Norway
Conference:	*The 8th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Barcelona, Spain, 08-10 October 2012
Copyright ©:	Academy Publisher

A Hybrid Network for Maritime on-board Communications

Liping Mu
Department of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway
E-mails: {liping.mu}@uia.no

Abstract — **Current maritime on-board communications have to be enhanced for safety and security, where ubiquitous technologies can help, together with providing comfort and convenience to the crew and passengers. Employing wireless sensor networks on board is one recent practice of implementing ubiquitous technology for ships, where further study is needed because of the connectivity challenges. Meanwhile, it is important for an on-board communication system to be reliable and flexible for handling emergency situations. In this paper, we propose a solution of employing ubiquitous technology on ships in a way that both connectivity and emergency handling are examined. Two key aspects of this proposal are: 1) combine different on-board communication networks into a hybrid system, enabling the cooperation between them, 2) smoothly integrate these networks with a consideration of backward compatibility, ease of deployment and connection to shore via Internet. Simulations are used to support our choice between the separate networks and a hybrid one from comparing mainly the communication performance, and a combined integration approach is selected based on comprehensive analysis.**

I. Introduction

Communication is essential for maritime safety, security, integrated operations and for infotainment purposes. Communication of a ship can be categorized based on the route between source and destination, e.g., ship-to-shore, ship-to-ship and the on-board case. In this paper, we focus on maritime on-board communication which provides basic data for the other two types. User groups involved in the local communication include on-board equipment, cargo elements, crew and passengers. Communication between them are supported by the on-board infrastructures and services, such as a sensor network, a personnel tracking system and a wireless local area network (WLAN). Due to the harsh environment of ships for wireless communication and the cost consideration, current communication solutions on board are not satisfying. For example, most ship owners nowadays have only a monitoring system covering very essential equipment based on wired sensor networks, and a personnel tracking system is seldom in place. Internet connection via a WLAN on board is available merely in cruise ships and ferries. Safety and security can be much more enhanced if more intensive monitoring on a larger amount of user groups on board is achieved. Since deploying full-scale wired sensor networks on a ship leads to complexity and high costs, in [1], a Wireless Sensor Network (WSN) solution was proposed. After that, more papers [2] [3] have reported the experimental results of implementing WSN technology on ships. Although feasibility can be justified from the literature, real deployments still meet difficulties.

As we know that neither a wired or wireless solution of deploying sensor networks on board will function well alone, to combine their strengths together becomes an intuitive thought. This idea was implemented in [4] [5] for an energy-efficiency purpose. We apply it to the ship application mainly for tackling the connectivity challenges and for increasing system reliability. Therefore, in addition to wires, we also integrate the sensor network with other more established or to-be-established networks, such as a WiFi-based mesh network, the crew/passenger network, a personnel tracking system, and the global Internet. Reasons for involving human beings and Internet into the picture are explained as follows. Sensor monitoring systems on board are used for detecting abnormal operations, disordered equipment and for fire prevention. Rapid human response is usually required if any disorder or abnormality is detected. In this sense, it is preferable that crew carrying mobile devices around can participate in the monitoring process anytime and anywhere. On the other hand, it is desirable that crew-carried devices can perform mobile sensing and data collection as well, especially during system failures. If this on-board monitoring system can be seamlessly integrated with Internet, an ultimate goal of future maritime shipping - integrated operations for ships - will become feasible, where functions and personnel can be relocated from ship to shore based on efficient land-based control, remote maintenance, real-time surveillance and so on.

In order to test that the integrated hybrid on-board network will increase communication efficiency and system reliability, we use the ONE simulator [6] to evaluate some common scenarios, i.e., data collection in a monitoring sensor network, information dissemination for personnel tracking, and direct communication among people during emergencies. In addition, we investigate different methods of integrating WSNs with external networks, and present our selection based on thorough analysis considering both deployability and the particular maritime context. The remainder of this paper is structured as follows. In Section II, related work is discussed. Section III describes some application scenarios for using WSNs on ships. Section IV presents an integrated hybrid on-board network solution, together with a discussion about different integration approaches and our choice. Simulation and evaluation on the proposed hybrid network are presented in Section V. Finally, Section VI concludes this paper and points out directions for further work.

II. RELATED WORK

Maritime communication is important in many aspects. One example is integrated operations for ships. The Norwegian Oil Industry Association has defined integrated operations in the oil and gas industry as ”real time data onshore from offshore fields and new integrated work processes”. An ultimate goal of integrated operations is to relocate personnel and functions from offshore to onshore, where communication plays a fundamental role. A traditional maritime communication solution can be described as a wired on-board monitoring system connected to a local area network which interacts with the Internet through satellites, shown in figure E.1.

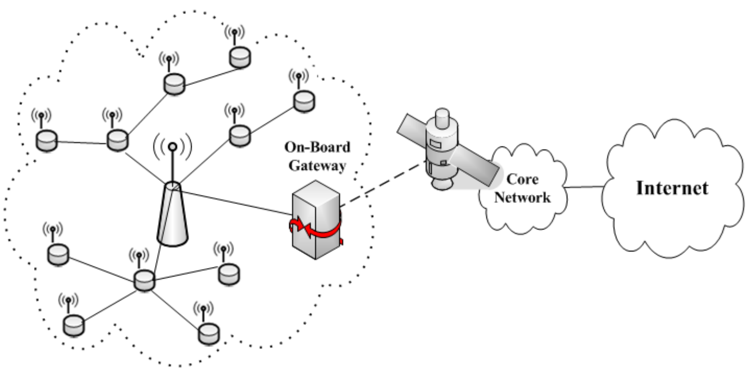

Figure E.1: The Traditional Maritime Communication Approach.

In papers [7] and [8], the ship-to-shore communication has been explored further by using multiple wireless access networks. Vertical handover among these networks based on the IEEE 802.21 standard is described in paper [7], and a supplementary capability of Delay Tolerant Networking (DTN) targeting at challenged satellite communication is depicted in paper [8]. From figure E.2, we get a basic idea of how to enhance maritime communication from the ship-to-shore part.

In this paper, we move our focus from ship-to-shore communication to the on-board case. A current popular suggestion of improving communication on board is to utilize the WSN technology. WSNs are distributed systems, consisting of low-power devices with integrated computation, sensing and wireless communications. This integrated characteristic has popularized WSNs in a wide range of applications

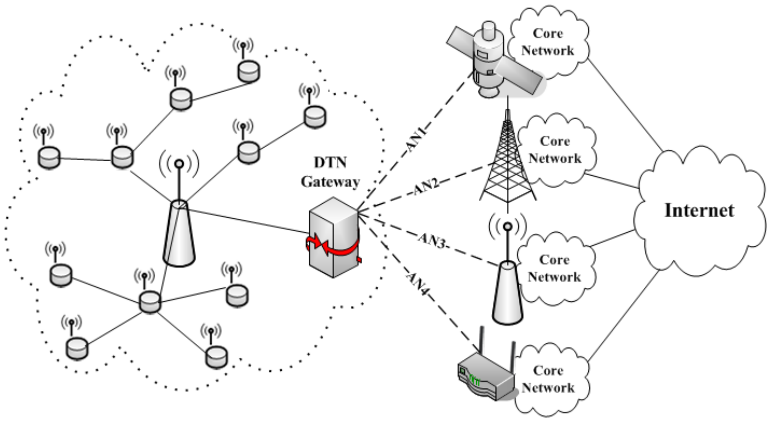

Figure E.2: Improved Ship-to-Shore Maritime Communication.

including operation monitoring, object tracking and detection. Different wireless communication technologies such as WiFi, Zigbee, and Bluebooth have been exploited for the construction of an efficient WSN. Due to resource constraints of a sensor node and the inherent feature of wireless communication, energy consumption and the connectivity problem are two major issues in WSNs and have attracted a lot of research work. In [9], a hybrid wired and wireless sensor network was adopted for energy efficiency. The used wires were acting as short cuts to bring down the average hop count, therefore reducing the energy dissipation per node and the energy consumption of the whole network. Compared with their work, we focus on addressing the connectivity problem when deploying WSNs on ships caused by steels and aluminium materials. Apart from energy efficiency, wires placed in some specific parts of the ship will function as connectors within an on-board WSN.

Connectivity problems in other traditional WSNs, such as a habitat monitoring system, are often introduced by mobility of the nodes and low node density. Therefore, increasing node density is one way to achieve connectivity, which is, however, not recommended in most cases due to the high cost. Deploying extra communication infrastructures like a base station is another way, e.g., integrating WSNs with a Wireless Mesh Network (WMN) for Internet connectivity, where the WMN acts as a wireless backbone. It is argued in paper [10] that a WMN should be used not only as the backbone but also for sensor node to sensor node communication. This deep-level interconnection applies to the situation on ships as well. However, the paper only considers the combination of WSNs with WMNs, in which, mesh nodes usually support limited or no mobility. We also add mobile nodes into the hybrid on-board network, because taking advantage of the mobility of some specific nodes is an efficient way to achieve connectivity in WSNs. These nodes are called data mules/ferries [11], and they have been well studied in the terrestrial WSNs scenarios but not in a ship environment. In this paper, we make use of both communication infrastructures and the crew carrying mobile devices with random or intentional mobility, to build an integrated hybrid sensor network on board. Communication infrastructures in the form of wires and WMNs are used here as an on-board core network, to address connectivity challenges, to reduce energy consumption and to avoid inefficient data routing. Introduction of the crew network consisting of mobile devices is a redundancy solution, aiming at the network reliability and robustness. Mobile devices behave as normal sensors under failure situations and as control centres when abnormality is detected, or they can function merely as relays.

Although networks integration, such as using WLAN and WPAN, together with WSNs on ships, is recommended in paper [2], for the adaptation to various environmental conditions, the paper does not specify how to integrate a sensor network with other, e.g., well-known TCP/IP networks. Integrating a WSN with external TCP/IP networks is not a trivial job. Numerous work has been dedicated to this. Basically there are three types of integration, a proxy-based solution, a DTN approach, and directly using TCP/IP stack in sensor nodes. Implementing a specific

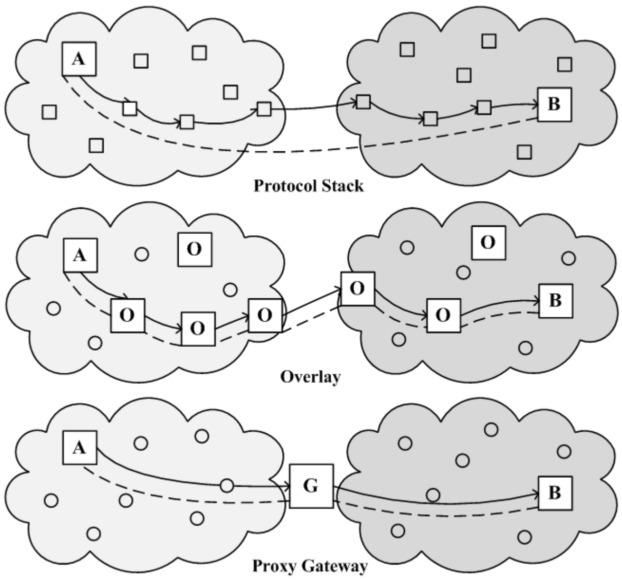

Figure E.3: Three Different Integration Methods.

proxy gateway [12] [13] between a sensor network and an external network is the most common way of doing integration, because of its no need of changing existing sensor networks or the external one. A DTN [14] architecture expands the proxy method by deploying a common Bundle layer as an overlay covering both TCP/IP network and non-TCP/IP network protocol stacks. Based on these overlay nodes, late address binding can be achieved independence of the underlying bearer protocols and addressing schemes. Directly running the TCP/IP protocol stack in the sensor network had been long argued as non-realistic until the micro IP (uIP) implementation [15]. After that, tremendous work has started following this direction, such as Tiny TCP/IP [16] and 6LoWPAN [17] related solutions. In paper [18], a complete IPv6/6LoWPAN stack for low-power wireless networks was presented. We make a comparison between these three methods from a deployment point of view, see figure E.3.

While implementing TCP/IP in sensor networks is considered as the ideal way of interconnecting sensor networks with TCP/IP networks seamlessly, and to unite all kinds of different sensor networks themselves, proxy-based approaches are the most realistic way to pursue integration because of their backward compatibility. Nevertheless, a specific proxy gateway only solves the problem of interconnecting a certain type of sensor networks with others and it is not a generic architectural approach. Therefore, in this paper, we make a combination of these three methods, in order to aggregate their strengths and to proceed integration smoothly. For example, the TCP/IP stack will be utilized in future deployed sensor networks; proxy gateways are added to existing non-TCP/IP sensor networks, and the DTN overlay architecture functions as a mediator when combining these two methods.

III. MARITIME ON-BOARD COMMUNICATION SCENARIOS

For monitoring abnormal operation or disorder of equipment

A ship has many types of equipment on board, and they are closely related to each other. A failure of any equipment may cause problems to other equipment or the main engine system, leading to abnormalities in the ship's operation. Therefore, it is important to monitor equipment as much as possible to reduce the risk of structural or engine failures. The current means of monitoring ship equipment are based on wired sensor networks, and the sensed data is collected and transmitted to the control server in the engine room in real time. Due to the high cost of cabling and the complexity of deploying a large wired sensor network on board, many ships today have a monitoring system only for their essential equipment [2], and the WSN technology has been proposed to improve this situation.

For environmental measurements

Crew and passengers wish to have better lives on ships, provided by ensured comfort living conditions, such as automatically controlled temperature, humidity, and similar environmental factors. Above comfort, safety on board has to be continuously guaranteed. External help is not easy to obtain if any crisis occurs on ships, e.g., fires and explosions. Therefore, it is of high significance to prevent emergency situations from happening by assessing their possibilities of occurring before their actual occurrences. This can be achieved by real-time environmental monitoring of dangerous areas and suspected containers or cargoes (for cargo ships). WSN technology is recommended here for monitoring areas like walls or above the ceiling or for a large amount of cargo containers. Optimization is necessary for efficient communication under complex structures inside ships, before any real implementation.

For personnel tracking and locating

Surveillance of the positions and movements of crew on board is a requirement for safety and efficient operations. It is especially important when crew members are sent to dangerous areas for maintenance or other tasks. The surveillance of passengers can help alarm them when they are close to a dangerous zone, or for counting passengers while they pass from one deck or territory of the ship to another. The most well-known ubiquitous technology that is currently in use for tracking personnel consists of Radio Frequency Identification (RFID) tags and readers. Cost is high for a RFID-based monitoring system being installed independently on shipboard, especially if the accuracy requirement is high. It is also not easy to reliably recognize an RFID tag carried by a highly mobile entity, and the equipment cannot last long in harsh environments. Therefore, if a personnel monitoring system can be based on the WSN technology and be integrated with other existing data networks, the communication efficiency and reliability will be increased at no extra cost. Meanwhile, considering the fast development of modern mobile devices, in the foreseeable future, a personnel tracking system will be able to employ personal mobile devices directly. It is more convenient to deploy such a system than an independent RFID-based one, and useful for emergency situations, where crew and passengers can self organize into a mobile ad-hoc or purely opportunistic network for information dissemination.

IV. Integration into a Hybrid On-Board Communication Network

A Hybrid On-Board Network

In the previous section, we listed some common scenarios of implementing ubiquitous technology on board a ship for local communications. We focused on the WSN technology. Since the network performance of a WSN on board depends strongly on the deployment of sensor nodes in real environments, which varies from ship to ship, it is preferable that more general optimization technologies can be applied, before any real-life implementation. Our method is to increase communication efficiency and system reliability by increasing system diversity from integrating different strategies, e.g., involving communication infrastructures (wires and WiFi nodes) and taking advantage of mobility of some specific nodes (crew carrying mobile devices). This way, a hybrid on-board sensor network, see figure E.4, is formed including wireless sensor nodes, wired sensor nodes, mobile nodes and WiFi nodes. This hybrid network can be used to cover all scenarios mentioned earlier.

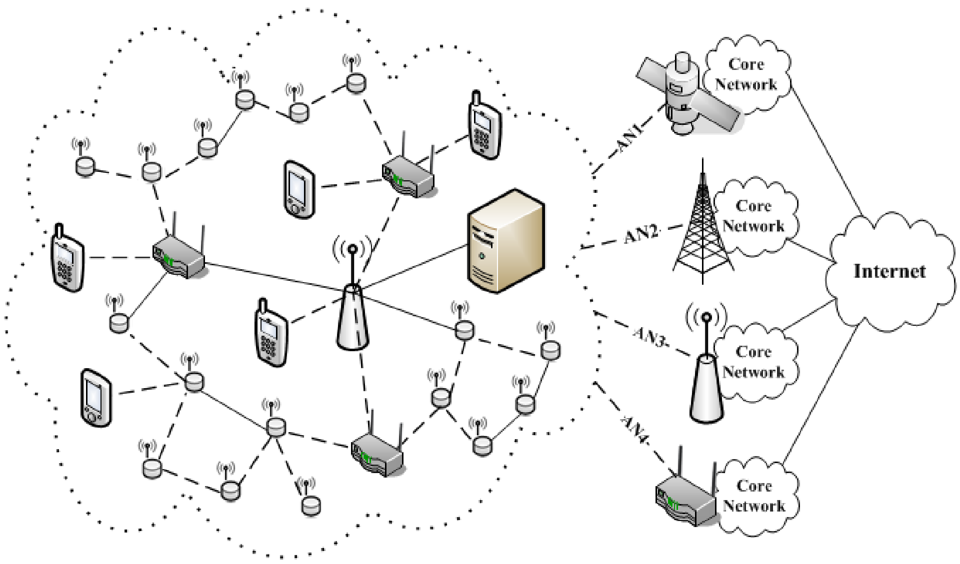

Figure E.4: The Future Maritime Communication Approach.

For a monitoring process to work, a sensor network needs to be connected to a central entity that can access and analyse the data. In our hybrid network (figure E.4), the sensor network is connected to multiple monitoring entities via multiple means, upon a WiFi-based network which directly attaches to an on-board control center, connecting to mobile devices carried by crew members which can behave as control points themselves, and to the global Internet through ship-to-shore communications. We explain this integrated on-board network in more detail by using following three individual networks.

(1): Figure E.5 shows a basic WSN with focusing on the communication infrastructure support from wires. Communication technology within this network includes wired transmission, ZigBee and/or Bluetooth and/or WiFi short-range transmission. A sensor node in the figure can be a source of data, an intermediate node to forward data or a sink node for collecting data. Data is delivered between a sensor and a sink based on either single-hop or multi-hop routing. After the initial deployment, this network is supposed to be fully connected and the data is transmitted in real time, whereas the reliability is probably low and there may be unanticipated disturbances or interferences in the future. The main purpose of this sensor network is to monitor abnormal operation, disorder of equipment and for the environmental measurements.

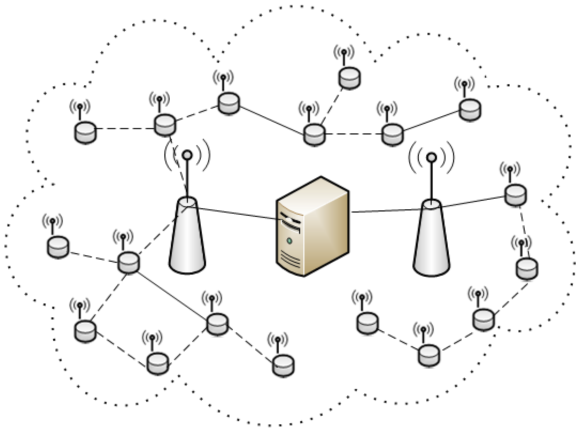

Figure E.5: A Basic Sensor Network.

(2): Figure E.6 shows a WiFi-based WMN that runs as a core network connecting on-board user groups to the on-board control center. Mesh nodes in a legacy WMN case, behave as a wireless backbone to extend network coverage, e.g., connecting an edge sensor network to the Internet via a sink. In our case, we seamlessly integrate these mesh nodes with the on-board WSNs by adding the sensor node capability to them. This way, a mesh node will have a twofold role: 1) as a powerful relay node to deliver large amounts of data between sensor nodes and 2) as an access point being contacted by any WiFi devices. Seamless integration is introduced

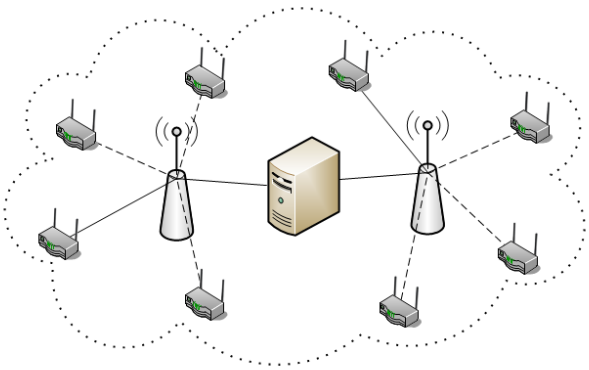

Figure E.6: A Mesh Core Network.

also because of situations such that data from sensors is needed by a nearby crew member carrying a smartphone. The only possible path in a traditional WSN is through the control server in the engine room, which leads to very inefficient data routing. If the mesh core network is interconnected with the sensor network seamlessly, data can be obtained efficiently via nearby mesh nodes. We go a step further in this direction by expecting that future mobile devices will be able to contact sensor nodes directly even without the help from an on-board core network, which will be explained in the next paragraph.

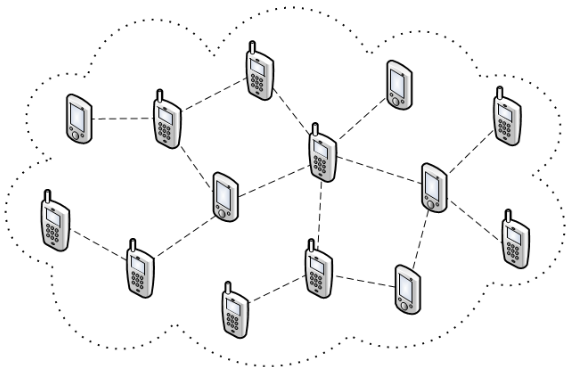

Figure E.7: A P2P Crew/Passenger Network.

(3): Figure E.7 shows an opportunistic P2P communication network formed by crew members and passengers. Modern mobile devices with WiFi direct, Bluetooth, or other communication technologies can be used for network construction. The network has an opportunistic networking capability [14], which means that data delivery can be carried out in a store-and-forward manner among opportunistic encountered nodes. Therefore, data transmission in this network is not based on real time but has possibly some delays. Such a network is useful in many circumstances, e.g., for personnel tracking, for cargo monitoring and for direct communication among passengers. If this network can be seamlessly integrated into the previous two networks, crew members with random or intentional mobility will be able to assist on-board sensing and equipment monitoring anytime and anywhere. A crew-passenger tracking system will be able to use mobile devices directly and cooperate with encountered sensor nodes or mesh infrastructure nodes. Direct communication requirements from crew and passengers can be satisfied, which is very useful for emergencies, e.g., when facing with communication infrastructure damages. Furthermore, the ship-to-shore communication attachment with the Internet is no longer a single point upon an on-board gateway, but many points through different mobile devices. This crew (passenger) network and the previous mesh core network are self-organizing networks, which facilitates fast built-up processes and changeability handling. Therefore, we can deploy them in a flexible and incremental manner according to different ship interiors and different user requirements.

An Integration Roadmap

An integrated hybrid network on board shall have increased system reliability and better network performance, which we will test in the simulation section. But before that, it is necessary to answer the question of how to integrate different networks. Given that the TCP/IP protocol suite is the de-facto networking standard both for the global Internet and for local-area networks, integration between sensor networks and the TCP/IP world is of major importance. As mentioned in related work, there are basically three approaches, and we make a comparison between them under the maritime context.

A proxy-based integration architecture relies on sink nodes to provide protocol bridging between the sensor network and the external one, which can be implemented by executing either protocols translation or stack virtualization on mesh nodes on board. It has the advantage of keeping specialized sensor nodes with almost no disturbances to the network. Therefore, a proxy-based architecture is very suitable for already deployed on-board sensor systems with dedicated protocols. As the development of hardware and software technologies, a sensor node will have more resources and more powerful computing capability. At the same time, most mobile devices in the future will have global unique IPv6 addresses and the Internet shall provide transparent pervasive accessibility and mobility to users. Therefore, from a long-term point of view, implementing TCP/IP in sensor networks is an ideal way of interconnecting sensor networks with others [16]. It is also the trend of integrated operations for ships, which requires seamless interactions between on-board equipment, devices and the on-shore control centres. Since existing TCP/IP architectures can not flexibly handle the case of different resource conditions of sensor nodes, and mobility support is lacking, they must be integrated into a larger architecture, e.g., the DTN architecture. A DTN-based integration architecture is similar to the proxy-based approach, but provides general mechanisms and an interface that can be used for more occasions. With a DTN architecture, it becomes easy to integrate different heterogeneous wireless networks from deploying a common DTN Bundle Layer into their protocol stacks. Deploying a Bundle layer into existing protocol stacks involves activities towards both the lower and application layer. Therefore, in our case, we allow the network to have a DTN networking capability gradually, e.g., first the on-board proxy gateways, then the mobile devices carried by crew, then more infrastructure nodes, and then specific sensor nodes. Besides the architectural support for integration, DTN also provides a set of features which can be used to overcome issues within problematic communication environments, i.e., the maritime communication. We can explain this from the following two aspects:

- *First, a local on-board WSN meets a lot connectivity challenges because of the materials and the structure used for building a ship. Participation from the crew and passengers carrying mobile devices, supported with the DTN* *networking capability, can help mitigate the communication problem and increase the system reliability.*

- *Second, it is anticipated that on-board networks will be connected to Internet via the ship-to-shore communication, and most of the time, ship-to-shore communication has to rely on challenged satellite networks. DTN is recommended to be used in this context [19], and it enables user preferred data migration from satellite to other terrestrial networks as well [8].*

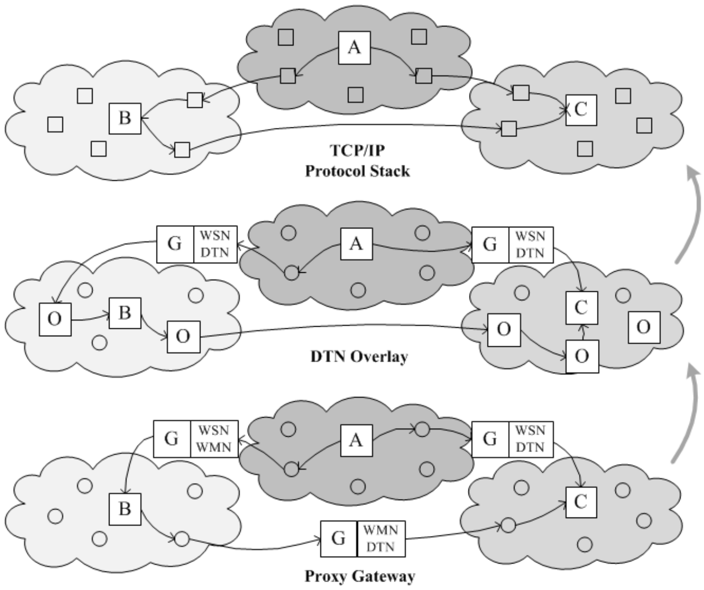

Figure E.8: The Integration Roadmap.

Therefore, we suggest an integration roadmap for maritime scenarios based on a combination of different methods using the DTN networking architecture as a transition phase, see figure E.8. Future on-board WSNs can be deployed based on the TCP/IP protocol stack, and they communicate with an on-board DTN gateway using local area networks. The DTN gateway will transmit the gathered information over the global Internet at places where it obtains preferred Internet access, e.g., WiFi at ports. This on-board DTN gateway is either a static server or a mobile device carried by a crew member. Thanks to the special position of a proxy gateway, it can be placed where traditional sensor networks exist with dedicated protocols, or where the sensor resources are extremely constrained and performance requirements are high, by running some specific code on it [20].

V. SIMULATION AND EVALUATION

Different ships often have different interiors and inner structures based on the owner's initial requirements, and these initial designs can change with time. Since communication within a WSN depends heavily on the actual environment, it is not trivial to generally simulate the connectivity challenge of deploying WSNs on ships. However, as a connectivity problem of a WSN can be caused not only by the actual surroundings, but also by mobility of the sensor nodes, we can simulate the connectivity challenge of deploying WSNs on board by adding random mobility to the nodes. This way, simulation is used to support our arguments from a more general perspective. The primary input parameters for the simulation are: the node density, the mobility model and the routing protocol. We use node density as an input variable and set random walk mobility to all nodes. Routing protocols are chosen according to the scenarios. Usually delivery rate and delivery delay are two main output parameters for the evaluation. Since we simulate the connectivity challenge by adding mobility to all nodes and data is based on a hop-by-hop non-real-time delivery, the delay parameter will lose its original value. It can be assumed that the real-life sensor network implementation is mostly real-time data transmission, and crew assisted data delivery is a redundancy solution which can tolerate some delays. Other scenarios such as the personnel tracking, cargo monitoring and direct communication among people are delay-tolerant applications. Therefore, we compare only the delivery rates collected from the simulation which is executed in three cases.

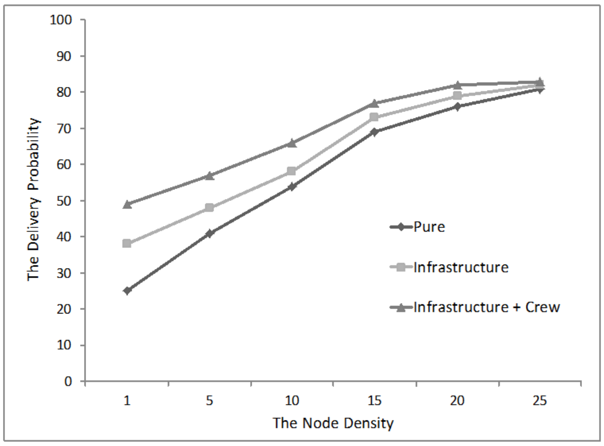

Figure E.9: For the Basic Sensor Network Scenario.

Case 1 is a scenario of WSNs on board for monitoring abnormal operation or disorder of equipment, and for environmental measurements. A pure WSN on board is simulated as a mobile network with all nodes having the same characteristics, whereas the proposed hybrid network can behave as including some special nodes.

Special infrastructure nodes have much larger transmission ranges and higher transmission speeds. Other special crew nodes will have higher mobility, added by the human walking speed. Regarding the routing protocol to be used in this network, we exploit the first-contact protocol from the ONE simulator, routing only a single copy of data to simulate the real-time ad-hoc data transmission in a discrete manner. Similarly, as all nodes have been configured with random mobility, a geographically fixed control center will not exist. Therefore, it is not necessary to have one single destination in the network for the data collection scenario, as long as the messages are unicast-based. Figure E.9 shows simulation results on the delivery ratio of data transmission in a pure WSN, a WSN with infrastructure nodes and the one with both infrastructure and crew nodes support. We can see that the higher node density, the higher delivery probability for all three networks. But if the node density is low, which represents the connectivity challenges aboard ships, roles played by communication infrastructures and the crew nodes are apparent.

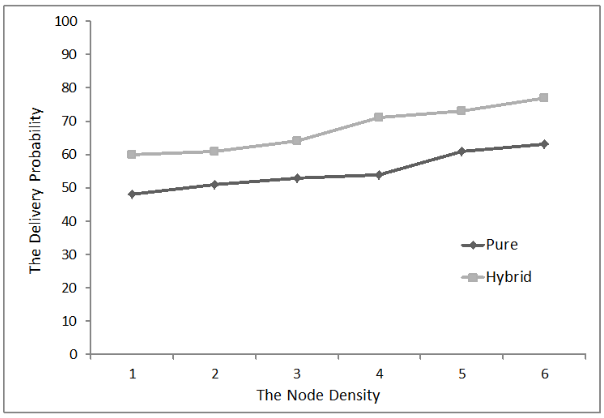

Figure E.10: For the Personnel Monitoring Scenario.

Case 2 is for the personnel monitoring. A traditional personnel monitoring system that uses RFID tags and PDA readers can be considered as a mobile network with low involvement of communication infrastructures. Personnel's coordinates are only available when they pass places where tag readers are installed. As the fast development of modern smartphones, e.g., being conjunction with RFID devices, future personnel tracking will be able to use personal devices directly. It is also possible for a future on-board personnel monitoring system to be integrated with other data networks due to the simultaneous location and data transmission capability of modern ubiquitous technologies. We can simulate the future integrated personnel monitoring system as the network having a much higher node density; data can be routed in an epidemic manner under critical conditions and message copies are reduced in a normal tracking case. Senders and receivers are personnel nodes which have higher moving speeds but an overall lower density, compared with message senders and receivers in case 1, but the density of the relay nodes is much higher. From the simulation results shown in figure E.10, we see that a future hybrid personnel tracking system behaves better than a pure one in terms of successful message delivery upon different node density settings.

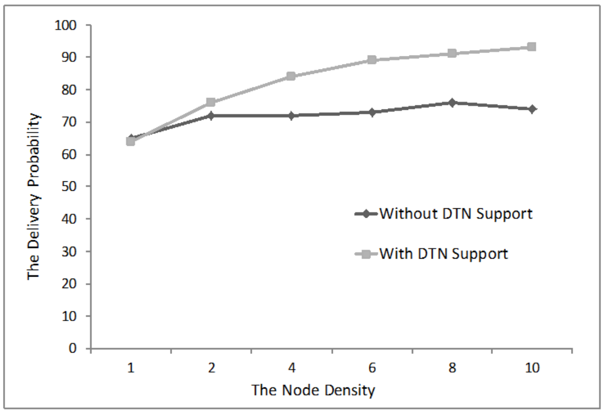

Figure E.11: For Direct Communication and Emergencies.

Case 3 simulates a situation where direct communication among the crew or passengers are required. Direct communication can happen frequently in the daily life for an entertainment purpose. It is useful for the personnel tracking, and it can be very critical in emergency situations. Therefore, it will be necessary for the on-board network to enable direct communications, supported by a DTN networking feature. A traditional on-board network without a DTN networking support will not function well if the communication infrastructures are damaged, which can be simulated by only a direct-delivery routing protocol being available. The hybrid DTN-based network can be treated as supporting a flooding-based routing mechanism, e.g., the spray-and-wait routing protocol. Results shown in figure E.11 tell us that if the node density is high, such as in passenger ships, an on-board network with the DTN networking feature will function better for direct communication situations.

VI. Conclusion and Future Work

In this paper, we explained the reasons for applying ubiquitous technology particularly the wireless sensor network to a ship environment and the challenges of doing so. A hybrid on-board network which consists of diverse communication systems was introduced to address the challenges and to handle emergency situations. We investigated different means of integrating wireless sensor networks with the TCP/IP world due to its popularity. From thorough analysis, we suggested a gradual integration roadmap for the ship application, based on a combination of three popular methods, using DTN overlay architecture as a transitional step to achieve the final seamless integration. We also implemented simulation to prove that the hybrid network can provide better communication performance and increased reliability than separate systems, when faced with connectivity challenges or infrastructure damages. However, future testing and implementation are needed especially in following directions:

- Develop DTN applications for maritime scenarios, such as data collection within on-board WSNs, smartphone-based personnel tracking and direct communication among crew and passengers
- Provide a common standard middleware layer to facilitate the communication between different DTN applications and the underlying data transmission, to further enable future application development

REFERENCES

[1] S.-R. Cho, D.-K. Lee, B.-G. Paik, J.-H. Yoo, Y.-H. Park, and B.-J. Park, "A study on USN technologies for ships, in *Proceedings of the 4th international conference on Ubiquitous Intelligence and Computing,* 2007, pp. 392-401.

[2] B.-G. Paik, S.-R. Cho, B.-J. Park, D. Lee, B.-D. Bae, and J.-H. Yun, "Characteristics of wireless sensor network for full-scale ship application, *Journal of Marine Science and Technology,* vol. 14, pp. 115-126, 2009.

[3] H. Kdouh, G. Zaharia, C. Brousseau, G. E. Zein, and G. Grunfelder, "ZigBee-based sensor network for shipboard environments, *CORD Conference Proceedings,* pp. 1-4, 2011.

[4] R. Chitradurga and A. Helmy, "Analysis of wired short cuts in wireless sensor networks, in *Proceedings of the The IEEE/ACS International Conference on Pervasive Services,* 2004, pp. 167-176.

[5] G. Sharma and R. R. Mazumdar, "A case for hybrid sensor networks, *IEEE/ACM Trans. Netw.,* vol. 16, no. 5, pp. 1121-1132, 2008.

[6] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation, in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques,* 2009.

[7] L. Mu, R. Kumar, and A. Prinz, “An integrated wireless communication architecture for maritime sector. in *Proceedings of the 4th International Workshop on Multiple Access Communications,* 2011, pp. 193-205.

[8] L. Mu and A. Prinz, “Delay-Oriented data traffic migration in maritime mobile communication environments, in *Proceedings of the Fourth International Conference on Ubiquitous and Future Networks,* 2012.

[9] G. Sharma and R. Mazumdar, “Hybrid sensor networks: a small world, in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing,* 2005, pp. 366-377.

[10] S. Bouckaert, E. De Poorter, B. Latre, J. Hoebeke, I. Moerman, and P. Demeester, “Strategies and challenges for interconnecting wireless mesh and wireless sensor networks, *Wirel. Pers. Commun.,* vol. 53, no. 3, pp. 443-463, 2010.

[11] R. C. Shah, S. Roy, S. Jain, and W. Brunette, “Data mules: Modeling a three tier architecture for sparse sensor networks, in *Proceedings of IEEE SNPA WORKSHOP,* 2003, pp. 30-41.

[12] M. Z. Z and B. Krishnamachari, “Integrating future large-scale wireless sensor networks with the internet, 2003.

[13] K. A. Emara, M. Abdeen, and M. Hashem, “A gateway-based framework for transparent interconnection between WSN and IP network, in *IEEE EUROCON,* 2009, pp. 1775-1780.

[14] K. Fall, “A delay-tolerant network architecture for challenged internets, in *Proceedings of conference on Applications, technologies, architectures, and protocols for computer communications,* 2003, pp. 27-34.

[15] A. Dunkels, “Full TCP/IP for 8-bit architectures, in *Proceedings of the 1st international conference on Mobile systems, applications and services,* 2003, pp. 85-98.

[16] G. Han and M. Ma, “Connecting sensor networks with IP using a configurable tiny TCP/IP protocol stack, in *Proceedings of the 6th International Conference on Information, Communications & Signal Processing,* 2007, pp. 1-5.

[17] G. Mulligan, “The 6LoWPAN architecture, in *Proceedings of the 4th workshop on Embedded networked sensors,* 2007, pp.78-82.

[18] J. W. Hui and D. E. Culler, "IP is dead, long live IP for wireless sensor networks, in *Proceedings of the 6th ACM conference on Embedded network sensor systems,* 2008, pp. 15-28.

[19] C. Caini, P. Cornice, R. Firrincieli, and D. Lacamera, "A DTN approach to satellite communications, *IEEE Journal on Selected Areas in Communications,* pp. 820-827, 2008.

[20] D. Bimschas, H. Hellbrück, R. Mietz, D. Pfisterer, K. Römer, and T. Teubler, "Middleware for smart gateways connecting sensornets to the internet, in *Proceedings of the 5th International Workshop on Middleware Tools, Services and Run-Time Support for Sensor Networks,* 2010, pp. 8-14.

Appendix F

Paper F - Integrated Operations for Ships

Title:	Towards Integrated Operations for Ships
Authors:	Liping Mu, Andreas Prinz and Frank Reichert
Affiliation:	Department of Information and Communication Technology, University of Agder, N-4898 Grimstad, Norway
Conference:	*Accepted by the 9th International Wireless Communication & Mobile Computing Conference (IWCMC)*, Italy, 01-05 July 2013
Copyright ©:	Academy Publisher

Towards Integrated Operations for Ships

Liping Mu, Andreas Prinz and Frank Reichert
Department of Information and Communication Technology, University of Agder (UiA), N-4898 Grimstad, Norway
E-mails: {liping.mu, andreas.prinz, frank.reichert}@uia.no

Abstract — **Maritime customers will benefit greatly from integrated operations between ship and shore. It requires integrating diverse information services within a variety of communication environments, which is a complex task. The common approach to dealing with complex tasks is to use a divide-and-conquer policy and solve problems independently. Since there are mainly two challenges involved - interoperability among heterogeneous applications and connectivity through difficult maritime networks, to handle them separately has become useful. While interoperability and connectivity solutions work well individually, if we put them together in the maritime context, problems arise. Web services based interoperability solution assumes continuous connectivity to the network, which is usually not available in the maritime environment. Though adapting applications to such network conditions will help, existing implementations are mostly specialized and non-generic. Therefore, negotiation between separate solutions is needed: 1) *applications* must adapt to different connectivity situations but in a service-oriented manner; 2) *networks* need to mediate the adaptiveness and follow the service-oriented trend. As a concrete example, we suggest to implement the negotiation via incremental deployment from wrapping existing mediation capabilities as Web services towards a possibly service-oriented network architecture, where mediation mechanisms and communication resources are standard services invoked directly by applications.**

Keywords—Integrated Operations, Wireless Connectivity, Service Interoperability, Negotiation, Mediation.

I. Introduction

In order to enhance safety and security of a ship and its freight, more intensive monitoring of the ship conditions is required. At the same time, comfort and convenience are expected by crew and passengers. Beyond these requirements is a vision of integrated operations for the ship, where functions and personnel can be relocated to shore based on efficient land-based control, surveillance, and management. According to the norwegian oil and gas association (http://www.norskoljeoggass.no), the term *integrated operations* in the oil and gas industry means real-time data onshore from offshore fields and new integrated work processes. For ship application, examples of integrated operations are real-time or near real-time updates of navigational data from a land-based assistance system, remote surveillance of on-board equipment and devices, and the like. Figure F.1 gives an abstract vision anticipated by integrated operations for ships, with benefits such as improved decision-making, higher efficiency and flexibility of ship operations, increased accuracy and consistency of information, optimized navigation, better ship monitoring, maintenance and resource exploitation, increased heath and safety, improved regulatory and legal obligation, etc.

However, turning this vision into reality is a complex task. Complexity involved includes integrating work process services which are heterogeneous, isolated, suffering monolithic functionalities and incompatible data formats, connecting equipment/devices located both on board and on shore in harsh communication environments. We can divide this complex problem into two primary challenges: 1) interoperability among heterogeneous applications and 2) connectivity through challenged maritime networks.

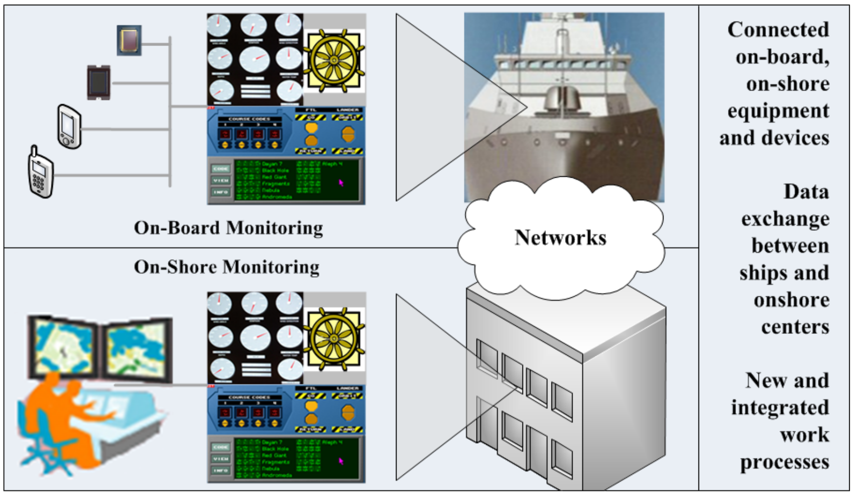

Figure F.1: The Vision of Integrated Operations for Ships.

For example, a ship is equipped with isolated sensor systems to capture operation data: GPS location (geographic coordinates), travel speed, engine temperature, engine fuel rate, fuel remaining from the tank, etc. In order to monitor fuel efficiency remotely, a monitoring system as a general-purpose IT system is running on shore. It needs to be fed with information captured on-line and from on-board sensors. However, as the sensor systems are independent, provided by different vendors and use different standards to represent and transmit data, interaction among them and the onshore monitoring system becomes difficult. Furthermore, crew carrying mobile devices desire to directly contact some sensors whenever they come within communication range, and there are requirements for new applications to be added to existing sensor systems. These are difficult to realize because of the interoperability issue caused by heterogeneous sensing applications.

Due to mobility and distances involved in maritime communication scenarios, connectivity is a challenge. So far, FM (frequency modulation) radio technology like narrowband UHF (ultra high frequency) and VHF (very high frequency) are widely used for the ship-to-shore and ship-to-ship communication, with satellites such as Inmarsat (international maritime satellite) and VSAT (very small aperture terminal) systems used for long-range cases. Compared with terrestrial networks, these systems show limitations in terms of transmission data-rate, transmission delay, bit error rate and communication cost. Although cellular networks and WiFi can be considered, they are only available for near port waters, and switching between these different networks is not trivial. Besides, a ship entity itself is a difficult environment for on-board communications, e.g., the steel material and ship structure impede deployment of local wireless sensor networks.

Regarding the interoperability problem, we used an ontology-based approach and Semantic Web technologies to integrate maritime data for enhanced information interoperability [1]. Integration at the application level is further recommended, where a common Service-Oriented Architecture (SOA) methodology [2] can be adopted. With respect to network connectivity, we proposed IP-based optimization mechanisms dedicated to the ship-to-shore case [3], and a supplementing strategy using Delay Tolerant Networking [4] for extreme challenging situations [5] [6]. Separate interoperability and connectivity solutions work well individually in their own contexts. However, if we combine them together for the integrated operations of ships, inconsistencies can be found. Therefore, reasons behind the inconsistencies need to be investigated, and compatibility between separate solutions must be ensured, which triggered the research work presented in this paper.

The rest of the paper is organized as follows. In Section II, separate solutions on addressing interoperability and connectivity problems are discussed. Section III explains the inconsistency issue between them and relevant reasons. Section IV presents a SOA-based negotiation approach following incremental deployment, with the intention to solve inconsistencies in a both backward-compatible and future-proof way. Finally, Section V concludes the paper.

II. Separate Solutions

Interoperability Solution

Integrated operations require frequent exchange of data and work processes between ships and onshore centers, where involved human interactions need to be minimized. To achieve that, information interoperability and application interoperability are desired.

Data Integration A data integration solution can help with information interoperability. Specifically in the database community, data integration refers to gathering and combining data residing in multiple heterogeneous data sources and presenting it in a unified view to users for transparent manipulations of information. As a case study, we used an ontology-based approach together with Semantic Web technologies to implement data integration in the Oil & Gas industry [1]. However, as data integration happens after the operation transaction and typically in either inter-day or intra-day batches, it does not apply to situations where real-time interactions are needed. Besides, more and more data is fragmented among distributed applications other than databases. Therefore, data integration alone is not enough any more for the integrated operations of ships. Integration at the application level has to be taken into consideration, and application interoperability needs to be ensured in order to respond to new data feeds, changes in logic, and new functions.

Application Integration Compared with data integration which focuses on the data level, application integration manages the transactions that operate on the data, for interoperability between applications or more generally among services. In the context of integrated operations of ships, an application integration solution should fulfill the following requirements: (1) expose a common standard mechanism through which applications can interact, (2) shield services from underlying different operating systems, programming languages, and other technologies, (3) allow for reuse of existing and newly developed services, and (4) offer backwards compatibility and migration to future solutions.

Service-oriented application integration as a possible approach has become essential in the software industry, especially the business domain where new complex applications are likely to base on the composition and collaboration of other services. Figure F.2 describes the Service Oriented Architecture (SOA) and a brokered implementation mechanism through a service bus. In a service-oriented view, the interoperability problem can be partitioned into two sub-problems [7], the definition of service interfaces and the identification of protocols that can invoke a particular interface. *Web services* use interfaces described in a machine-processable format and leverage standard Internet protocols for interaction, allowing reusable software

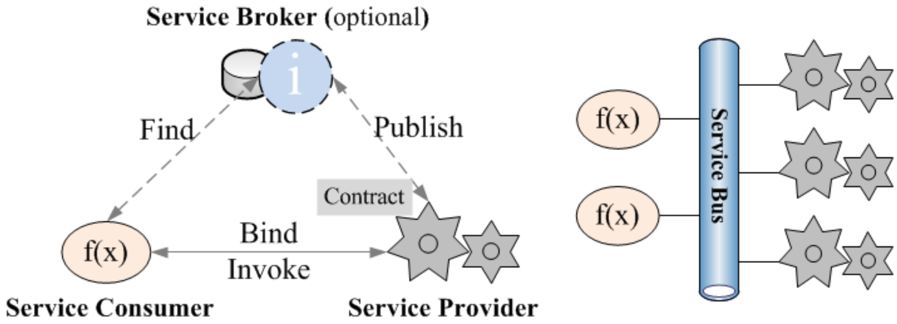

Figure F.2: The Service Oriented Architecture.

applications to be independent of operating systems, platforms and programming languages. Web services have been widely used for SOA implementation which can fulfill the aforementioned requirements (1), (2), (3), (4). Therefore, we expect that service-oriented integration based on Web services technology can function as the interoperability solution for integrated operations of ships.

Connectivity Solution

In order to integrate data and applications, which are distributed in different equipment/devices located both aboard a ship and on shore, connection between them is necessary. However, it is not easy to build or maintain the connectivity, because of mobility, inherent characteristics of wireless communication and diverse maritime networks. Hence, a communication integration solution is needed to: (A) provide applications with connectivity through heterogeneous and challenged networks, (B) shield applications from the underlying communication problems, (C) use communication resources efficiently and (D) allow for independent and incremental deployment.

Two major trends for dealing with connectivity challenges are worth mentioning, one is various IP-based solutions and another is overlay approaches [8][9]. IP-based solutions can be considered as different cross-layer optimization mechanisms based on a common network layer - IP - to adapt networks as much as possible to (traditional) Internet applications. Overlay approaches, on the other hand, enable applications to adapt to changing or difficult network conditions, based on specific overlay infrastructures.

IP-based Optimizations Given that the TCP/IP protocol suite is the networking standard for Internet and has contributed to its rapid expansion, a lot of efforts focus on IP-based connectivity with associated optimizations for challenging situations. These optimization solutions employ variants of incremental fixes and patches to adapt the communication environment as much as possible to what the respective applications expect [10], in order to keep them less impacted or unchanged. Different IP-based optimizations can be classified according to their different implementation layers, see figure F.3. Examples include HIP and mobile IP techniques for mobility handling, transport and session layer policies for QoS enhancement, etc.

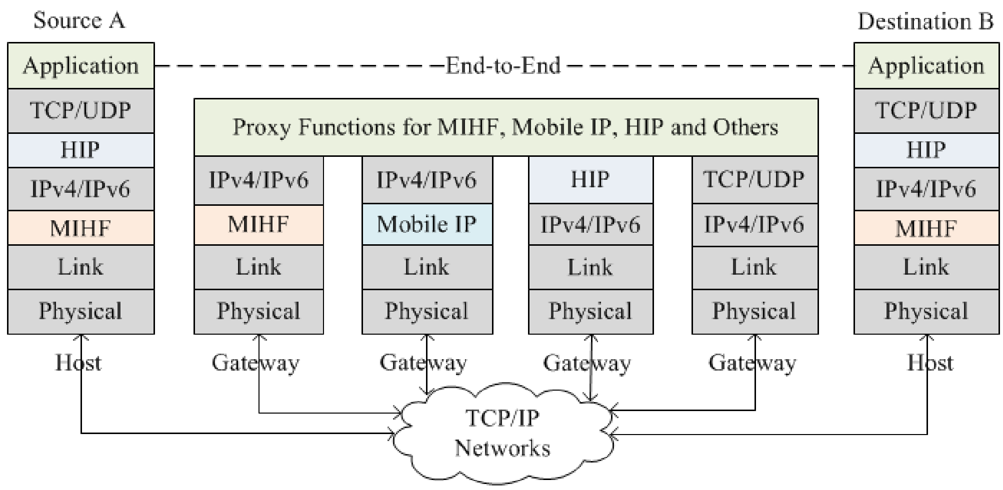

Figure F.3: The IP-based Solution with Optimizations.

Normally, we can use IP as the unifying technology to integrate heterogeneous ship-to-shore communication systems, and exploit the IEEE 802.21 standard together with a satellite extension to ensure seamless handover between them [3]. Optimized upper layer protocols can be further included for mobility and quality handling. While such approaches definitely improve connectivity, they are unable to address disconnections due to coverage gaps, missing roaming agreements, or user policy (e.g., minimizing access charges), and are difficult to apply to existing non-IP networks.

Overlay-based Approaches The aforementioned IP-based solutions focus on mitigating the risk of connectivity loss and maximizing the connection quality by hiding the particulars of the network from applications using abstraction. This abstraction works well as long as, from a higher layer perspective, connectivity is not lost for too long and the necessary end-to-end communication can be established [10]. However, this assumption does not hold for many maritime scenarios where satellite communication is dominant. Satellite communication usually has limited capacity, high propagation delay, and high bit error rate. There are special problems related to lack of satellite coverage in places such as fjords, ports and at high latitudes. Moreover, maritime customers tend to use other means to communicate rather than satellites due to the cost, especially when a large amount of data usage can be incurred [5].

Therefore, in addition to IP-based mechanisms, alternative solutions should be considered, e.g., overlay approaches, application-specific mobility and disconnection support via various application protocols, and new clean-slate designs. An overlay network is an application-specific network built on top of another network, which can be incrementally deployed on end-hosts running the overlay protocol software, without having to compete with the existing infrastructure. By contrast, a clean-slate approach is to redesign the network from scratch to offer improved abstractions and/or performance, while providing similar functionality based on new core principles.

Compared with the application-specific mobility or disconnection support, an overlay is relevantly a more general approach from the network perspective to tackling connectivity problems, and it can be implemented incrementally and independently rather than clean-slate designs. Therefore, we choose the overlay approach to supplement IP-based solutions for the maritime scenario. Specifically, a Delay Tolerant Networking (DTN) overlay architecture was explored because of its ability of handling both network heterogeneity and extreme connectivity challenges [5].

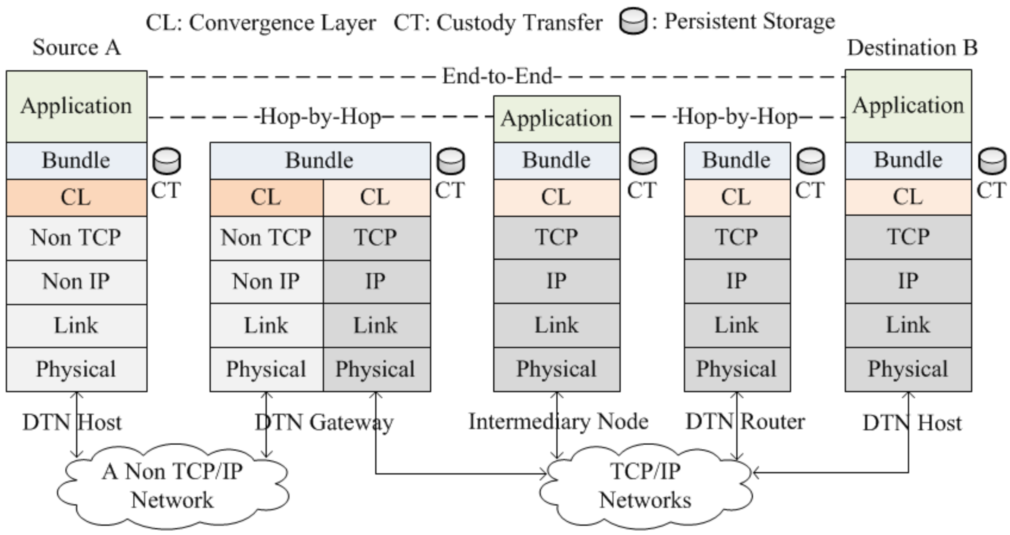

Figure F.4: The DTN Example of Overlay Approaches.

The DTN overlay architecture, see figure F.4, is capable of dealing with intermittent connectivity, via asynchronous message exchanges between decoupled end nodes, at an expense of losing interactivity. It can take advantage of other types of connection as well: scheduled, predicted, opportunistic and continuous, which is meaningful for ship applications because of ships' route repetitiveness and predictability. Furthermore, it is a generic approach to interoperability across radically heterogeneous maritime networks.

Therefore, we anticipate that a combination of IP-based optimizations and the DTN overlay can satisfy the aforementioned communication requirements (A), (B), (C), (D) and act as the connectivity solution to integrated operations for ships.

III. Inconsistency between separate solutions

After the separate interoperability and connectivity solutions are settled, the next step is to put them together under the maritime context. Within service integration that relies on Web services technology, the traditional TCP/IP protocol suite is usually exploited. When using the TCP/IP protocol suite, applications are tightly coupled with the communication mechanism. Applications must be responsible for establishing all bindings necessary to perform communication, without awareness of the nature of the underlying communication networks. As a result, applications have to assume the implicit design, conventions, and operating modes of the networking underneath, e.g., expecting continuous and end-to-end connectivity.

However, as IP-based networks have difficulties functioning well in extremely challenging maritime environments, alternative communication mechanisms are required. To that, we have proposed to take the DTN overlay approach. Although the DTN architecture can help provide better network connectivity under harsh conditions, limitations still exist [11], and modification of legacy applications and specialized implementations are often necessary. For example, due to the lack of deployment standardization, existing DTN applications are based on different development platforms, deployed in different operating systems and using different ways to communicate with DTN daemons [12]. Specialized DTN applications are not interoperable with each other or with traditional applications.

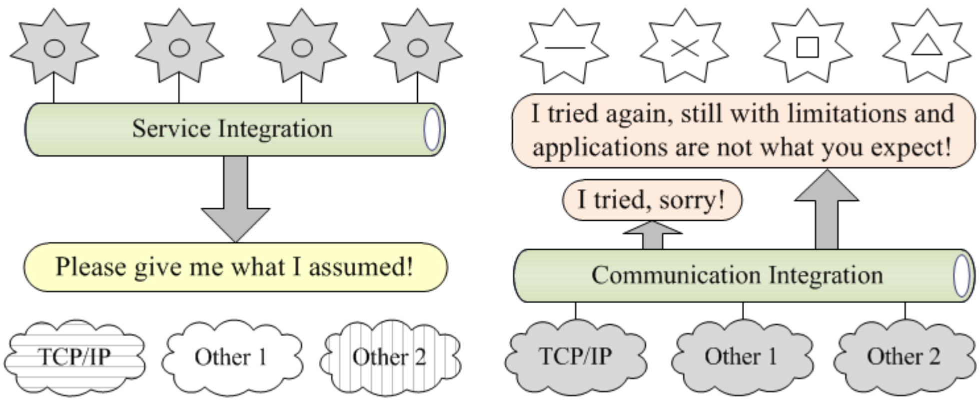

Figure F.5: Inconsistencies between the Separate Solutions.

Therefore, inconsistencies exist between the separate service integration and communication integration mechanisms, see figure F.5. In order to address the inconsistencies, negotiation between these separate solutions has to be performed and there are basically three aspects:

- *First, it is necessary for applications to be able to adapt to changing network conditions, responding to different connectivity options, protocols, etc., in addition to relying on the network to adapt to existing or legacy applications through incremental fixes or patches.*

- *Second, instead of letting each application adapt to different communication characteristics individually or specifically, there should be a generic mechanism in between applications and networks to mediate the interaction between them, decoupling applications from the underneath networking details.*

- *Third, the mediation mechanism should be provided in a way that newly developed or modified applications are interoperable with each other and with standard services, and future application requirements and emerging network technologies can be easily brought in.*

IV. NEGOTIATION BETWEEN SEPARATE SOLUTIONS

The first aspect of the negotiation is from the service integration point of view to agree to change. But the change cannot be specific or individual, it has to rely on the communication integration to perform mediation as mentioned in the second aspect. Furthermore, service integration must keep its SOA character, which requires the mediation mechanism to be implemented in a service-oriented fashion as well, namely the third aspect. Figure F.6 depicts the negotiation process.

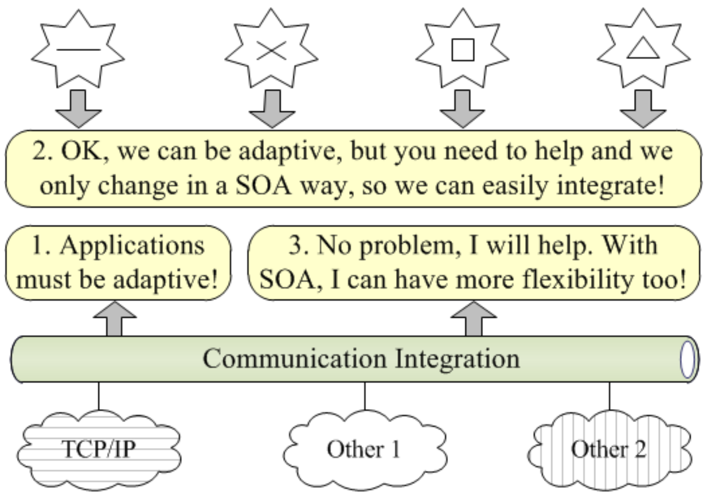

Figure F.6: Negotiation between the Separate Solutions.

In order to enable the applications to adapt to different communication conditions, following features must be realized from the network part:

1) allow fast detection and dynamic attachment to available resources
2) provide reliable transmission between network elements
3) support naming and late binding
4) ensure robustness in presence of intermittent connectivity, e.g., with enhanced routing mechanisms, etc.
5) support multiple classes of service
6) provide various communication patterns, unicast, multicast, anycast, etc.
7) facilitate content-, context- and network-awareness
8) enable cooperation via effective trust & cost and enhanced security & privacy mechanisms

If the implementation should follow a SOA methodology, the easiest way of imaging that would be to deploy a communication mediator that provides all the aforementioned network features to applications as standard services in a layer-less manner, see figure F.7. This is very similar to the network functional composition approach for a flexible Internet architecture [13], which decomposes the layered network stack into loosely coupled building blocks to enable customized functionality composition in respect to application specific requirements.

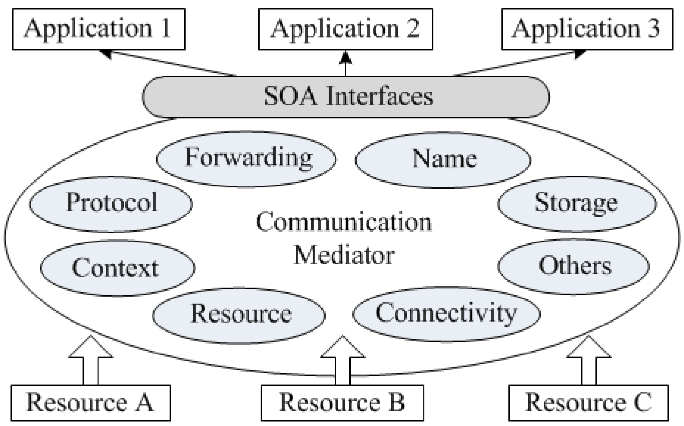

Figure F.7: SOA-based Mediation Implementation

However, real life deployment of the mediator is not trivial. Exiting mediation solutions shall be examined. Haggle [14] and the DTN architecture [4] are two well-known examples, offering applications mechanisms to tune their behaviour to challenging connectivity situations and to change their way of operation according to the observed communication characteristics. Haggle is a layer-less network architecture design targeted for mobile devices. Applications delegate the task of handling and communicating data to Haggle, which in turn adapts to the current network environment using the best available connectivity and protocol for the situation and user-specified policies that allow trading speed, cost and power constraints. DTN architecture is designed as an overlay network to support interoperability among regional networks and for handling intermittent connectivity, long or variable delay, asymmetric data rates, and high error rates. DTN connects regional networks, e.g., to the Internet, by translating between incompatible network characteristics and acting as a buffer for mismatched network delays.

Haggle is a clean-slate design which is considered relatively impractical because of the difficulty of deployment on the large worldwide base of IP networks. Although DTN can act as an overlay on top of both IP and non-IP networks, the Bundle implementation of the DTN architecture still has many problems [11]. Furthermore, not all the mediation features listed before can be fulfilled by the current Bundle protocol or Haggle. Therefore, in order to implement SOA-based mediation in the maritime domain, we propose a three-step strategy explained in the following sections. Step I encapsulates existing mediation APIs as Web services and provides extra functionalities covering, e.g., security, resource and context management. Step II deploys this mediator on specific mobile devices and devices on ships first, applying independent and incremental deployment. Step III is to evolve with the trend of service-oriented network architecture.

Encapsulating existing mediation APIs

While neither Haggle nor DTN architecture are ideal mediation candidates for all communication environments, each has its unique capabilities which can be utilized in the maritime scenario. However, as both have their own implementations and APIs which are quite possible to change in the future, to develop applications using these APIs directly is not a secure approach in terms of future proofing or reliability, and causes interoperability problems. Besides, it is possible that another efficient peer-to-peer or ad-hoc protocol can replace Haggle or a better algorithm implementation can replace Bundle. Such scenarios dictate the need for an abstraction which can hide the architectural details from applications, a common standardized platform which can provide sufficient flexibility to incorporate diversity under widely accepted guidelines and standards. Web Services provide one such platform for abstraction, allowing portability on top and the innovation underneath.

Therefore, our first step is to take advantage of existing mediation mechanisms, e.g., DTN, but encapsulate the functionalities as Web services. This abstraction is in line with the generic communication API (netAPI) proposed in [15]. In addition, we need to provide functionalities that are not covered by current DTN solutions, such as security, trust mechanisms, context-awareness services, and routing management.

Research efforts which simplify the development of DTN applications exist, like the TierStore filesystem [16] and the DTN Service Adaptation Middleware (DSAM) [12]. TierStore is a solution that combines DTN aspects with the standard filesystem, aimed at applications portability onto DTN by offering a familiar and easy-to-use API. DSAM facilitates delay-tolerant application development from reusable components, and away from the details of implementing the DTN Bundle protocol for different platforms and operating systems.

Although TierStore can work immediately for an application that only makes simple use of the file system, wrapping applications as standard services is still needed in order to be integrated with other services. While the DSAM approach allows application modules to be distributed over heterogeneous platforms and reduces the complexity of developing DTN applications over multiple operating systems and network protocols, middleware standardization is further preferred.

With our approach of encapsulating DTN APIs as standardized Web services, some of the benefits achieved would be: easier development of delay-tolerant applications, ensured interoperability with other standard services, enabling evolution underneath and speeding up the process towards integrated operations for ships.

Deploying the Mediator

Regarding a SOA-based communication mediator, we can imagine it with providing a connectivity service that dynamically trades between service requirements and the available communication resources, e.g., network connectivity with longer delay in harsh conditions. In practice, this abstract connectivity service involves diverse functionalities covering different network features: naming/addressing, forwarding, message fragmentation, data management (storage), transportation selection (protocols, network interfaces) and so on.

Within a strictly layered architecture, these functionalities may be buried under interfaces that are not visible to applications, e.g., implemented via network- and link- layer protocols. With an overlay approach, most functionalities can be realized on the overlay that are used directly by applications. Further via the standard Web service exposure, these functionalities can be invoked by applications via a local environment or over a network transparently, whereas gateway translation is needed if services using different transport mechanisms are involved.

Using the overlay approach, we can implement this SOA-based DTN mediator on specific customer devices first: crew mobile phones (with consideration to many existing implementations of DTN capability on smartphone platforms, such as Bytewalla [17]) and servers on ships (similarly for other vehicles), to facilitate standard DTN applications development. Underneath, the DTN capability of these overlay nodes ensures efficient communication in challenged areas, along with other gateway nodes placed on board, in satellite core networks (probably), in ports and home offices.

Evolving with service-oriented network architecture

While todays Internet has evolved to an applications (services) oriented critical infrastructure, it still uses its early principles, such as best-effort packet switching, end-to-end connectivity and the layered architecture. It leads to an inflexible system that has limited abilities to keep up with the constantly changing demand of applications and hinders new functionalities to be deployed in the network. Therefore, a lot of new architecture proposals appear and functional composition is one of them. Functional composition design originates in the component-based architecture approach, where the mediation functionality can exist as a modular application component as other functional building blocks in a loosely-coupled way. If these components are provided with standardized interfaces (e.g., Web services) following the SOA methodology, networking resources and mediation mechanisms will become reusable services interoperable with standard computing services. Flexibility can be maximized, and changes can be easily embraced both from applications and from network technologies.

We anticipate that such a service-oriented network architecture will emerge and spread in future [18, 19, 20, 21]. However, evolution towards this vision needs careful implementation, e.g., in the maritime scenario, we suggested a SOA-based DTN mediator following incremental and independent deployment, by deploying services as overlays first. This mediator can be seamlessly integrated into future service-oriented network architectures. Connectivity will be provisioned as standard services to applications with many adaptive parameters and delay will become one of them. These parameters can be adjusted in real time between applications as the service consumer and communication mechanisms as the service provider.

V. Conclusions

In this paper, we explained two main challenges to be handled by integrated operations for ships, interoperability among heterogeneous applications and connectivity through difficult maritime networks. The common approach was to address them independently by investigating application integration and communication integration solutions. However, as inconsistencies were identified after we put separate solutions together within the maritime context, negotiation in between became necessary: 1) applications need to adapt to diverse and changing communication conditions, and follow a service-oriented integration trend, 2) network solutions must mediate the adaptiveness by interacting with application services for managing communication resources, with consideration of the service-oriented integration paradigm. We believe that this negotiation is significant for challenging communication scenarios like the maritime case. Our way of approaching it based on SOA and via incremental deployment can push the negotiation mediation to becoming a normal capability of future network architectures.

REFERENCES

[1] L. Mu, L. Kuang, and A. Prinz, “Maritime data integration using standard ISO 15926, in *Proceedings of the 20th International Offshore and Polar Engineering Conference,* 2010.

[2] T. Erl, Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services. Prentice Hall PTR, 2004.

[3] L. Mu, R. Kumar, and A. Prinz, “An integrated wireless communication architecture for maritime sector. in *Proceedings of the 4th International Workshop on Multiple Access Communications,* 2011, pp. 193-205.

[4] K. Fall, “A delay-tolerant network architecture for challenged internets, in *Proceedings of conference on Applications, technologies, architectures, and protocols for computer communications,* 2003, pp. 27-34.

[5] L. Mu and A. Prinz, “Delay-oriented data traffic migration in maritime mobile communication environments, in *Proceedings of Fourth International Conference on Ubiquitous and Future Networks,* 2012, pp. 417-422.

[6] L. Mu, “A hybrid network for maritime on-board communications, in *Proceedings of the 8th International Conference on Wireless and Mobile Computing, Networking and Communications,* 2012, pp. 761-768.

[7] I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke, “Grid services for distributed system integration, *Computer,* vol. 35, no. 6, pp. 37-46, 2002.

[8] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, “Resilient overlay networks, in *Proceedings of the eighteenth ACM symposium on Operating systems principles,* 2001, pp. 131-145.

[9] Z. Duan, Z.-L. Zhang, and Y. Hou, “Service overlay networks: SLAs, QoS, and bandwidth provisioning, *IEEE/ACM Transactions on Networking,* vol. 11, no. 6, pp. 870-883, 2003.

[10] J. Ott, “Delay tolerance and the future internet, in *Proceedings of the 11th International Symposium on Wireless Personal Multimedia Communications (WPMC),* 2008.

[11] L. Wood, W. Eddy, and P. Holliday, “A bundle of problems, in *Proceedings of IEEE Aerospace conference,* 2009, pp. 1-17.

[12] H. Zhuang, H. Ntareme, and B. Pehrson, “A Service Adaptation Middleware for Delay Tolerant Network based on HTTP Simple Queue Service, in *Proceedings of the 6th USENIX/ACM Workshop on Networked Systems for Developing Regions,* 2012.

[13] C. Henke, A. Siddiqui, and R. Khondoker, “Network functional com- position: State of the art, in *Proceedings of the Telecommunication Networks and Applications Conference,* 2010, pp. 43-48.

[14] J. Su, J. Scott, P. Hui, J. Crowcroft, E. D. Lara, C. Diot, A. Goel, M. H. Lim, and E. Upton, “Haggle: Seamless networking for mobile applications, in *Proceedings of the 9th international conference on Ubiquitous Computing,* 2007, pp. 391-408.

[15] G. Ananthanarayanan, K. Heimerl, M. Zaharia, M. Demmer, T. Kopo- nen, A. Tavakoli, S. Shenker, and I. Stoica, “Enabling innovation below the communication API, *Tech. Rep. UCB/EECS-2009-141,* Oct 2009.

[16] M. Demmer, B. Du, and E. Brewer, “Tierstore: a distributed filesystem for challenged networks in developing regions, in *Proceedings of the 6th USENIX Conference on File and Storage Technologies,* 2008, pp. 1-14.

[17] H. Ntareme, M. Zennaro, and B. Pehrson, “Delay tolerant network on smartphones: Applications for communication challenged areas, in *Proceedings of the 3rd Extreme Conference on Communication,* 2011.

[18] P. Müller and B. Reuther, “Future internet architecture - a service oriented approach, *it - Information Technology,* vol. 50, no. 6, pp. 383-389, 2008.

[19] A. Karouia, R. Langar, T.-M.-T. Nguyen, and G. Pujolle, “SOA-Based Approach for the Design of the Future Internet, in *Proceedings of the 8th Annual*

Communication Networks and Services Research Conference, 2010, pp. 361-368.

[20] R. Kumar, "Fifth generation networking principles for a service driven future Internet architecture, *Wirel. Pers. Commun.,* vol. 57, pp. 393-411, 2011.

[21] A. J. Gonzalez, R. M. de Pozuelo, M. German, JesusAlcober, and F. Pinyol, "New framework and mechanisms of context-aware service composition in the future internet, *ETRI Journal,* vol. 35, no. 1, pp. 7-17, 2013.