Piërre van de Laar · Jan Tretmans
Michael Borth *Editors*

Situation Awareness with Systems of Systems

Springer

Situation Awareness with Systems of Systems

Piërre van de Laar • Jan Tretmans • Michael Borth
Editors

Situation Awareness with Systems of Systems

Springer

Editors
Piërre van de Laar
Embedded Systems Institute
Eindhoven, Netherlands

Michael Borth
Embedded Systems Institute
Eindhoven, Netherlands

Jan Tretmans
Embedded Systems Institute
Eindhoven, Netherlands

ISBN 978-1-4614-6229-3 ISBN 978-1-4614-6230-9 (eBook)
DOI 10.1007/978-1-4614-6230-9
Springer New York Heidelberg Dordrecht London

Library of Congress Control Number: 2012956306

Foreword

Nothing less than a disruption in thinking about naval systems was taking place at Thales when we, together with ESI, conceived the POSEIDON project. Thales has a long track record in defense systems – using many proprietary solutions designed to perform reliably under extreme operational conditions of armed conflict anywhere in the world. For the future, we wanted to enter the market of maritime safety and security (MSS). Systems supporting MSS missions impose very different requirements and in fact open up the possibility to utilize open source and state-of-the-art technologies from the civil domain.

Thales had a challenge in using commercial off-the-shelf technologies as they were traditionally not qualified to meet the demanding requirements of our core business of building highly reliable defense systems: How to conceive and develop a different type of system targeted at a new market opportunity? This had to be achieved with our existing pool of highly talented technical professionals with mission critical defense systems in their blood.

As with all high-tech organizations, it all begins with a handful of key people with a vision who are capable of convincing decision makers to allocate budgets to new projects that will result in attractive and smart solutions. This was all set in place with a number of projects running to achieve our MSS ambition. Expectations, however, were very high in the sense that it was assumed that we could reach the level of deliverable products very quickly; after all it was "R&D business as usual." In our pragmatism, we missed the point somewhere in this palette of projects.

We were in need of a more out-of-the-box thinking that would come up with new concepts to pull the population at Thales across the line. This was where ESI, together with its partners, came in and POSEIDON was born. The Industry-as-Laboratory approach used by ESI was key in our decision to proceed. It matched our belief that product goals and research goals can go hand in hand. Given the right environment of quality staff, who understand, respect, and support each other's goals and are backed up by facilities and processes, they catalyze each other and can achieve great results. This is exactly why POSEIDON has been so successful. It gave birth to numerous product concepts that found their way into our naval systems portfolio and also resulted in many publications and dissertations.

Now after 5 years, you will find everything you may want to know about the results of POSEIDON in this book. I would like to add, with reference to a famous Gestalt law, known from psychology, that "the whole is more than the sum of its parts." Above all, POSEIDON has been a highly inspiring journey that substantially contributed to the mindset change now helping Thales to develop advanced naval systems for the future.

Delft, September 2012

Jimmy Troost
Director TRT-Delft
Thales Netherlands

Preface

It is with great pleasure that I welcome you to the final book on the Embedded Systems Institute project POSEIDON. The project was funded under the Dutch BSIK program "Embedded Systems." The project partners were the Embedded Systems Institute (ESI), Thales Netherlands, Noldus Information Technology, Delft University of Technology, Eindhoven University of Technology, University of Amsterdam, Tilburg University, and VU University Amsterdam. The project started in June 2007, ended in May 2012, and encompassed an overall volume of 84 fte.

As for all of ESI's large projects, POSEIDON has followed the by now well-known Industry-as-Laboratory paradigm, in which scientific research is performed in the context of an industrial case. For POSEIDON, the case was defined in the context of the new emerging market of support systems for maritime safety and security. The POSEIDON partners addressed a variety of research topics ranging from integration and testing to systems-of-systems, from visualization to security, from vessel trajectory segmentation to adapter generation, and from situation awareness to trustworthy information interoperability.

The POSEIDON project has been highly successful. Among the results we count the following highlights:

- An architectural framework for information-centric systems of systems and an integrated demonstrator, showing how the combination of many new technologies can be applied to offer improved system support to coast guard operators for a higher level of situation awareness.
- An extendable method to analyze and visualize the kinematic behavior of moving objects. This method offers powerful solutions for the construction of user-defined operational pictures in next-generation maritime systems.
- A highly efficient data reduction method resulting in vessel trajectories using only 2 % of the original amount of data.
- A formal definition of a semantic concept hierarchy of maritime information, enabling automatic reasoning on semantic level with maritime concepts, implemented in a knowledge base.

- A new method for trust management and distributed access control for use in a systems-of-systems environment in the absence of a central security authority.
- Concepts and techniques for systems integration and acceptance at runtime: systems join-and-leave, runtime acceptance testing, and system health diagnosis.
- Adaptor generation techniques for the quick realization of reliable connections between systems.
- A method for runtime anomaly detection by mining of semantic information about ship movements.
- Strong cooperation between universities resulting in a number of shared publications.
- Over 100 scientific and professional publications and PhD. and MSc. theses.

All partners in the project are satisfied with the results achieved in the POSEIDON project. Some of the results and insights obtained in POSEIDON will find their way in the Thales Netherlands product portfolio. Other achievements have found their way to the portfolio of projects that ESI is executing together with industrial and academic partners, including the successor project METIS, where new research topics are tackled that were instigated by POSEIDON.

I would like to thank all project participants for their commitment and contributions: as a team they have turned POSEIDON into a success! The support of Thales Netherlands and the Dutch Ministry of Economic Affairs (now EL&I) through AgentschapNL is gratefully acknowledged. We also thank Springer for their willingness to publish this book. With this book, we expect to share the important results achieved with a larger, worldwide audience, both in industry and academia.

Eindhoven, September 2012

Prof. dr. ir. Boudewijn Haverkort
Scientific Director and Chair
Embedded Systems Institute

Acknowledgements

Situation Awareness with Systems of Systems is a result of the POSEIDON project. The Industry-as-Laboratory project POSEIDON would not have happened without the technological and managerial vision of the Embedded Systems Institute (ESI) and the provision of a "laboratory" inside their company by Thales Netherlands. POSEIDON was partially supported by the Dutch Ministry of Economic Affairs under the BSIK program.

We gratefully acknowledge the cooperation with the employees of Thales Netherlands throughout the 5 years of the POSEIDON project. We are also grateful to the Dutch Coastguard and Marin for providing us with domain knowledge, data, and valuable insights. We thank the employees of our academic partners (Eindhoven University of Technology, VU University Amsterdam, University of Amsterdam, Tilburg University, and Delft University of Technology), Thales Netherlands, Noldus Information Technology, and ESI for writing and reviewing a variety of book chapters. All efforts together resulted in this book that shows the current state of the art in situation awareness with systems of systems.

Contents

Part I
General

Chapter 1
Introduction: Situation Awareness, Systems of Systems, and Maritime Safety and Security

Jan Tretmans and Piërre van de Laar

1.1 Introduction

Situation awareness, i.e., being aware of the environmental situation by collecting and interpreting information, is a prerequisite for many organizations to make informed decisions and take appropriate actions. In many domains, such as air traffic control, chemical plant surveillance, combating emergency situations, and controlling maritime safety and security, computer-based support is thereby indispensable for gathering and processing all the relevant data. Such a computer system for supporting situation awareness is often implemented as a *system-of-systems*, i.e., as an evolving collection of distributed, heterogeneous, autonomous, cooperating systems, without a clearly identifiable centralized control.

This book presents and discusses various aspects, challenges, and solutions for developing systems-of-systems for situation awareness, with applications in the domain of maritime safety and security. This chapter introduces the book, provides an overview of the chapters it contains in Sect. 1.6, and introduces the core topics. First, the concept of situation awareness is elaborated in Sect. 1.2, which is followed by a discussion on computer support for situation awareness in Sect. 1.3. Section 1.4 discusses the characteristics of systems-of-systems. Situation awareness in the domain of maritime safety and security is further investigated in Sect. 1.5. Since the results presented in this book were obtained in the Dutch research project POSEIDON, Sect. 1.7 concludes this introductory chapter by putting the results in the context of this project.

Since the area of situation awareness with systems-of-systems is dynamic, lively, and broad, it is impossible to be complete and cover all relevant topics in full detail. This book is also not a blueprint for building systems-of-systems for situation

J. Tretmans (✉) • P. van de Laar
Embedded Systems Institute, Eindhoven, The Netherlands
e-mail: jan.tretmans@esi.nl; pierre.van.de.laar@esi.nl

P. van de Laar et al. (eds.), *Situation Awareness with Systems of Systems*,
DOI 10.1007/978-1-4614-6230-9_1,

awareness. Yet, the book does discuss many challenges when building such systems and proposes solutions in many areas, including the construction of a demonstrator in Chap. 4. The maritime domain, more specifically maintaining safety and security in the Dutch part of the North Sea, is chosen as the primary application area, but many, if not all of the issues discussed in this book are easily transferable to other domains of situation awareness and to other kinds of systems-of-systems.

This book intends to give an accessible overview of the results of the POSEIDON project for anybody interested, for academics as well as for technical professionals working in the area of situation awareness and/or systems-of-systems. It is not the intention to give a full scientific treatment of the topics covered; where necessary references to other publications, such as journal and conference papers, are made. A list of all POSEIDON publications is contained in Appendix B. The different chapters are independent, so that sequential reading is not necessary.

1.2 Situation Awareness

Collecting, aggregating, and interpreting information in order to know what is happening in the environment and to be aware of the situation in the surroundings, i.e., *situation awareness*, is a prerequisite for many animals, humans, and organizations to make informed decisions and take appropriate actions. A rabbit observes its environment, it looks around, listens, and smells to identify a potentially dangerous situation, such as a fox approaching, to be able to react in time.

A car driver observes, interprets, and tries to predict the behavior of other cars in the vicinity. He, or she, adapts his own behavior, and combines his observations with previous experience and knowledge about cars in similar situations, while incorporating additional information from traffic signs, traffic information on the radio, and instructions from his navigation device, in order to prevent accidents, avoid traffic jams, and safely reach his destination.

An organization such as a traffic control center must be constantly aware of the situation on the roads in its designated area, in order to optimize traffic throughput, minimize congestion, maintain safety, enforce traffic rules, respond to emergency situations and accidents, deal with road blocks and reconstruction works, and minimize environmental pollution. The traffic center observes and monitors the traffic using different sources of information, e.g., cameras, detection loops, visual observation, intelligent road sensors, information obtained from satellites, and, if possible, messages sent from cars themselves. In addition, external information sources that are only indirectly linked to the current traffic are important for appropriate traffic control, such as current and predicted weather conditions, historical information about traffic streams and rush hour patterns, holiday periods, planned reconstruction work, special or voluminous transports, events that attract a lot of people and cars such as a football match and a rock concert, and information from neighboring traffic control areas.

There are many organizations for which knowledge about what is happening in the environment is of prime importance as a starting point for making decisions and taking actions. Examples are air traffic control, chemical plant surveillance, monitoring large and complex machines, responding to emergency situations and natural disasters, monitoring and controlling safety and security at sea including tsunami warnings, knowing positions, movements, and threats in military operations, and crowd surveillance such as knowing how people move during a soccer match, a demonstration, or a concert.

Situation awareness involves acquiring information about the environment, about who is doing what and where, and then interpreting this information for a particular goal [1]. Apart from directly observing the environment, additional, indirect sources of information can be used to help with the interpretation and understanding of the environment, e.g., historical information, extra information about actors in the environment, or public information in the news or available on the web.

Vast amounts of information can be produced by different sources. Often these sources will agree, but sometimes they may provide inconsistent or contradicting information. Selecting, aggregating, filtering, combining, and interpreting information, reasoning about the acquired information, searching for correlations, assessing the trust and reliability of information, and trying to predict how the environment will evolve, are all part of creating situation awareness.

In many domains a main goal of situation awareness is to detect abnormal or unusual events that can lead to dangerous, threatening, or undesired situations, e.g., a traffic jam, a tsunami threat, a potentially explosive situation in a chemical plant, a hostile missile approaching with high speed, or squashed people in a crowd. Perceiving and alerting to such anomalous situations in the vast amounts of information is then important, while filtering out normal situations as much as possible.

1.3 Systems Supporting Situation Awareness

In most domains, computer-based support is indispensable for attaining good situation awareness. A support system for situation awareness helps with gathering, processing, and interpreting the vast amounts of relevant data. Typically, such a system presents its output to a human operator, e.g., to an operator surveilling traffic in a traffic control center. With the help of such a system the operator will get a better overview of what is happening, and, consequently, can make better decisions and take more effective actions.

Some situation awareness systems are able to perform actions autonomously. Most of such systems are either simple so that appropriate actions are straightforward, or time-critical so that human intervention would cost too much time. Our focus, however, is on systems that only support decision making by presenting a view of the current situation. Taking appropriate actions is then left to the human surveillance operator.

Support systems for situation awareness face several challenges. Whereas at first sight such systems look like straightforward information processing systems, i.e., gathering input data, processing these data, and presenting the information to the (human) user, a more precise analysis shows that there is more involved. We mention a couple of challenges.

First, the data sources provide data in large quantities. This means that filtering, focusing, compression, and selection of relevant data are needed in such a way that no important information is lost. Reduction of data quantity is necessary to make incoming data more easily processable, to make it presentable to a human user, and to enable storage of data to gradually build a set of historical data.

Availability of historical data allows to recognize patterns, to compare current data with what happened in the past, and to use data as training set for learning purposes. But it adds a second challenge of managing these historical data: keeping the amounts of data under control, recognizing and removing obsolete data, alignment of historical data with changing situations, and keeping the integrity and consistency of historical data.

A third challenge concerns the heterogeneity and independence of data sources, ranging from (intelligent) sensors, radar, and satellite links, to databases and the (semantic) web. This implies that there are differences in format, syntax, semantics, and protocols, which must be aligned. Format and syntax transformations must deal with syntactic interoperability, and protocol converters and adapters are needed to bridge the gaps between various protocols. Semantic differences, such as using the same term for different concepts (consider all the different meanings of the sentence "The girl saw the man with the glasses."), or using different terms for the same concept (such as the use of 'car', 'vehicle', 'voiture', or 'auto' to denote the same concept) require semantic and ontology alignment. Information fusion is needed to make it possible to combine different pieces of aligned information into larger chunks. Semantic reasoning shall be applied to add knowledge and understanding: aggregating the pieces of information into meaningful new information and deducing higher level knowledge, such as patterns, clusters, and classifications of situations.

An additional challenge in this reasoning is that trust, reliability, and also privacy and confidentiality of information have to be taken into account. Since the information comes from different sources, which are probably not equally reliable, they may provide mutually inconsistent or contradicting information. This leads to notions of trust and uncertainty in information that a system for situation awareness must deal with. Moreover, due to different privacy and confidentiality rules, it can be that not everybody has access to the same information.

Finally, all information and deduced knowledge must be presented to the human operator in such a way that it is easily accessible, manageable, and tractable. Sophisticated visualization techniques are necessary to present the information, both as an overview picture of the environmental situation, and in detail, e.g., to indicate anomalies and explain why a situation is considered abnormal.

1.4 Systems of Systems

A computer system for supporting situation awareness in complex domains is not a monolithic, coherent system. Such a system needs to perform many tasks, it gathers information from different sources at distinct locations, and it interacts with various stakeholders and other systems. Many of the components that perform these tasks or that serve as sources are actually complex, independent systems themselves, which are not under the full control of the situation awareness system. Such a system in which the constituent components are autonomous, complex systems themselves, is called a *system-of-systems*.

A system-of-systems (SoS; sometimes called collaborative system, or federation of systems) is a large-scale, non-monolithic, distributed, heterogeneous, complex system, built from multiple interacting sub-systems, which are complex, autonomous, independently operating systems themselves. There is no central control, and there is no single owner or responsible for the entire system-of-systems. Yet, by collaborating in a system-of-systems, functionalities can be provided that its constituent systems alone would never be able to provide [9].

Research and development in the area of systems-of-systems started in the late 1990s. It was triggered by the growing connectivity between systems and the recognition that connections and collaborations between systems would enable many new applications and opportunities, but that they would also generate many new challenges that surpass the feasibility of traditional system engineering. These challenges have various dimensions involving technological, political, and organizational aspects. Different institutions, universities, government agencies, as well as commercial companies work on them, and also within the European Union research programmes they form a key area [2].

Examples of systems-of-systems are found in traffic management where various systems, including in-car devices, collaborate to optimize traffic flow in urban areas; smart cities where systems for traffic management, public transportation, energy management, etc. work together to optimize sustainability; smart buildings, where surveillance, access control, fire emergency, heating, climate control, and lighting interact; cross-company integrated business process management; and many systems in the domain of situation awareness.

The main characteristic of a system-of-systems, as opposed to a classical system consisting of components, is the autonomy and operational independence of the constituent systems, and thus the lack of central control. This has a couple of important consequences which challenge the design, validation, testing, deployment, and maintenance of systems-of-systems.

A first consequence is the evolving nature of systems-of-systems and the necessity to perform activities like testing, acceptance, and reconfiguration at runtime, i.e., online. Since each constituent system runs independently from the others, it can autonomously be started, stopped, removed, replaced, updated, or degraded. This means that the configuration of the system-of-systems changes and evolves dynamically without central control. Other systems and the entire

system-of-systems must be able to cope with such reconfigurations and must adapt to them, e.g., searching for a substitute system that delivers a service that can replace the service of a leaving system. But also the other way around, the system-of-systems must adapt itself and can remove or disconnect an individual system, e.g., if the quality of its service degrades too much. The necessity to perform dynamic, runtime reconfigurations is strengthened by the requirement of typical application areas of systems-of-systems, e.g., systems for situation awareness, that the systems are always up and running.

A derived consequence is that in systems-of-systems that perform runtime reconfigurations, several validation and quality checks must also be performed dynamically. Examples are runtime monitoring of systems and their quality of service, runtime testing to decide whether a new system can join the system-of-systems, and runtime fault diagnosis to pinpoint the malfunctioning system in case a failure occurs. Runtime testing, however, may lead to additional complexity by causing side-effects through undesired interactions between the operational system-of-systems and the tested system, e.g., during a test of the fire alarm system it is not always desirable that also the entire automatic sprinkler system is activated. Runtime verification and validation activities must continuously check and maintain the quality and reliability of the entire system-of-systems, in particular also during the reconfigurations.

A second consequence of autonomy is that the constituent systems in a system-of-systems were designed independently, i.e., they were not specifically designed to work together. Consequently, their combined behavior may lead to emergent behavior, i.e., behavior that is not fully predictable from knowing the behaviors of the constituent systems, thus leading to uncertainty about the overall behavior.

In addition, if systems have to interact with other systems that are not known in advance, interfaces must be flexible enough to adapt to such interactions, or special connectors or adapters have to be made that can bridge both the syntax and semantic differences. It is a challenge to design systems that are flexible enough to operate, adapt, and connect to other systems in the dynamic, evolving context of systems-of-systems.

A third point that follows from the lack of central control together with dynamic reconfigurations and tests, is the difficulty to precisely know the global state of the entire system-of-systems. This involves the configuration, i.e., knowing which constituent systems are available at a given moment, the quality of the constituent systems, i.e., knowing which systems are healthy and operating correctly, as well as the quality, reliability, and validity of the information being processed by the constituent systems: an unhealthy system may produce unreliable information.

The entire system-of-systems, as well as the constituent systems, must be able to cope with the uncertainty caused by the lack of knowledge about the global state. The consequence is that a system-of-systems, in addition to dealing with its primary information, must also communicate and reason with meta-information about its own configuration, its own health and the health of its constituent systems, and the quality and reliability of the primary information. This means that a system-of-systems must reflect on its own operation. One might say that this constitutes a

'meta-situation awareness system' for system awareness: like a situation awareness system for road traffic monitors and controls traffic streams and warns for anomalies on the roads, the meta-situation awareness system monitors and controls information streams and anomalies in the system-of-systems.

A final issue in systems-of-systems concerns security, privacy, and confidentiality. The autonomy of the constituent systems implies that they will all have their own policies with respect to sharing and protecting sensitive information. There is no central authority arranging all security issues. This requires special policies and methods to communicate allowances to specify who is allowed to have which information at what occasions. Special care must be taken that also during reconfigurations and runtime testing no confidential information leaks away.

Compared with traditional systems and system development, systems-of-systems have to deal with blurring boundaries, both in space and in time. In space, a starting point for traditional system engineering is the distinction between a system and its environment. The above discussion shows that for a system-of-systems the boundary between what belongs to the system-of-systems and what belongs to its environment, is not sharp. In time, the boundaries between the traditional development phases, such as design, building, validation, testing, deployment, operation, maintenance, and decommissioning, diminish. After some time of continuous operation, while systems are leaving and joining, a completely new system-of-systems may have emerged, consisting of completely new constituent systems, but still performing the same tasks.

1.5 Situation Awareness for Maritime Safety and Security

Our seas have many functions and are used in many ways, for several purposes, and involving various stakeholders. A challenging and important application area for situation awareness is *Maritime Safety and Security* (MSS). In maritime safety and security, the goal is to keep the seas safe and secure, in particular, the coastal regions that are under control and responsibility of a specific country (Fig. 1.1).

Keeping the seas safe and secure involves many aspects. First of all, the seas serve as one of the most important transportation infrastructures. Many ships use the seas and they shall adhere to sea-traffic rules that must be monitored. Shipping lanes must be marked and maintained, traffic in and out of harbors must be controlled, collisions shall be avoided, and in case of emergency assistance shall be provided. Second, the seas are important for food production. Fishing must be monitored, illegal fishing must be prevented and detected, and fish farms shall be regulated and guarded. Third, the seas often constitute the border between countries, implying that border control, customs, smuggling, illegal immigration, and general defense are issues at sea. The coastal area being a part of the country implies that law enforcement is a fourth issue. This includes, for example, combating piracy, terrorism, and drug trafficking, and protection of assets such as pipelines, historical ship wrecks, and war graves. Fifth, the seas constitute an important ecological system, susceptible

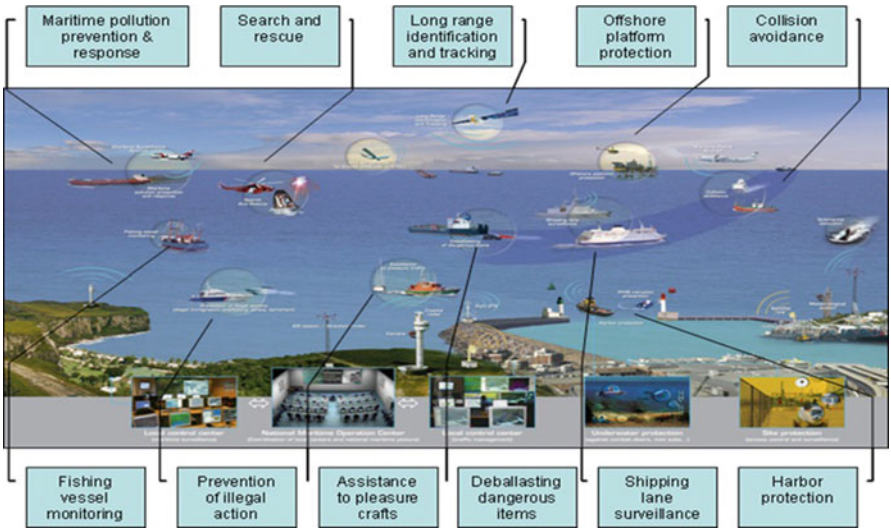

Fig. 1.1 Examples of maritime safety and security (©Thales Nederland B.V.)

to contamination, so pollution monitoring, prevention, and control are important tasks at sea. A sixth aspect of the usage of the seas is the provision of energy through off-shore oil and gas platforms, wind parks, pipelines, and, perhaps in the near future, algae farms. Also these shall be regulated, protected, monitored, and accidents inhibited. A final aspect are safety threats through sea mines, ship wrecks, and lost cargo, and emergency situations ranging from search and rescue to neutralization of oil spills that must be dealt with.

The various activities taking place at sea entail risks, threats, and potential dangers that may jeopardize safety and security. Maintaining maritime safety and security is typically a task that is coordinated by the coast guard, working together with harbor authorities, (water) police, customs, navy, rescue-teams, etc. The first step is attaining situation awareness, i.e., knowing what is happening at sea. Considering the large numbers of ships, the many activities at sea, and the large areas that have to be covered, computer-based support herewith is indispensable. A system for maritime situation awareness will support and guide the surveillance officers and operators in a coast guard control center with monitoring and controlling what is happening at sea, so that better decisions can be made and appropriate actions can be taken.

An important task for maritime safety and security is anomaly detection. A situation awareness system must support the surveillance operators in identifying and focusing on suspicious or abnormal situations, while filtering out normal situations as much as possible. Abnormal situations may indicate undesired or unlawful activities, risks, or threats. Examples include ships violating traffic rules or sailing outside shipping lanes, ships being too close to oil platforms, wind parks, or the coast, and ships being (too) close to each other. The latter may indicate a

near-collision, it may point to handing-over of illegal goods at sea such as drugs, but if one of the vessels is a tugboat or a pilot boat then it probably is completely normal behavior. In general, any vessel that is somewhere where it does not belong, or that makes strange maneuvers, is an anomaly. Of course, this depends on the type of ship, e.g., a fishing ship sailing on fishing grounds is not abnormal, but a passenger ship is. Other examples of abnormal behavior are a ship that drifts too much or seems to be out of control; a vessel providing inconsistent information such as claiming to be an oil tanker while making a 180°-turn within a minute; a small ship approaching the coast with high speed somewhere where there is no harbor which might be an indication that smuggled goods are dropped on the shore; and two ships decreasing speed at the same place outside of traffic lanes just a short period after each other, which might indicate that one ship dropped a packet that is picked up by the second.

Challenges for anomaly detection in the maritime domain are long time frames and external influences. For some abnormal behaviors observations must cover a long time frame, e.g., a potential collision or strange behavior of an oil tanker stretches over several hours, and also the "drop a packet–pick a packet" scenario described above may require observations over several hours before it is clear that an anomaly occurred. External influences, like weather conditions, play an important role when determining whether some situation is abnormal or not, e.g., strange maneuvers of a ship outside the shipping lane may be completely normal during stormy weather conditions.

1.5.1 Vessel Tracking

A core functionality for anomaly detection in the maritime domain is the monitoring and tracking of vessels: where are vessels and what are they doing. Nowadays a main source of information for vessel tracking is AIS – the Automatic Identification System. All vessels, except the smallest ones, are required to be equipped with an AIS transceiver, which broadcasts messages with status data about the vessel and its movements according to a protocol standardized by the International Telecommunications Union (ITU) [13]. Depending on what the ship is doing, it sends AIS messages every 2–10 seconds when underway, and every 3 minutes when at anchor.

There are different kinds of AIS messages. One type contains kinematic information about the ship such as its current position, speed, heading, turn rate, and navigational status ('under way using engine', 'at anchor', 'moored', etc.). Another type of message contains information about the ship itself and its journey such as its name, internationally recognized identifiers (MMSI number – Maritime Mobile Service Identity number[1] and IMO number – International Maritime Organization

[1]The MMSI numbers used in the examples in this book are fictitious and do not relate to existing ships.

number), call sign, dimensions, draught, the type of ship (oil tanker, passenger ship, cargo, fishing, dredger, pilot boat, etc.), cargo, destination, and expected time of arrival.

AIS messages can be received by neighboring vessels to prevent collisions, and by coast guards and similar organizations for vessel tracking and monitoring. AIS is the main source of information for a maritime situation awareness system. Many web sites provide AIS information.[2]

Other sources of information in the maritime domain are radar, satellite, and visual observations, data from harbors and customs, and the web where many web sites, both paid and free, are found that provide useful information for the maritime domain. Examples are weather and news sites, geographic information,[3] Lloyds,[4] and The Paris Memorandum of Understanding on Port State Control (Paris MOU[5]), which contains a lot of information about ships.

1.6 Overview of the Chapters

Many of the challenges described in the previous sections present themselves in maritime situation awareness systems. This book, in its subsequent chapters, elaborates these challenges and discusses solutions. This section introduces these chapters. Although examples in the chapters are taken from the maritime domain, most of the presented topics are more widely applicable to any system-of-systems or domain of situation awareness.

The chapters are divided into three parts: the chapters in Part II discuss situation awareness, whereas Part III focuses on the systems-of-systems aspects. Before starting with these two parts, Part I investigates some general topics. Chapter 2: *Improving Situation Awareness in the Maritime Domain*, further investigates the application domain of maritime situation awareness systems. The role and activities of an organization for maritime safety and security, the kind of support, alerts, and visualizations that may help the operators, the complexity of their decisions, and the various stakeholders with their conflicting interests are all discussed.

Building a system to support attaining situation awareness, whether in the maritime domain or elsewhere, requires the consideration of various, partly conflicting, functional and non-functional concerns. Chapter 3: *On the Architecture of Systems for Situation Awareness*, discusses such issues by reflecting on the architecture of systems-of-systems for situation awareness. The chapter discusses

[2]Examples of web sites providing AIS information, are www.vesseltracker.com, www.shipais.com, www.vesselfinder.com, www.marinetraffic.com, www.aishub.net, and www.shipspotting.com.

[3]www.geonames.org

[4]www.lloydslistintelligence.com

[5]www.parismou.org

the architecture starting from general principles, considering both a functional view on the information processing that results from domain analysis, and a system architect's view on properties required to deliver that functionality.

Many of the ideas presented in this book have been implemented, and these implementations have been integrated into a demonstrator that presents an elaborate scenario in the maritime safety and security domain. Chapter 4: *The* POSEIDON *Demonstrator*, describes the demonstrator, including screen shots of its operation, reflections on the building process, and the value of such a prototype for stakeholder communication and validation of the research.

1.6.1 Overview of Part II: Situation Awareness

A system supporting situation awareness must present its output in a useful and manageable way to a human user. Using textual output is not an option given the large amounts of vessel information. Powerful visualization techniques are necessary to enable the operator to quickly understand and interpret the current situation. Chapter 5: *Visualization of Vessel Traffic*, presents advanced visualization methods based on density maps. They enable the operator to attain an overview of movement patterns over a period of time, as well as to zoom in on particular situations. Current behavior of vessels can be visually combined with historical vessel movements, presented in a density map, to detect abnormal behavior, i.e., outliers. Filtering on vessel attributes allows to focus on particular ships or situations, e.g., on all passenger ships sailing on some place where there are normally (in the historical data) no passenger ships.

As discussed in Sect. 1.5.1, an important function in situation awareness in the maritime domain is the tracking of vessels and the analysis of their movements. There are many other domains where tracking of objects and the analysis of behavior are important tasks, e.g., traffic monitoring and control, transportation of system parts in a warehouse, crowd monitoring for public safety and security, migrating animals such as whales, reindeer, and birds, and animals moving in a confined area such as a cage or aquarium. Chapter 6: *Extending Track Analysis from Animals in the Lab to Moving Objects Anywhere*, compares vessel tracking with animal tracking. A specialized tool for video tracking of animal behavior in a cage in a laboratory setting is analyzed and adapted for use in the maritime situation awareness domain. The goal of Chap. 6 is to increase insight in the specificities of both domains of object tracking, and to gather requirements for a universal tracking tool which is applicable to multiple domains.

The main source of data for a maritime safety and security system is AIS data (Automatic Identification System); see Sect. 1.5.1. There are many ships, each of them sending AIS messages every few seconds, even if the ship follows a straight, predictable course. Chapter 7: *Recognizing Vessel Movements from Historical Data*, presents in three steps how AIS data is prepared and used for the recognition of vessel movements and behavior analysis. First, AIS data is compressed into track

segments using a technique called piecewise linear segmentation. This enables higher level reasoning about ship trajectories, and reduces the quantity of AIS data with more than 95 % without sacrificing the quality of subsequent behavioral analyses. Second, the level of similarity between different trajectories is quantified using a distance function. Knowing which trajectories are similar to each other enables clustering of similar trajectories and the recognition of movement patterns. Third, the movement patterns are combined with other knowledge about ships and their context, such as the ship type or the geographical location of its position, because such knowledge may influence what is normal: a movement pattern that is normal for a ship of type 'fishing boat' may be abnormal for an oil tanker, and a speed which is normal for open waters can be too fast when we know that the ship's position corresponds to the entrance of a harbor.

The idea of detecting abnormal vessel behavior by calculating similarities and differences between behaviors using distance functions is further elaborated in Chap. 8: *Density-Based Anomaly Detection in the Maritime Domain*. Vessel behavior is then classified as an outlier if it has a large distance to other behaviors. Chapter 8 introduces a method coined Stochastic Outlier Selection, that automatically identifies outliers.

Chapters 7 and 8 use statistical comparison of current behavior with historical behaviors using distance functions to define what normal behavior is. This leads to an implicit (and circular) definition: normal is that what everyone does. An alternative approach is to explicitly define normal behavior via a set of rules. A violation of the rules is then an anomaly. A typical example is fixing a set of traffic rules that all ships, or cars, must satisfy.

Chapter 9: *Analyzing Vessel Behavior using Process Mining*, uses rule-based anomaly detection. A graph-based language is introduced, founded on linear temporal logic, in which rules specifying ship behavior can be expressed, for example "whenever a ship is moored, then eventually in the future it will be under way using engine". Satisfaction of such rules is checked at runtime. Behavior rules can be explicitly expressed, but they can also be learned from historical data, thus providing a combination of rule-based and history-based anomaly detection. Chapter 9 also presents a template-based method for learning behavior rules.

Situation awareness involves knowing about the events happening in the environment. Consequently, the concept of an 'event' is important, and a system supporting situation awareness must be able to handle events. Events are observed, modeled, and defined, they must be stored, manipulated, and related to each other, and they must allow various kinds of (formal) reasoning. Therefore, Chap. 10 *The Simple Event Model*, introduces an ontology-based event model, that is used to model all kinds of events and their related concepts like actors, places, times, and their types. Events can be modeled at various levels of abstraction: a ship moving at a particular place and time, corresponding to one AIS message, is an event, but also a ship trajectory, the hijacking of a tanker in the Strait of Malacca, and an armada sailing from Wellington to Amsterdam can be modeled as an event. In the context of situation awareness, where information comes from different sources, events must be flexible, they must deal with partial, duplicate, contradicting, or uncertain

information, and they must enable enrichment with new information and additional aspects, that are obtained via observations, from the web, or through semantic reasoning.

1.6.2 Overview of Part III: Systems-of-Systems

One of the challenges in the development of systems-of-systems is the fast, runtime integration of autonomous components or systems, which were originally not designed to interact with each other. In such cases a dedicated adapter may be developed that bridges the differences and incompatibilities between the systems. Chapter 11: *Specification and Generation of Adapters for System Integration*, discusses two methods to generate such an adapter semi-automatically from a model of the interface behavior of the systems. The first method uses techniques from controller synthesis; the second one builds on incremental view maintenance in databases. The methods provide a generic and systematic approach for the construction of adapters. This is illustrated with an example in the maritime domain: the connection of a system providing AIS messages to Google Earth, in order to display these messages.

Chapter 12: *The POLIPO Security Framework*, discusses security issues in systems-of-systems. Since systems-of-systems are dynamic coalitions of autonomous systems, a central security policy cannot be implemented. Each of the constituent systems will have its own policy. To cope with this situation, Chap. 12 introduces the POLIPO security framework that protects the information exchanged among the systems in a system-of-systems, while preserving autonomy and interoperability of the systems. It uses context-aware access control and trust management to protect information from unauthorized access, while ontology-based services maintain autonomy and interoperability.

Apart from security, i.e., the question who is allowed to know what, the distributed, autonomous, and heterogeneous nature of systems-of-systems also raises the question of trust, i.e., which information can be counted on. In particular, if different sources of information provide inconsistent or contradicting opinions about the same ship or about the same event, it is important to know which information can be trusted. Chapter 13: *Assessing Trust for Determining the Reliability of Information*, discusses how trust can be assessed and quantified, and how combinations of different opinions increase the level of trust if they agree, and decrease it if they contradict each other.

Systems-of-systems change and evolve dynamically in ways not designed and anticipated in advance. Consequently, during and after each change the quality of the newly integrated system has to be checked again without disturbing or interfering too much with the normal operations of the system-of-systems. This requires runtime monitoring and testing techniques. Moreover, if a failure occurs, runtime diagnosis techniques must be able to localize and isolate the faulty system. Chapter 14: *Online Fault Localization and Health Monitoring for Software Systems*,

discusses the detection of failures and the localization of the faults that led to the failures by adapting existing design-time techniques to the dynamic context. In particular, the technique of spectrum-based fault localization is combined with health monitoring and extended to runtime fault localization.

Since runtime fault localization shall as little as possible disturb the normal operations of the system-of-systems, it shall be effective and fast. Chapter 15: *Prioritizing Tests for Fault Localization*, shows that current test selection and prioritization techniques mainly optimize the fast detection of failures, but not their fast localization. Therefore, Chap. 15 presents techniques for selecting and prioritizing test cases such that fault localization is optimized, i.e., the time to localize the fault is minimized.

1.7 POSEIDON

POSEIDON[6] was a collaborative, industrial-academic Dutch research project, managed by the Embedded Systems Institute (ESI). The goal of the project was to develop new concepts, methodologies, and prototype components for situation awareness systems-of-systems, and to apply them in the domain of maritime safety and security. In POSEIDON, researchers and engineers from the companies Thales Netherlands and Noldus Information Technology, and from ESI, worked closely together with researchers from five universities: Eindhoven University of Technology, Delft University of Technology, VU University Amsterdam, University of Amsterdam, and Tilburg University (Figs. 1.2 and 1.3).

Fig. 1.2 The POSEIDON team at the final POSEIDON symposium

[6] www.esi.nl/poseidon