

Ship's environmental performance monitoring in real time using big data techniques

Leventakis G¹, Nikitakos N¹,

¹Dept. of Shipping Trade and Transport

University of the Aegean

Korai 2a - Chios - GREECE

*N.Nikitakos:

e-mail: nnik@aegean.gr

Abstract

Recently European Union has been proposed a system for monitoring, reporting and verifying (MRV) of CO₂ emissions from large ships using EU ports. Ships would thereby be obliged to monitor four parameters on a voyage basis, namely fuel consumption, distance travelled, time spent at sea, and cargo carried. The monitored parameters would need to be verified and different indicators based on these parameters would have to be reported on an annual basis. In our paper a real time systems is proposed based on complex event processing for early detection of any risks created from the monitoring of MRV's. Stream reasoning is an approach that can be used if information (in the form of assertions) arrives as a stream of (time stamped) inputs. The approach has two features that could be helpful: the knowledge base can be continuously updated and reasoning goals continuously re-evaluated as new assertions arrive, the reasoner considers events from a finite time window, and not only at a single instant. A conceptual framework, inspired from airport electronic data real time risk assessment for resilience, for MRV anomaly detection is presented helping ships to early identify and correct arising risks.

Keywords: maritime environment, MRV, big data, stream reasoning.

1. Introduction

Among the main issue in the maritime sector has always been the transmission of the information from the vessels due to low speed and at the same time extremely expensive transmission cost based on the existing technology. The last decade revolution in maritime communications is a fact. The innovation, as a main reason, in maritime communications is huge and we have passed from Morse code to high bandwidth terminals that can transmit big amount of data in short time and low cost. This technological breakthrough has inspired all maritime related manufactures to upgrade their components by adding several sensors in order to produce a very important amount of information for the operation of the vessel. So vessels today can produce a big amount of data that are very important for the proper operation of the vessel and better management which is guiding to cost reducing

decisions. Big Data, in recent years, has emerged to describe a new paradigm for data applications. New technologies tend to emerge with a lot of hype, but it can take some time to tell what is new and different. While Big Data has been defined in a myriad of ways, the heart of the Big Data paradigm is that is too big (volume), arrives too fast (velocity), changes too fast (variability), contains too much noise (veracity), or is too diverse (variety) to be processed within a local computing structure using traditional approaches and techniques. The technologies being introduced to support this paradigm have a wide variety of interfaces making it difficult to construct tools and applications that integrate data from multiple Big Data sources. To capture the value from Big Data, we need to develop new techniques and technologies for analyzing it. Until now, scientists have developed a wide variety of techniques and technologies to capture, curate, analyze and visualize Big Data. Even so, they are far away from meeting variety of needs. These techniques and technologies cross a number of discipline, including computer science, economics, mathematics, statistics and other expertise. Tools (platforms) are needed to make sense of Big Data. Current tools concentrate on three classes, namely, batch processing tools, stream processing tools, and interactive analysis tools. Most batch processing tools are based on the Apache Hadoop infrastructure, such as Mahout and Dryad. The latter is more like necessary for real-time analytic for stream data applications. Storm and S4 are good examples for large scale streaming data analytic platforms. The interactive analysis processes the data in an interactive environment, allowing users to undertake their own analysis of information. The user is directly connected to the computer and hence can interact with it in real time. The data can be reviewed, compared and analyzed in tabular or graphic format or both at the same time. Google's Dremel and Apache Drill are Big Data platforms based on interactive analysis. In this paper a new stream reasoning framework is proposed applied to ships' new maritime environmental instrument MRV (Monitoring, Recording, Verifying) introduced by EU to IMO. The framework is providing in real time to maritime stakeholders awareness for a potential violation of existing maritime environmental thresholds

2. MRV Regulation

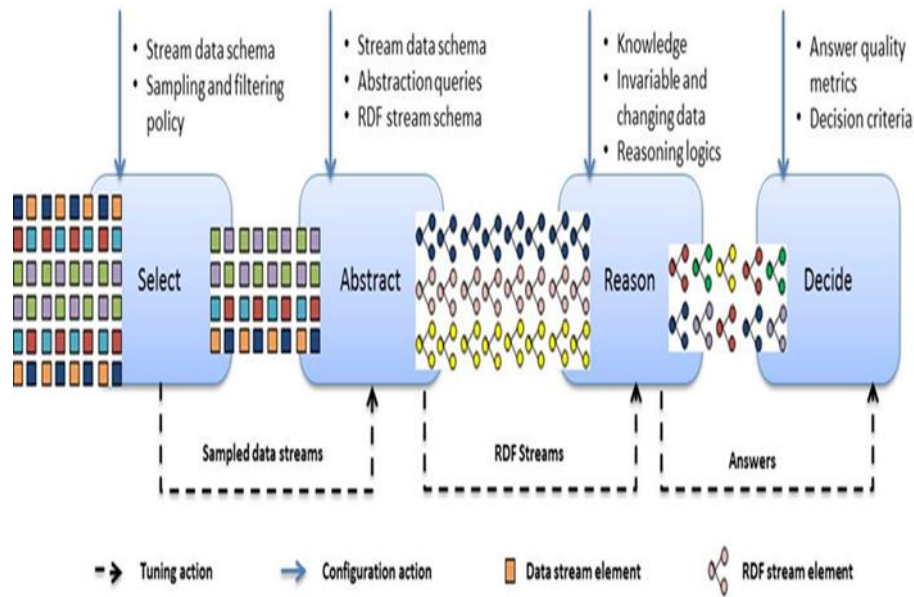


Figure 1: Information processing steps in a stream reasoner

Maritime transport has an impact on global climate change. International shipping remains the only means of transportation not currently included in the European Union's commitment to reducing greenhouse gas emissions [IMO, 2014]. In order to reduce emissions from shipping at the European Union level, a system is being introduced for the monitoring, reporting and verification (MRV Regulation) of CO₂ emissions based on the fuel consumption of ships. It is considered the first step of a staged approach for the inclusion of maritime transport emissions into the European Union's greenhouse gas reduction commitment, European Union's (EU) Regulation 2015/757 that is dealing with monitoring, re-orting and verification of carbon dioxide (CO₂) emissions in relation to maritime transport entered into force on July 1, 2015. The objective of this regulation, also known as the MRV Regulation, is to gain a better understanding of fuel consumption and CO₂ emissions from shipping activities within Europe. The MRV Regulation requires per-voyage and annual monitoring of CO₂ emissions; other parameters are also included, such as quantities of cargo carried and miles travelled. Furthermore, the annual disclosure of aggregated data on a ship-by-ship basis is required. Ships that are scheduled to perform more than 300 voyages per year or operate solely within Europe during the annual monitoring period are exempt from monitoring the parameters on a per-voyage basis. Instead, they can report on an annual basis in order to reduce the administrative burden. It is very important for every ship-owner to have an indication in real time on MRV's data to decide and take corrective actions accordingly. In the following sections we will describe a proposed framework to fulfil those needs.

3. Real time decision – stream reasoning

Decision support is a broad concept that prescribes using computerized systems and other tools to assist in individual, group, and organizational decision making. In the practice any system that processes and stores data or

reports them as they are happening is considered to be an on-line real-time system. More specific real time means that informational inputs to decision-making processes are available as soon as there are changes in the environment that alter those informational inputs. It should be clarified that real time also means 'near real time' in practice because there is always some latency between (a) the actual state change, (b) the reflection of that state change in data in one or more systems of record and (c) the availability of the changed data to decision makers." Real time is not the same for every decision task. These factors like data availability, data capture, and transmission speeds are less of a problem and technological advances provide new tools to receive and access real-time data. Technological progress such as Internet of things facilitate real-time decision support innovation. [Valle, 2008] Real-time decision support has six major capabilities : (1) Monitoring activity using decision rules to send alerts. (2) Fast exchange and transmission of data to systems and decision makers. (3) Information available to a decision maker as soon as a change occurs. (4). Processes and stores data and reports raw or summarized data as events happen (5). Provides analytics or knowledge at the point of use. (6). Provides synchronous exchange from data origination to data use. Stream reasoning is an approach that can be used if information arrives as a stream of (time stamped) inputs. The approach has two features that could be helpful:

- the knowledge base can be continuously updated and reasoning goals continuously re-evaluated as new assertions arrive;
- the reasoner considers events from a finite time window, and not only at a single instant.

Introducing stream reasoning in MRV data processing could therefore overcome some of the current limitations by:

- allowing the concrete system model to be continuously updated, which should be faster than generating a completely new model each time we need an update;
- reducing the time lag between the evolution of the real system and that of the concrete system model, making it possible to resolve recent and rapid changes in the real system;
- representing protracted as well as instantaneously observed behaviors in the model by including information over an extended time window;
- allowing reasoning algorithms to take account of system changes during the time window, target than only the instantaneous system composition and status.

However it is important to appreciate the research on stream reasoning is still in its infant [Stuckenschmidt H,2010]. Recent research efforts are still in early step and focus on the investigation of architecture approaches to support stream reasoning. As illustrated in Figure 1, stream reasoning consists of four main processing steps:

- **Select:** the first step in the stream reasoning to select relevant data from input streams by exploiting load-shedding techniques by introducing sampling policies that probabilistically drop stream elements to deal with bursty streams that may have unpredictable peaks.
- **Abstract:** the sampled streams are fed into the abstract step that generates aggregate events by enforcing aggregation queries continuously. Outputs of the abstract step are consolidated as RDF (Resource Description Framework) streams, an unbounded bag of pairs $\langle \rho, \tau \rangle$ where ρ is a RDF triple and τ is the timestamp that denotes the logical arrival time of RDF statement. This step entails the development of aggregate query language and system for query RDF data in the form of data streams [Barbieri D F, 2009].
- **Reason:** RDF streams are injected into background knowledge in order to perform reasoning tasks. Given that the reasoning process is not aware of expiration time, the reasoning results remain valid until the next update.
- **Decide:** Before producing answers to reasoning tasks, the answering process reaches the decision step where quality metrics and decision criteria defined by application developer are used to evaluate the quality of the answer is good enough and otherwise adapt the behavior of each previous step.

After some consideration, we concluded that it is not possible to use stream reasoning in a simple way to address MRV real time intelligence implementation. However, the underlying concepts can be used to enrich the proposed run-time architecture and to provide more flexibility

4. Proposed real time framework

In this subsection we briefly overview the approach taken in conceptual framework, This is designed to exploit semantic system models to enable the use of machine

reasoning to support the end user in making and implementing decisions at run-time. This translates into: · creating a semantic model of the running system based on the available monitoring data and using it to reason about the status of the system. presenting information from this model to the user, to help them understand and address current situation awareness risks [Kostopoulos, 2012]. The tools developed support machine-assisted design time system modelling, allowing its structure and properties to be described before the actual system is created by dynamic runtime composition. This model is called an abstract system model since it describes the structure of the system but not its actual composition. The proposed conceptual framework which schematically is given in Fig. 2. [Nikitakos,2016] Then constructs a concrete system model representing a snapshot of the running system, based on monitoring data and semantic reasoning over the abstract system model. Avoiding further analysis which would be beyond the scope of this work we mention that two separate reasoning processes are taking place:

1. Semantic reasoning for potential threat of anomaly behavior classification based on whether these are addressed by the controls present in the running system
2. Bayesian inference for likelihood estimation that each threat is currently being carried out. Within the proposed framework the user is presented with three types of information :
 1. What are the system vulnerabilities, or what threats is the system unable to manage
 2. What is the current likelihood probability each threat is being carried out
 3. What is the threat impact on the maritime emissions performance.

Moreover that's classified into three classes:

Blocked threat/activity if the system has the appropriate control to prevent the abnormal behavior to create any problem Mitigated activity when the abnormal behavior cannot be prevented, but the system controls provide a response that will counteract its effect on maritime safety and security. Vulnerability meaning the system does not have any means to prevent the abnormal behavior or counteract its effects on the targeted system asset.

The objectives of the monitoring and decision support tool are basically four.[IEC 2011]

1. Risk Classification (low, medium, high according their potential impact and blocked, mitigated, vulnerabilities depending on how well are addressed by controls)
2. Periodic assessment (the Decision Support Tool (DST) refreshes in a periodic fashion the model and dynamically reduces the involved risk factors)
3. Threat explanations (the DST provides explanation of threats which is very helpful to the operator in the loop for understanding the system and to take appropriate actions)

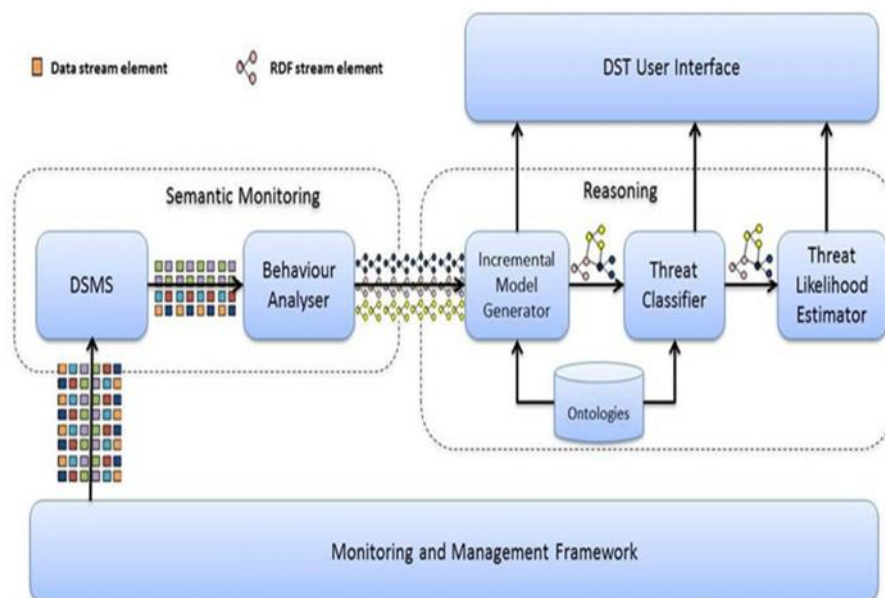


Figure 2: A proposed conceptual framework incorporating stream reasoning concepts

4. Propositions (the DST allows the operator to revert to past model versions when required allowing the user to make “what – if” tests on his model by adding controls and comparing the results with the original model). So the fault monitoring DST tool provides continuous feedback and suggests new control actions that can be useful while provides the capability to test their effect to “what – if” scenarios. Notice that the user is presented with the three vulnerability classifications: the good ones are to the left (blocked and mitigated threats) and the most troubling threats (vulnerabilities) are on the right. The core semantic language is OWL, the Web Ontology Language meaning that the models in the DST must be in OWL format. The version of the OWL language is OWL2. The support tool is built on JAVA 1.6 and SWT 3.738. Most web semantic projects are built on JAVA and this is the main reason JAVA is used in systems DST. The reasoner has a great role in the DST. The reasoner used is Hermit 1.3.5. There exist also other reasoners were used as well but they were unable to handle real and large volumes of data. Though Hermit so far manages well with the volume data, a new reasoner is designed in order to adapt reasoning to Bayesian inference used in the approach. Conclusively semantic models have been proved very useful in the application area of security and risk management of several critical infrastructures including maritime. The conceptual tool presented made this fact clear especially to the end users and decision makers

5. Conclusions

In the current analysis, a stream reasoning architecture was presented in order to give a warning signal to a shipping company related with measurements performed under the MRV’s framework. The concept of stream reasoning is useful for MRV data intelligence to improve current limitations. However, there are still many re-research challenges. To begin with, the improvement of Behavior Analyser component by using appropriate algorithms to convert raw monitoring data from MRV into RDF streams is necessary. The associated threat classifier should also be

extended, in order to handle different types of classifications, but still defined by SWRL rules. Additionally, the creation of a specialized classifier that would be more effective/faster than the general-purpose reasoner should be considered in order to improve performance. Furthermore, the “Bayesian threat likelihood estimator” implementation should be optimized, under the notion to reduce processing time. Finally, the abnormal activity hypothesis sampling should take into account any secondary effects. All these, are not simple tasks: evaluating performance of the conceptual framework in real conditions, as well as implementing the necessary improvements can be achieved only via extensive future research activities.

References

- Barbieri D F, Braga D, Ceri S, Della Valle E, Grossniklaus M. (2009) C-sparql: Sparql for continuous querying. In: Proceedings of the 18th international conference on World Wide Web, Madrid, Spain, 20-24 April 2009
- Kostopoulos D, Leventakis G, Tsoulkas V, Nikitakos N (2012) An Intelligent Fault Monitoring and Risk Management Tool for Complex Critical Infrastructures. Paper presented at the SERSCIS Approach in Air-Traffic Surface Control, UKSim 14th International Conference on Modelling and Simulation, 28-30 March 2012
- Nikitakos N (2016) Real Time Intelligence of AIS data. presented at Maritime Knowledge Discovery and Anomaly Detection Workshop, JRC-Ispira, 5-6 July 2016
- Valle E, Ceri S, Barbieri D F, Braga D, Campi A (2008) A First Step Towards Stream Reasoning. Paper presented at the Future Internet Symposium, Vienna, Austria, 28-30 September 2008
- Stuckenschmidt H, Ceri S, Valle E D, Harmenlen F V (2010) Towards Expressive Stream Reasoning. Paper presented at the Dagstuhl Seminar on Semantic Aspects of Sensor Networks, Dagstuhl, Germany, 24 - 29 January 2010
- Pellet: OWL 2 Reasoner for Java. <http://pellet.owdl.com>.
- W3C OWL Working Group. OWL 2 Web Ontology Language: Document Overview. <http://www.w3.org/TR/owl2-overview/>.

Third IMO GHG Study (2014); International Maritime Organization (IMO) London, UK, April 2015; Smith, T. W. P.; Jalkanen, J. P.; Anderson, B. A.; Corbett, J. J.; Faber, J.; Hanayama, S.; O'Keeffe, E.; Parker, S.; Johansson, L.; Aldous, L.; Raucci, C.; Traut, M.; Ettinger, S.; Nelissen, D.; Lee, D. S.; Ng, S.; Agrawal, A.; Winebrake, J. J.; Hoen, M.; Chesworth, S.; Pandey, A.

ISO/IEC 27005:2011; Information technology -- Security techniques -- Information security risk management, International Organization for Standardization, 2011.