Chapter 9
Event Processing for Maritime Situational Awareness

Manolis Pitsikalis, Konstantina Bereta, Marios Vodas, Dimitris Zissis, and Alexander Artikis

Abstract Numerous illegal and dangerous activities take place at sea, including violations of ship emission rules, illegal fishing, illegal discharges of oil and garbage, smuggling, piracy and more. We present our efforts to combine two stream reasoning technologies for detecting such activities in real time: a formal, computational framework for composite maritime event recognition, based on the Event Calculus, and an industry-strong maritime anomaly detection service, capable of processing daily real-world data volumes.

9.1 Introduction

Numerous illegal and dangerous activities take place at sea, such as pollution (illegal discharges of oil and garbage, violations of ship emission rules, etc.), illegal fishing, smuggling (drugs, arms, oil, etc.), piracy and many more [8]. Often the vessels involved in such activities attempt to behave as common commercial ships, concealing their true intentions. Unlike in the past, though, when there was no way of detecting these activities as they happened, today numerous monitoring systems,

M. Pitsikalis
Institute of Informatics & Telecommunications, NCSR Demokritos, Athens, Greece
e-mail: manospits@iit.demokritos.gr

K. Bereta · M. Vodas
MarineTraffic, Athens, Greece
e-mail: konstantina.bereta@marinetraffic.com; marios.vodas@marinetraffic.com

D. Zissis
Department of Product & Systems Design Engineering, University of the Aegean, Syros, Greece

MarineTraffic, Athens, Greece
e-mail: dzissis@marinetraffic.com

A. Artikis (✉)
Department of Maritime Studies, University of Piraeus, Piraeus, Greece

Institute of Informatics & Telecommunications, NCSR Demokritos, Athens, Greece
e-mail: a.artikis@unipi.gr

G. A. Vouros et al. (eds.), *Big Data Analytics for Time-Critical Mobility Forecasting*, https://doi.org/10.1007/978-3-030-45164-6_9

such as the automatic identification system (AIS), produce constant streams of surveillance data, often revealing the true intentions of suspicious vessels.

"Collaborative" monitoring systems use equipment that is mostly installed aboard the vessels that should be monitored and rely on the collaboration of the vessels' crew. This equipment is used for reporting the position of the vessel, along with information about its navigational status (e.g., destination, speed, heading, course over ground, etc.). The automatic identification system (AIS) [6], for example, is such a system and is based on the transmission of messages through VHF transponders installed aboard the vessels themselves and received using VHF receivers installed on other vessels sailing close by, coastal stations and/or satellites. Other collaborative systems include the long-range identification and tracking (LRIT) system [7], and the vessel monitoring system (VMS) [4], which is a system designed for managing fishing activity. 'Non-collaborative' monitoring systems, on the other hand, do not rely on the collaboration of the crew and include systems such as coastal and high-frequency radar, active and passive sonar, ground- and vessel-based (e.g., thermal) cameras, satellite and airborne Earth Observation systems, such as optical and synthetic aperture radar systems.

The data streams produced by maritime monitoring systems may be consumed by stream reasoning systems, in order to support *Maritime Situational Awareness* [20, 21], i.e. the effective understanding of activities, events and threats in the maritime environment that could impact the global safety, security, economic activity and the environment (Chap. 1 of this book provides further details on maritime data sources and operational needs). Terosso-Saenz et al. [18], for example, presented a system detecting abnormally high or low vessel speed, as well as when two vessels are in danger of colliding. SUMO [5] is an open-source system combining AIS streams with synthetic aperture radar images for detecting illegal oil dumping, piracy and unsustainable fishing. van Laere et al. [19] evaluated a workshop aiming at the identification of potential vessel anomalies, such as tampering, rendez-vous between vessels and unusual routing. Mills et al. [14] described a method for identifying trawling using speed and directionality rules, thus helping in studies of how trawling impacts on species, habitats and the ecosystem. Millefiori et al. [13] proposed a distributed framework that uses AIS data to identify port operational regions.

In this chapter, we present our effort to combine two stream reasoning technologies for maritime situational awareness. First, a formal, computational framework for composite maritime event recognition, based on the Event Calculus [16].[1] Second, an industry-strong maritime anomaly detection service, processing daily real-world data volumes [8].[2] Our integrated system aims to pave the way for the real-time recognition of a wide variety of maritime events of high significance, including forms of illegal and suspicious vessel activity. We build on our previous work in this domain by presenting the architecture and implementation details of our approach, as used in real-world operational conditions to analyse actual

[1] http://cer.iit.demokritos.gr/cermm.

[2] www.marinetraffic.com.

sensor data, received from the MarineTraffic network. Our novel approach combines automated reasoning (based on complex event processing) with massive amounts of historical observational data used for the extraction of contextual information, so as to improve maritime situational awareness and assist an operational end user in the decision-making process by providing real-time notifications. The novel data processing architecture presented here supports large-scale multistage analysis of data generated by a distributed network of sensors and in situ data (from computations). In this context we address the modalities of processing combinations of data stored in massive observational archives and streaming data. The multistage workflow demonstrates how several state-of-the-art technologies can be combined (synopsis engine, distributed processing, etc.) to achieve real-time response times.

The remainder of this chapter is structured as follows: Section 9.2 presents the proposed architecture of the integrated stream reasoning technology. Then, Sect. 9.3 presents a set of maritime events that may be identified by our technology. Subsequently, Sect. 9.4 presents a service making available the detected events to users. Finally, in Sect. 9.5 we summarise our work and present further work directions.

9.2 System Architecture

9.2.1 Setting the Scene

We support maritime situational awareness following two online tasks/steps: (a) computing a set of spatial relations among vessels, such as proximity, and among vessels and areas of interest (e.g., fishing areas), as described in Chap. 6 of this book and (b) labelling position signals of interest as 'critical'—such as when a vessel changes its speed, turns, stops, moves slowly or stops transmitting its position, as Chap. 4 of this book describes. Figure 9.1 illustrates these steps. Streaming-in AIS position signals go through a spatial preprocessing step, for the computation of the spatial relations required by maritime situational awareness [17]. These relations are displayed at the top of Table 9.1. Then, the relevant position signals are annotated as critical—see the middle part of Table 9.1. Subsequently, the position signals may be consumed by our stream reasoning technology either directly (see 'enriched AIS stream' in Fig. 9.1) or after being compressed, that is, after removing all signals that have not been labelled as critical (see 'critical point stream' in Fig. 9.1).

Critical point labelling is performed as part of trajectory synopsis generation, whereby major changes along each vessel's movement are tracked (cf. Chap. 4). This process instantly identifies critical points along each trajectory, such as a stop, a turn, or slow motion. Using the retained critical points, we may reconstruct a vessel trajectory with small acceptable deviations from the original one. Empirical results have indicated that 70–80% of the input data may be discarded as redundant, while

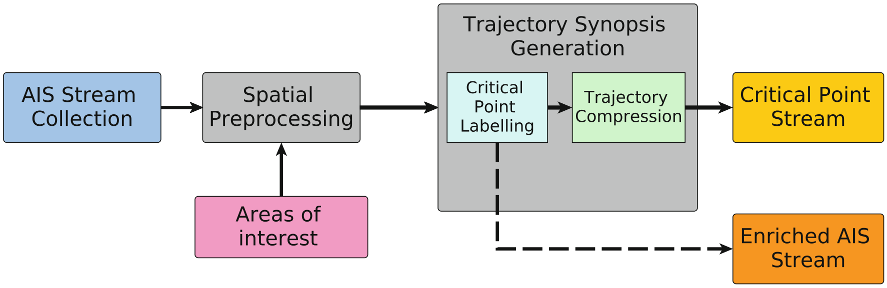

Fig. 9.1 Steps required for maritime situational awareness

compression ratio can be up to 99% when the frequency of position updates is high [15].

9.2.2 *Maritime Stream Reasoning*

We have been developing two stream reasoning technologies for maritime situational awareness. More precisely, we have been constructing a formal, computational framework for composite maritime event recognition, based on RTEC [1], a logic programming implementation of the Event Calculus [16]. RTEC is designed to compute continuous narrative assimilation queries for pattern matching on data streams and includes several optimisation techniques allowing for real-time event recognition. To facilitate the interaction of RTEC with state-of-the-art distributed systems, we have been re-implementing RTEC in the Scala programming language. This way, we may integrate RTEC with the industry-strong maritime anomaly detection service of MarineTraffic [8]. This service is based on a hybrid architecture, comprised of stream and batch processing components. The stream processing component is based on the actor model—specifically, the Akka framework—for concurrency and event-driven processing. In what follows, we briefly present RTEC and illustrate its use for maritime stream reasoning. Then, in Sect. 9.4 we present MarineTraffic's service, and the way these two stream reasoning technologies are being integrated.

The 'Event Calculus for Run-Time reasoning' (RTEC) is an Event Calculus dialect optimised for composite event recognition over high-velocity data streams [1]. For example, RTEC may detect the composite events displayed at the bottom of Table 9.1. The time model in RTEC is linear and includes integer time-points. An *event description* includes rules that define the event instances with the use of the happensAt(predicate, the effects of events on *fluents*—time-varying properties—with the use of the initiatedAt(and terminatedAt(predicates and the values of the fluents with the use of the holdsAt(and holdsFor(predicates. Table 9.2 summarises the main predicates of RTEC.

Table 9.1 Events for maritime situational awareness: Input events are presented above the double horizontal line, while the output stream is presented below this line. The input events above the single horizontal line are detected at the spatial preprocessing step, while the remaining ones are detected by the trajectory synopsis generator (critical events). With the exception of *proximity*, all items of the input stream are instantaneous, while all output activities are durative

		Event/Activity	Description
Input	Spatial	*entersArea*(V, A)	Vessel V enters area A
		leavesArea(V, A)	Vessel V leaves area A
		proximity(V_1, V_2)	Vessels V_1 and V_2 are close
	Critical	*gap_start*(V)	Vessel V stopped sending position signals
		gap_end(V)	Vessel V resumed sending position signals
		slow_motion_start(V)	Vessel V started moving at a low speed
		slow_motion_end(V)	Vessel V stopped moving at a low speed
		stop_start(V)	Vessel V started being idle
		stop_end(V)	Vessel V stopped being idle
		change_in_speed_start(V)	Vessel V started changing its speed
		change_in_speed_end(V)	Vessel V stopped changing its speed
		change_in_heading(V)	Vessel V changed its heading
Output	Composite	*highSpeedNC*(V)	Vessel V has high speed near coast
		anchoredOrMoored(V)	Vessel V is anchored or moored
		drifting(V)	Vessel V is drifting
		trawling(V)	Vessel V is trawling
		tugging(V_1, V_2)	Vessels V_1 and V_2 are engaged in tugging
		pilotBoarding(V_1, V_2)	Vessels V_1 and V_2 are engaged in pilot boarding
		rendez-Vous(V_1, V_2)	Vessels V_1 and V_2 are having a rendez-vous
		loitering(V)	Vessel V is loitering
		sar(V)	Vessel V is engaged in a search and rescue (SAR) operation

Fluents are 'simple' or 'statically determined'. In brief, simple fluents are defined by means of initiatedAt(and terminatedAt(rules, while statically determined fluents are defined by means of application-dependent holdsFor(rules, along with the interval manipulation constructs of RTEC: union_all(, intersect_all(and relative_complement_all(. See Table 9.2 for a brief explanation of these constructs and Fig. 9.2 for an example visualisation. Composite events/activities are typically durative; thus, the task generally is to compute the maximal intervals for which a fluent expressing a composite activity has a particular value continuously. Below, we discuss the representation of fluents/composite maritime activities and briefly present the way in which we compute their maximal intervals.

Table 9.2 Main predicates of RTEC. '$F = V$' denotes that fluent F has value V

Predicate	Meaning
happensAt(E, T)	Event E occurs at time T
holdsAt($F = V, T$)	The value of fluent F is V at time T
holdsFor($F = V, I$)	I is the list of the maximal intervals for which $F = V$ holds continuously
initiatedAt($F = V, T$)	At time T a period of time for which $F = V$ is initiated
terminatedAt($F = V$, T)	At time T a period of time for which $F = V$ is terminated
union_all(L, I)	I is the list of maximal intervals produced by the union of the lists of maximal intervals of list L
intersect_all(L, I)	I is the list of maximal intervals produced by the intersection of the lists of maximal intervals of list L
relative_complement_all (I', L, I)	I is the list of maximal intervals produced by the relative complement of the list of maximal intervals I' with respect to every list of maximal intervals of list L

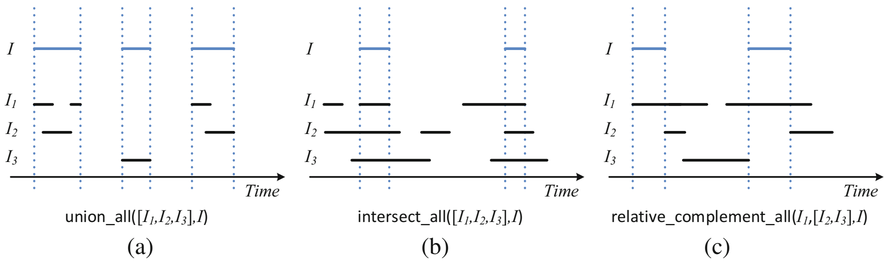

Fig. 9.2 A visual illustration of the interval manipulation constructs of RTEC. In these examples, there are three input streams, I_1, I_2 and I_3, coloured black. The output of each interval manipulation construct I is coloured light blue. (**a**) Union. (**b**) Intersection. (**c**) Relative complement

9.3 Maritime Events

To analyse the behaviour at sea, we need to spatially define the implicated sites, such as ports, sea structures, fishing areas and sea zones. In [13], for example, we presented a data-driven implementation of a distributed method for calculating port operational areas and other activity areas. Then, we need to define the spatiotemporal patterns of vessel activity of interest. Such patterns have been documented in various papers, including [8, 13, 16, 18, 19]. In this section, we present a set of indicative patterns in the language of RTEC following [16]. The hierarchy supported by our integrated stream reasoning technology is displayed in Fig. 9.3. In this figure, an arrow from pattern A to pattern B denotes that A is used in the specification of B.

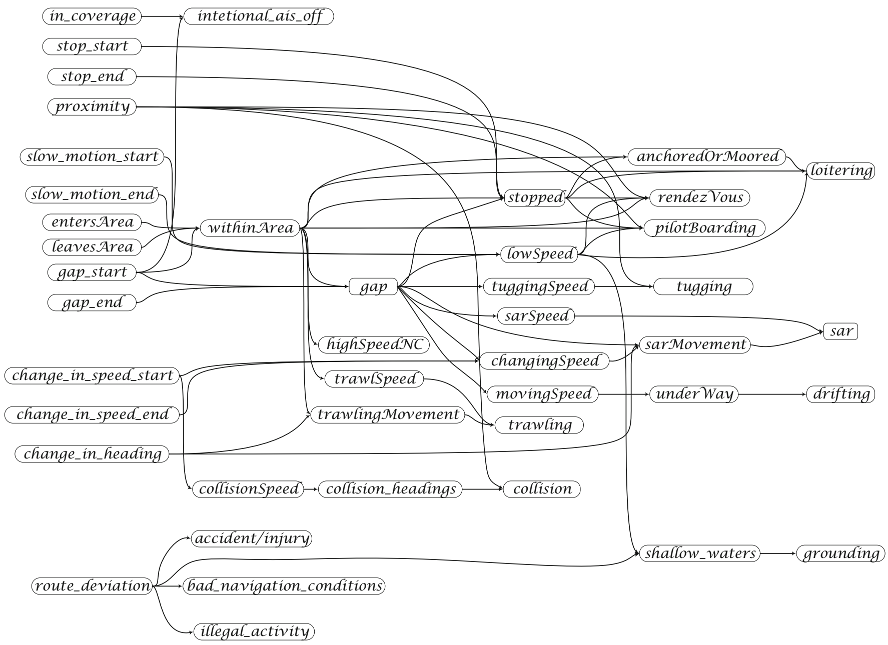

Fig. 9.3 Event hierarchy

9.3.1 Building Blocks

We begin by presenting a set of building blocks that will be later used for the construction of more involved patterns.

9.3.1.1 Vessel Within Area of Interest

Calculating the intervals during which a vessel is in an area of some type, such as a (protected) Natura 2000, fishing or anchorage area, is particularly useful in maritime (e.g., fishing) patterns. Consider the formalisation below:

$$\begin{aligned}
&\mathsf{initiatedAt}(withinArea(Vessel, AreaType)=\mathsf{true},\ T) \leftarrow\\
&\quad \mathsf{happensAt}(entersArea(Vessel,\ AreaID),\ T),\\
&\quad areaType(AreaID,\ AreaType).\\
&\mathsf{terminatedAt}(withinArea(Vessel, AreaType)=\mathsf{true},\ T) \leftarrow\\
&\quad \mathsf{happensAt}(leavesArea(Vessel, AreaID),\ T),\\
&\quad areaType(AreaID,\ AreaType).\\
&\mathsf{terminatedAt}(withinArea(Vessel, AreaType)=\mathsf{true},\ T) \leftarrow\\
&\quad \mathsf{happensAt}(gap_start(Vessel),\ T).
\end{aligned} \tag{9.1}$$

Variables start with an uppercase letter, while predicates and constants start with a lower-case letter. $withinArea(Vessel, AreaType)$ is a simple fluent indicating that a *Vessel* is within an area of some type. $entersArea(Vessel, AreaID)$ and $leavesArea(Vessel, AreaID)$ are input events computed at the spatial preprocessing step (see the top part of Table 9.1), indicating that a $Vessel$ entered (respectively, left) an area with *AreaID*. $areaType(AreaID, AreaType)$ is an atemporal predicate storing the areas of interest of a given dataset. $withinArea(Vessel, AreaType)=\mathsf{true}$ is initiated when a *Vessel* enters an area of *AreaType* and terminated when the *Vessel* leaves the area of *AreaType*. $withinArea(Vessel, AreaType)=\mathsf{true}$ is also terminated when the trajectory synopsis generator produces a *gap_start* event (see the middle part of Table 9.1), indicating the beginning of a communication gap (in the subsection that follows we discuss further communication gaps). In this case we chose to make no assumptions about the location of the vessel. With the use of rule-set (9.1), RTEC computes the *maximal intervals* during which a vessel is said to be within an area of some type.

9.3.1.2 Communication Gap

According to the trajectory synopsis generator, a communication gap takes place when no message has been received from a vessel for at least 30 min. All numerical thresholds, however, may be tuned—for example, by machine learning algorithms—to meet the requirements of the application under consideration. A communication gap may occur when a vessel sails in an area with no AIS receiving station nearby, or because the transmission power of its transceiver allows broadcasting in a shorter range, or when the transceiver is deliberately turned off. The rules below present a formalisation of communication gap:

$$\begin{aligned}
&\mathsf{initiatedAt}(gap(Vessel) = nearPorts,\ T) \leftarrow \\
&\quad \mathsf{happensAt}(gap_start(Vessel),\ T), \\
&\quad \mathsf{holdsAt}(withinArea(Vessel, nearPorts) = true,\ T). \\
&\mathsf{initiatedAt}(gap(Vessel) = farFromPorts,\ \ T) \leftarrow \\
&\quad \mathsf{happensAt}(gap_start(Vessel),\ T), \\
&\quad \mathsf{not\ holdsAt}(withinArea(Vessel, nearPorts) = true,\ T). \\
&\mathsf{terminatedAt}(gap(Vessel) = _Value,\ T) \leftarrow \\
&\quad \mathsf{happensAt}(gap_end(Vessel),\ T).
\end{aligned} \tag{9.2}$$

gap is a simple, multi-valued fluent, *gap_start* and *gap_end* are input critical events (see Table 9.1), 'not' expresses Prolog's negation-by-failure [3], while variables starting with '_', such as *_Value*, are free. We chose to distinguish between communication gaps occurring near ports from those occurring in the open sea, as the first ones usually do not have a significant role in maritime monitoring. According to rule-set (9.2), a communication gap is said to be initiated when the synopsis generator emits a 'gap start' event and terminated when a 'gap end' is detected. Given this rule-set, RTEC computes the maximal intervals for which a vessel is not sending position signals.

9.3.2 Maritime Situational Indicators

Our aim is to detect in real-time maritime situational indicators [9], that is, composite maritime activities of special significance, building upon the blocks presented above. As mentioned earlier, Fig. 9.3 displays the hierarchy of our formalisation, that is, the relations between the indicators' specifications.

9.3.2.1 Vessel with High Speed Near Coast

Several countries have regulated maritime zones. In French territorial waters, for example, there is a 5 knots speed limit for vessels or watercrafts within 300 m from the coast. One of the causes of marine accidents near the coast is vessels sailing with high speed; thus, the early detection of violators ensures safety by improving the efficiency of law enforcement. Figure 9.4 displays the case of a vessel not conforming to the above regulations. Consider the following formalisation:

$$\begin{aligned}
&\mathsf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vesseltext}) = \mathsf{true},\ T) \leftarrow \\
&\quad \mathsf{happensAt}(\mathit{velocity}(\mathit{Vessel},\ \mathit{Speed},\ _CoG,\ _\mathit{TrueHeading}),\ T), \\
&\quad \mathsf{holdsAt}(\mathit{withinArea}(\mathit{Vessel},\ \mathit{nearCoast}) = \mathsf{true},\ T), \\
&\quad \mathit{threshold}(v_{hs},\ V_{hs}), \quad \mathit{Speed} > V_{hs}. \\
&\mathsf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathsf{true},\ T) \leftarrow \\
&\quad \mathsf{happensAt}(\mathit{velocity}(\mathit{Vessel},\ \mathit{Speed},\ _CoG,\ _TrueHeading),\ T), \\
&\quad \mathit{threshold}(v_{hs},\ V_{hs}), \quad \mathit{Speed} \leq V_{hs}. \\
&\mathsf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathsf{true},\ T) \leftarrow \\
&\quad \mathsf{happensAt}(\mathsf{end}(\mathit{withinArea}(\mathit{Vessel},\ nearCoast) = \mathsf{true}),\ T).
\end{aligned} \tag{9.3}$$

highSpeedNC(*Vessel*) is a Boolean simple fluent indicating that a *Vessel* is exceeding the speed limit imposed near the coast. *velocity* is input contextual information expressing the speed, course over ground (CoG) and true heading of a vessel. This information is attached to each incoming AIS message. Recall that variables starting with '_' are free. $\mathit{withinArea}(\mathit{Vessel}, nearCoast) = \mathsf{true}$ expresses the time periods during which a *Vessel* is within 300 m from the French coastline (see rule-set (9.1) for the specification of *withinArea*). *threshold* is an auxiliary atemporal predicate recording the numerical thresholds of the maritime patterns. The use of this predicate supports code transferability, since the use of different thresholds for different applications requires only the modification of the *threshold* predicate, and not the modification of the patterns. $\mathsf{end}((F = V)$ (respectively, $\mathsf{start}((F = V))$ is an RTEC built-in event indicating the ending (resp. starting) points for which $F = V$ holds

Fig. 9.4 A vessel near the port of Brest, France, with speed above the 5 knots limit. The marked circles denote the AIS position signals that are labelled as 'critical' by the synopsis generator

continuously. According to rule-set (9.3), therefore, $highSpeedNC(Vessel) = \mathsf{true}$ is initiated when the *Vessel* sails within 300 m from the French coastline with speed above 5 knots and terminated when its speed goes below 5 knots, sails away (further than 300 m) from the coastline or stops sending position signals (recall that *withinArea* is terminated/ended by *gap_start*).

9.3.2.2 Anchored or Moored Vessel

A vessel lowers its anchor in specific areas; for example, when waiting to enter into a port or taking on cargo or passengers where insufficient port facilities exist. Figure 9.5 displays an example of a vessel stopped in an anchorage area. Furthermore, a vessel may be moored, that is, when a vessel is secured with ropes in any kind of permanent fixture such as a quay or a dock. Consider the specification below:

$$
\begin{aligned}
&\mathsf{holdsFor}(anchoredOrMoored(Vessel(=\mathsf{true}, I) \leftarrow \\
&\quad \mathsf{holdsFor}(stopped(Vessel(=farFromPorts,\ I_{sffp}), \\
&\quad \mathsf{holdsFor}(withinArea(Vessel, anchorage(=true,\ I_{wa}), \\
&\quad \mathsf{intersect\_all}([I_{sffp}, I_{wa}], I_{sa}), \\
&\quad \mathsf{holdsFor}(stopped(Vessel) = nearPorts,\ I_{sn}), \\
&\quad \mathsf{union\_all}([I_{sa}, I_{sn}], I_i), \\
&\quad threshold(v_{aorm}, V_{aorm}), \quad intDurGreater(I_i, V_{aorm}, I).
\end{aligned}
\tag{9.4}
$$

$anchoredOrMoored(Vessel)$ is a statically determined fluent, that is, it is specified by means of a domain-dependent $\mathsf{holdsFor}($ predicate and interval manipulation

Fig. 9.5 Anchored vessel

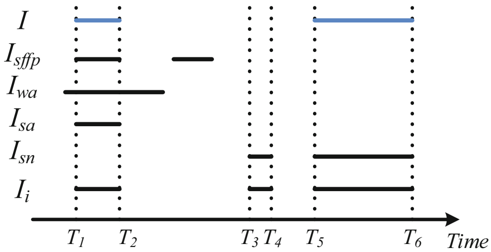

Fig. 9.6 Interval computation example of *anchoredOrMoored*

constructs—intersect_all(and union_all(in this case, that compute, respectively, the intersection and union of lists of maximal intervals (see Table 9.2 and Fig. 9.2). *stopped* is a fluent recording the intervals in which a vessel is stopped—this may be far from all ports or near some port(s). $intDurGreater(I', V_t, I)$ is an auxiliary predicate keeping only the intervals I of list I' with length greater than V_t. *anchoredOrMoored*(*Vessel*)= true, therefore, holds when the *Vessel* is stopped in an anchorage area or near some port, for a time period greater than some threshold (see V_{aorm} in rule (9.4)). The default value for this threshold is set to 30 min, as suggested by domain experts.

Figure 9.6 illustrates with the use of a simple example the computation of the *anchoredOrMoored* intervals. The displayed intervals I, I_{sffp}, etc., correspond to the intervals of rule (9.4). In the example of Fig. 9.6, the second interval of I_i, $[T_3, T_4]$, is discarded since it is not long enough according to the V_{aorm} threshold.

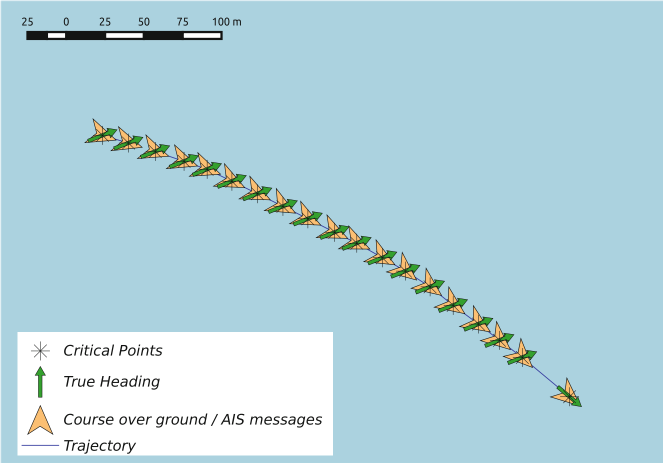

Fig. 9.7 A drifting vessel. In this example, all AIS position signals have been labelled as 'critical' (*change_in_heading*(

9.3.2.3 Drifting Vessel

A vessel is drifting when its course over ground, that is, the direction calculated by the GPS signal, is heavily affected by sea currents or harsh weather conditions, or when the vessel is not under control (say due to engine failure). Typically, as illustrated in Fig. 9.7, when the course over ground deviates from the true heading of a sailing vessel, that is, the direction of the ship's bow, then the vessel is considered drifting. Consider the formalisation below:

$$\begin{aligned}
&\mathsf{initiatedAt}(\mathit{drifting}(\mathit{Vessel}) = \mathsf{true},\ T) \leftarrow \\
&\quad \mathsf{happensAt}(\mathit{velocity}(\mathit{Vessel},\ _Speed,\ CoG,\ \mathit{TrueHeading}),\ T), \\
&\quad \mathit{angleDiff}(CoG,\ \mathit{TrueHeading},\ Ad), \\
&\quad \mathit{threshold}(v_{ad},\ V_{ad}), \quad Ad > V_{ad}, \\
&\quad \mathsf{holdsAt}(\mathit{underWay}(\mathit{Vessel}) = \mathsf{true},\ T). \\
&\mathsf{terminatedAt}(\mathit{drifting}(\mathit{Vessel}) = \mathsf{true},\ T) \leftarrow \\
&\quad \mathsf{happensAt}(\mathit{velocity}(\mathit{Vessel},\ _Speed,\ CoG,\ \mathit{TrueHeading}),\ T), \\
&\quad \mathit{angleDiff}(CoG,\ \mathit{TrueHeading},\ Ad), \\
&\quad \mathit{threshold}(v_{ad},\ V_{ad}), \quad Ad \leq V_{ad}. \\
&\mathsf{terminatedAt}(\mathit{drifting}(\mathit{Vessel}) = \mathsf{true},\ T) \leftarrow \\
&\quad \mathsf{happensAt}(\mathit{end}((\mathit{underWay}(\mathit{Vessel}) = \mathsf{true}),\ T).
\end{aligned} \tag{9.5}$$

drifting is a Boolean simple fluent, while, as mentioned earlier, *velocity* is input contextual information, attached to each AIS message, expressing the speed, course over ground (CoG) and true heading of a vessel. *angleDiff*(A, B, C) is an auxiliary predicate calculating the absolute minimum difference C between two angles A and B. The use of *underWay* in the initiation and termination conditions of *drifting* (see rule-set (9.5)) expresses the constraint that only moving vessels can be considered to be drifting.

9.3.2.4 Tugging

A vessel that should not move by itself, such as a ship in a crowded harbour or a narrow canal, or a vessel that cannot move by itself is typically pulled or towed by a tug boat. Figure 9.8 shows an example. During tugging, the two vessels are typically close and their speed is lower than normal, for safety and manoeuvrability reasons. We have formalised tugging as follows:

$$\begin{aligned}
&\mathsf{holdsFor}(\mathit{tugging}(\mathit{Vessel}_1, \mathit{Vessel}_2) = \mathsf{true}, I) \leftarrow \\
&\quad \mathit{oneIsTug}(\mathit{Vessel}_1, \mathit{Vessel}_2), \\
&\quad \mathsf{holdsFor}(\mathit{proximity}(\mathit{Vessel}_1, \mathit{Vessel}_2) = \mathsf{true},\ I_p), \\
&\quad \mathsf{holdsFor}(\mathit{tuggingSpeed}(\mathit{Vessel}_1) = \mathsf{true}, I_{ts1}), \\
&\quad \mathsf{holdsFor}(\mathit{tuggingSpeed}(\mathit{Vessel}_2) = \mathsf{true}, I_{ts2}), \\
&\quad \mathsf{intersect_all}([I_p, I_{ts1}, I_{ts2}], I_i), \\
&\quad \mathit{threshold}(v_{tug}, V_{Tug}), \quad \mathit{intDurGreater}(I_i, V_{Tug}, I).
\end{aligned} \tag{9.6}$$

tugging is a relational fluent referring to a pair of vessels, as opposed to the fluents presented so far that concern a single vessel. *oneIsTug*(V_1, V_2) is an auxiliary predicate stating whether one of the vessels V_1, V_2 is a tug boat. *proximity* is a durative input fluent computed at the spatial preprocessing step (see Table 9.1), expressing the time periods during which two vessels are ‘close’ (that is, their distance is less than a 100 m). *tuggingSpeed* is a simple fluent expressing the intervals during which a vessel is said to be sailing at tugging speed. According to rule (9.6), two vessels are said to be engaged in tugging if one of them is a tug boat, and, for at least V_{Tug} time-points, they are close to each other and sail at tugging speed.

9.3.2.5 Vessel rendez-vous

A scenario that may indicate illegal activities, such as illegal cargo transfer, is when two vessels are nearby in the open sea, stopped or sailing at a low speed. See Fig. 9.9 for an illustration. A specification of a potential ‘rendez-vous’, or

Fig. 9.8 Example of bulk carrier tugging. In this example, all position signals are labelled as 'critical'

'ship-to-ship transfer', may be found below:

$$
\begin{aligned}
&\mathsf{holdsFor}(\mathit{rendez\text{-}Vous}(\mathit{Vessel}_1, \mathit{Vessel}_2) = \mathsf{true},\ I) \leftarrow \\
&\quad \mathsf{not}\mathit{oneIsTug}(\mathit{Vessel}_1, \mathit{Vessel}_2), \\
&\quad \mathsf{holdsFor}(\mathit{proximity}(\mathit{Vessel}_1, \mathit{Vessel}_2) = \mathsf{true},\ I_p), \\
&\quad \mathsf{holdsFor}(\mathit{lowSpeed}(\mathit{Vessel}_1) = \mathit{true},\ I_{l1}), \\
&\quad \mathsf{holdsFor}(\mathit{stopped}(\mathit{Vessel}_1) = \mathit{farFromPorts},\ I_{s1}), \\
&\quad \mathsf{union\_all}([I_{l1}, I_{s1}], I_1), \\
&\quad \mathsf{holdsFor}(\mathit{lowSpeed}(\mathit{Vessel}_2) = \mathit{true},\ I_{l2}), \\
&\quad \mathsf{holdsFor}(\mathit{stopped}(\mathit{Vessel}_2) = \mathit{farFromPorts},\ I_{s2}), \\
&\quad \mathsf{union\_all}([\mathrm{I}_{l2}, I_{s2}], I_2), \\
&\quad \mathsf{intersect\_all}([I_1, I_2, I_p], I_f), \\
&\quad \mathsf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}_1, \mathit{nearPorts}) = \mathit{true},\ I_{np1}), \\
&\quad \mathsf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}_2, \mathit{nearPorts}) = \mathit{true},\ I_{np2}), \\
&\quad \mathsf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}_2, \mathit{nearCoast}) = \mathit{true},\ I_{nc1}), \\
&\quad \mathsf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}_2, \mathit{nearCoast}) = \mathit{true},\ I_{nc2}), \\
&\quad \mathsf{relative\_complement\_all}(I_f, [I_{np1}, I_{np2}, I_{nc1}, I_{nc2}], I_i), \\
&\quad \mathit{threshold}(v_{rv}, V_{rv}), \quad \mathit{intDurGreater}(I_i, V_{rv}, I).
\end{aligned}
\tag{9.7}
$$

rendez-Vous is a relational fluent, while *lowSpeed* is a fluent recording the intervals during which a vessel sails with a speed between 0.5 and 5 knots. relative_complement_all(is an interval manipulation construct of RTEC (see Table 9.2 and Fig. 9.2). According to rule (9.7), $rendez - Vous(V_1, V_2)$ holds when neither of the two vessels V_1, V_2 is a tug boat, V_1, V_2 are close to each other, and they are

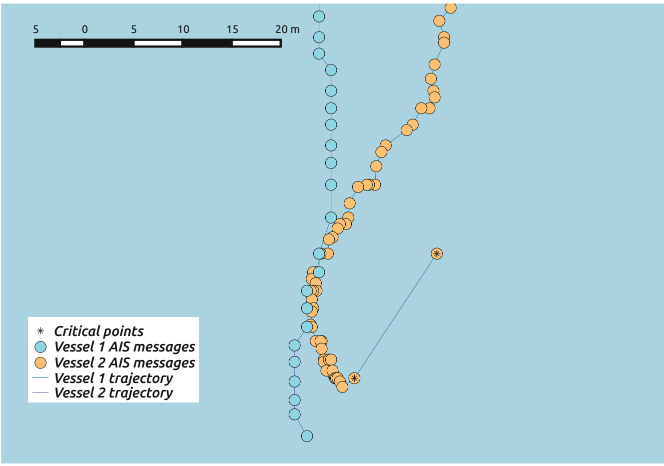

Fig. 9.9 Fishing vessels in close proximity. In this example, the vessels started sailing at a low speed before they came close to each other. Hence, these critical events (*slow_motion_start*) are not displayed in the figure

stopped or sail at low speed far from the coast and ports. Depending on the chosen distance thresholds for *nearCoast* and *nearPorts*, a vessel may be 'far' from the coastline and at the same time 'near' some port. Moreover, a vessel may be 'far' from all ports and 'near' the coastline.

We require that the two vessels are not near the coastline since illegal ship-to-ship transfer typically takes place far from the coast. We also require that both vessels are far from ports, as two slow moving or stopped vessels near some port would probably be moored or departing from the port.

9.4 Anomaly Detection Service

As mentioned earlier, we have been re-implementing RTEC in the Scala programming language, in order to integrate it with the industry-strong maritime anomaly detection service of MarineTraffic [8]. MarineTraffic is currently the world's leading platform offering vessel tracking services and actionable maritime intelligence. It offers an end-to-end service that tracks vessel positions across the globe based on AIS and disseminates this information to the general public though its interactive website, www.MarineTraffic.com. With an open community network of more than 3200 coastal AIS stations, MarineTraffic is capable of tracking vessels on their journeys across the coastlines of more than 140 countries. While the MarineTraffic terrestrial-based AIS network provides excellent coverage of several thousands of ports, the limited range of AIS results in restricted ocean coverage. To address this,

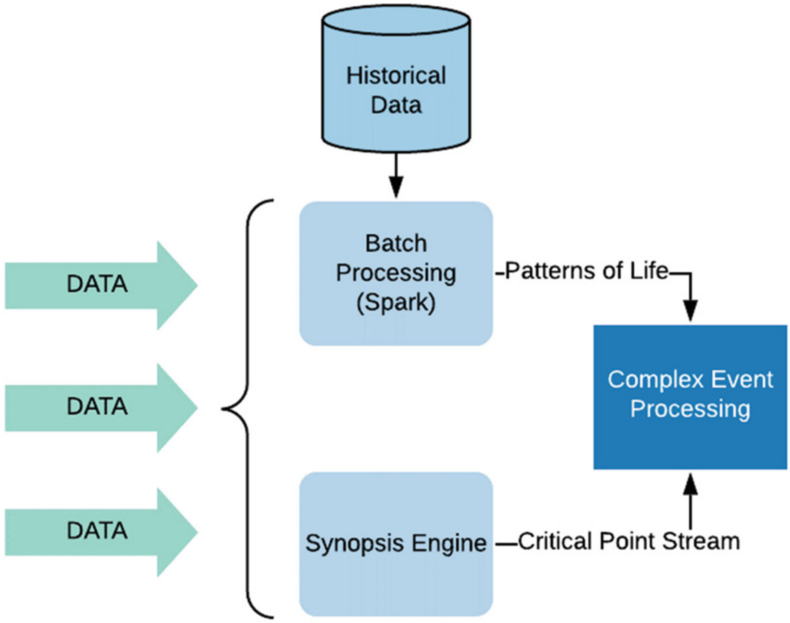

Fig. 9.10 MarineTraffic anomaly detection service architecture

Terrestrial AIS is combined with Satellite AIS in order to support almost global vessel monitoring.

On top of the vessel tracking services, MarineTraffic has deployed an anomaly detection service, which is based on a modified Lambda architecture [11]. This scheme allows the decoupling of batch processing, performed upon historical data, and online streaming analysis, which typically exploits the knowledge extracted from the batch processing (see Fig. 9.10).

The speed layer involves the trajectory synopsis generation engine as described in the previous section.

The batch layer performs the analysis of historical positional data of vessels and extracts the so-called Patterns of Life, that is, ‘normal’ maritime activity [2]. Batch processing is a long-running process which takes several hours to complete. Once completed, the Patterns of Life are fed into the online layer in order to accommodate detection of vessel anomalies in real time. Upon detection, those incidents are displayed to the end user. Furthermore, historical data are sent from this layer back to the batch layer at specific time intervals, defined from the seasonality of the data, thus replacing previously constructed patterns with new ones.

The online component is based on the actor model, specifically the Akka framework, for concurrency and event-driven processing. Actors are versatile, lightweight objects that have a state, communicate with each other and process the messages sent to them sequentially. By designing different types of actors and flows of information between them, we can create quite complex topologies, such

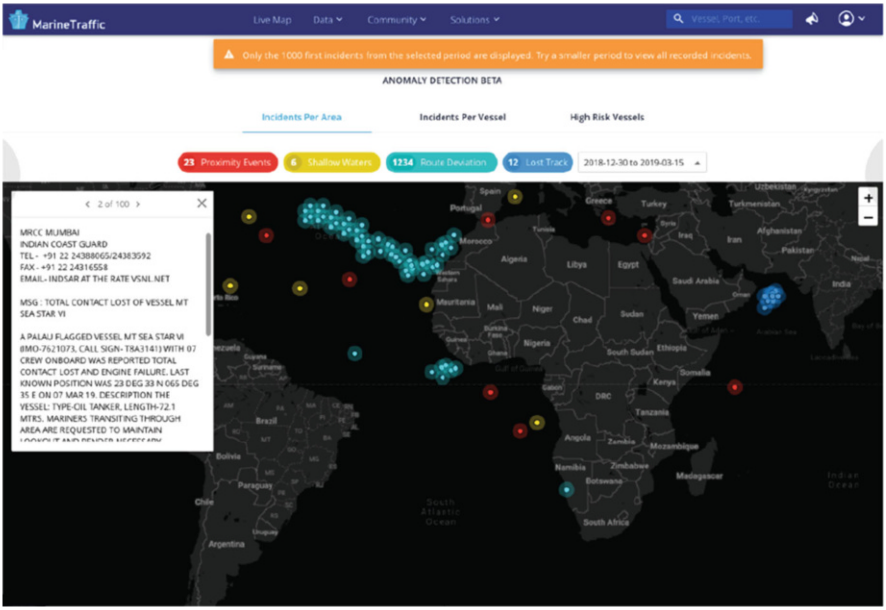

Fig. 9.11 Area's incidents page: the incidents/events occurring in an area are displayed on the map. Red dots, for example, display potential 'rendez-vous' incidents

as that displayed in Fig. 9.3, where each actor is responsible for recognising a maritime event and may be implemented in (the Scala version of) RTEC. Our current implementation includes 'ship actors' and 'cell actors'. The former type of actor receives the stream of AIS messages of a particular ship and detects propositional events concerning that ship, such as 'high speed near coast' (see Sect. 9.3.2.1) and 'drifting' (see Sect. 9.3.2.3). The latter type of actor detects relational events, where data from different ships within a grid cell need to be compared, such as tugging (Sect. 9.3.2.4) and potential 'rendez-vous' (Sect. 9.3.2.5). In the occasion of a recognition of a maritime event, users registered in the anomaly detection service are automatically alerted via email, or through the graphical user interface (see Figs. 9.11 and 9.12). The main goal of the design of the user interface was to reduce fatigue and the cognitive overload of the operators when having to search through numerous surveillance datasets and alerts. Towards this, the entire structure and format of the produced maritime situational awareness picture follows an interactive goal-driven approach.

An Example of a Real Case *SBI Jaguar* is a vessel sailing under the flag of Marshall Islands. On the 28th of March 2018, as the vessel was heading out of the Western Scheldt, a cylinder of the steering gear sheared off and the vessel was not under command. Then, *SBI Jaguar* ran aground on a sandbar in line with Perkpolder, and seven tugs were engaged in the vessel's re-floating operation. Subsequently, the vessel was towed to the Everingen's anchorage for an underwater inspection.

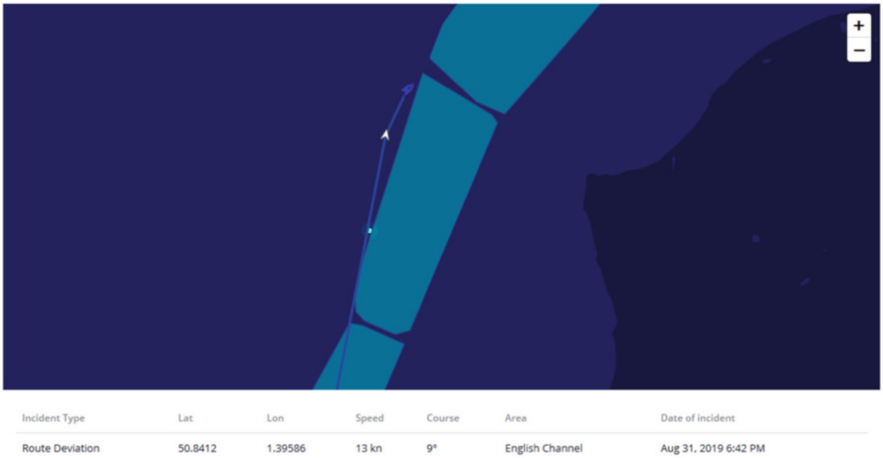

Fig. 9.12 Incident's details page. This screenshot shows a 'route deviation' incident/event (see also Fig. 9.3). Such an event is recognised when a ship is found travelling outside its expected route or speed patterns in a given area and time. The past track (in blue colour) and the normal route (in blue-green colour) are displayed on the map together with the vessel's speed, course over ground and the time the incident occurred

Fig. 9.13 The *SBI Jaguar* grounding

Figure 9.13 showcases that the vessel was sailing within its 'safe path', displayed by the polygon with turquoise colour, when the cylinder sheared off. At that point, the vessel started drifting having multiple changes of course over ground, which was detected by our service.

9.5 Summary & Further Work

We presented our on-going efforts towards integrating two state-of-the-art stream reasoning technologies for maritime situational awareness. A formal, computational framework has been re-implemented in the Scala programming language to allow for the interaction with the distributed implementation of MarineTraffic, thus paving the way for the real-time recognition of a wide variety of maritime events.

There are several directions for current and further work. First, we are currently evaluating the performance of the integrated system on real-world and synthetic datasets. (Results on the individual technologies have been already documented and indicate their capability for real-time performance [8, 16].) Second, we are developing online machine learning techniques for continuously refining maritime patterns given new streaming data [10, 12]. Finally, we are integrating satellite images with position signals and geographical information for a more complete account of maritime situational awareness.

Acknowledgments This work was supported by the datACRON and the INFORE projects, which have received funding from the European Union's Horizon 2020 research and innovation programme, under grant agreements No 687591 and No 825070.

References

1. Artikis, A., Sergot, M.J., Paliouras, G.: An event calculus for event recognition. IEEE Trans. Knowl. Data Eng. **27**(4), 895–908 (2015)
2. Chatzikokolakis, K., Zissis, D., Vodas, M., Spiliopoulos, G., Kontopoulos, I.: A distributed lightning fast maritime anomaly detection service. In: OCEANS 2019 - Marseille, June 2019, pp. 1–8 (2019)
3. Clark, K.L.: Negation as failure. In: Gallaire, H., Minker, J. (eds.) Logic and Databases, pp. 293–322. Plenum Press, New York (1978)
4. FAO. VMS for fishery vessels. http://www.fao.org/fishery/topic/18103/en. Accessed 15 May 2019
5. Greidanus, H., Alvarez, M., Santamaria, C., Thoorens, F.-X., Kourti, N., Argentieri, P.: The sumo ship detector algorithm for satellite radar images. Remote Sens. **9**(3), 246 (2017)
6. IMO. Technical characteristics for an automatic identification system using time division multiple access in the vhf maritime mobile frequency band. Tech. rep., ITU (2017)
7. IMO. Long-range identification and tracking system. Tech. rep., IMO (2018)
8. Improving maritime situational awareness through big data analytics, machine learning and artificial intelligence. Anomaly detection white paper, MarineTraffic Research (2019). https://www.marinetraffic.com/research/publication/anomaly-detection-white-paper/
9. Jousselme, A.-L., Ray, C., Camossi, E., Hadzagic, M., Claramunt, C., Bryan, K., Reardon, E., Ilteris, M.: Maritime use case description, h2020 datACRON project deliverable d5.1. http://datacron-project.eu/ (2016)
10. Katzouris, N., Artikis, A., Paliouras, G.: Online learning of event definitions. Theory Pract. Log. Program. **16**(5–6), 817–833 (2016)
11. Marz, N., Warren, J.: Big Data: Principles and Best Practices of Scalable Realtime Data Systems, 1st edn. Manning Publications Co., Shelter Island (2015)

12. Michelioudakis, E., Artikis, A., Paliouras, G.: Semi-supervised online structure learning for composite event recognition. Mach. Learn. **108**(7) , 1085–1110 (2019)
13. Millefiori, L.M., Zissis, D., Cazzanti, L., Arcieri, G.: A distributed approach to estimating sea port operational regions from lots of AIS data. In: 2016 IEEE International Conference on Big Data, BigData 2016, Washington DC, 5–8 December 2016, pp. 1627–1632 (2016)
14. Mills, C.M., Townsend, S.E., Jennings, S., Eastwood, P.D., Houghton, C.A.: Estimating high resolution trawl fishing effort from satellite-based vessel monitoring system data. ICES J. Mar. Sci. **64**(2), 248–255 (2007)
15. Patroumpas, K., Alevizos, E., Artikis, A., Vodas, M., Pelekis, N., Theodoridis, Y.: Online event recognition from moving vessel trajectories. GeoInformatica **21**(2), 389–427 (2017)
16. Pitsikalis, M., Artikis, A., Dreo, R., Ray, C., Camossi, E., Jousselme, A.: Composite event recognition for maritime monitoring. In: Proceedings of the 13th ACM International Conference on Distributed and Event-based Systems, DEBS 2019, Darmstadt, 24–28 June 2019, pp. 163–174 (2019)
17. Santipantakis, G.M., Vlachou, A., Doulkeridis, C., Artikis, A., Kontopoulos, I., Vouros, G.A.: A stream reasoning system for maritime monitoring. In: 25th International Symposium on Temporal Representation and Reasoning, TIME 2018, Warsaw, 15–17 October 2018, pp. 20:1–20:17 (2018)
18. Terroso-Saenz, F., Valdes-Vela, M., Skarmeta-Gomez, A.F.: A complex event processing approach to detect abnormal behaviours in the marine environment. Inf. Syst. Front. **18**(4), 765–780 (2016)
19. van Laere, J., Nilsson, M.: Evaluation of a workshop to capture knowledge from subject matter experts in maritime surveillance. In: Proceedings of FUSION, pp. 171–178 (2009)
20. Vouros, G.A., Vlachou, A., Santipantakis, G.M., Doulkeridis, C., Pelekis, N., Georgiou, H.V., Theodoridis, Y., Patroumpas, K., Alevizos, E., Artikis, A., Claramunt, C., Ray, C., Scarlatti, D., Fuchs, G., Andrienko, G.L., Andrienko, N.V., Mock, M., Camossi, E., Jousselme, A., Garcia, J.M.C.: Big data analytics for time critical mobility forecasting: recent progress and research challenges. In: Proceedings of the 21th International Conference on Extending Database Technology, EDBT 2018, Vienna, 26–29 March 2018, pp. 612–623 (2018)
21. Vouros, G.A., Vlachou, A., Santipantakis, G.M., Doulkeridis, C., Pelekis, N., Georgiou, H.V., Theodoridis, Y., Patroumpas, K., Alevizos, E., Artikis, A., Fuchs, G., Mock, M., Andrienko, G.L., Andrienko, N.V., Claramunt, C., Ray, C., Camossi, E., Jousselme, A.: Increasing maritime situation awareness via trajectory detection, enrichment and recognition of events. In: Web and Wireless Geographical Information Systems - Proceedings of 16th International Symposium, W2GIS 2018, A Coruña, 21–22 May 2018, pp. 130–140 (2018)