18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

A framework for dynamic context exploitation

Lauro Snidaro*, Lubos Vaci*, Jesus García†, Enrique D. Martí†,
Anne-Laure Jousselme‡, Karna Bryan‡, Domenico D. Bloisi§, Daniele Nardi§
*Dept. of Mathematics and Computer Science, University of Udine, Italy
Email: {lauro.snidaro, lubos.vaci}@uniud.it
†Dept. of Computer Science and Engineering, Universidad Carlos III de Madrid, Spain
Email: {jgherrer, emarti}@inf.uc3m.es
‡STO NATO, Centre for Maritime Research and Experimentation
Email: {Anne-Laure.Jousselme, Karna.Bryan}@cmre.nato.int
§Dept. of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy
Email: {bloisi, nardi}@dis.uniroma1.it

***Abstract*—While the benefits of exploiting Contextual Information (CI) are starting being recognized by the Information Fusion (IF) community [1], most current approaches for CI inclusion lead to stove-piped solutions that hardly scale or adapt to new input or situations. This paper makes a step in the direction of better CI exploitation by presenting some results of an international collaboration investigating the role of CI in IF and proposing an adaptive framework that dynamically takes into consideration CI to better support mission goals.**

In particular, we discuss some architecture concepts to be considered in the development of fusion systems including CI and we present how context can be dynamically exploited at different levels of a fusion engine. The concepts are illustrated in a maritime use-case.

I. Introduction

By surveying recent proceedings on contex in IF [1], [2], [3], three important conclusions could be made about context information (CI) exploitation. First, domain knowledge is in a vast number of cases tailored for application driven solutions of limited scalability and adaptability. Second, CI is not given the same level of importance throughout the levels of fusion, which reduces system performance. Third, frameworks which will be able to capture the nature of the context regardless of the target application are merely not existent. Furthermore, it seems that nowadays context aware systems (CAS) do not consider the fact that context is typically of dynamic nature. That is, a context variable may be latent, but it could be discovered through an inference process or it could be dependent on the user's and target's goals [1].

Llinas in [4] surveyed available frameworks for IF over the last decade. Based on his findings one should consider: a) graphs/network methods for creating contextual relations between events and entities; b) Common Referencing (CR), Data Association (DA), and State Estimation (SE) as basic functionalities of all fusion nodes; c) a Resource Management module (RM) to be coupled to the fusion engine in order to promote adaptation. Frameworks are expected to accommodate hard and soft information as well. From these premises, arguably the greatest weakness of current frameworks lies in their inability to provide adaptive feedback and to dynamically control the fusion process.

Steinberg and Bowman envisioned adaptability issues in [5], by introducing the concept of adaptive context discovery and exploitation. Their proposal is to seek, discover, select and fuse CI, modeled as context variables, as a part of goal-driven decision process e.g. through problem variables. Engineering implications posed by adaptive context discovery and exploitation were addressed subsequently in [6], and [7] and led to the development of the Data Fusion and Resource Management (DF & RM) Dual Node (DNN) architecture [5]. DF & RM DNN allows any decision process to be completely characterized in terms of IF and RM processes. The architecture has proven to be particularly useful in the design and evaluation of large, complex decision systems.

It is therefore particularly apparent the importance of adaptability in presence of CI that can be very transient depending on the current situation and target's and mission goals [1]. The weaknesses of current approaches are therefore stimulating the efforts for finding a truly adaptive CAS architecture in order to improve the performance of fusion processes.

In his work [4], Llinas goes beyond the survey and sets a stepping stone for further CAS development. The architecture he propose further develops ideas originated from Bowman and Steinberg [8], and from his own work [9], along with the already mentioned suggestions originated from the survey [4]. The design aspects of this architecture will be explained and expanded within the body of this paper.

This paper presents some results of an international collaboration investigating the role of CI in IF and proposing an adaptive framework that dynamically takes into consideration CI to better support mission goals. Before introducing the architecture here proposed (Section III), some terminology and fundamental concepts need to be recalled in the following section. A maritime use-case (Section IV) has been used to illustrate some functionalities of the designed architecture.

II. Fundamentals

Finding relevant CI is not self-evident and often involves a complex integration of IF with planning, abductive logic and control functions. Contextual reasoning is therefore seen as an inference process, where desired information i.e. problem variables can be in some sense enhanced (e.g., reducing uncertainty, augmenting accuracy) by CI. As of now, no unified framework for designing such context aware system exist, but one might consider concepts for *a priori* and *a posteriori* CI exploration respectively as a good reference [4]. We provide in the rest of this section some definitions of the key concepts used in this paper.

A. Definitions

Context: Context is understood as information that surrounds an element of interest, whose knowledge may help understanding the (estimated) situation and also in reacting to that situation [1]. As pointed out by Steinberg and Rogova [10], context can be used in IF to:

- Refine ambiguous estimates
- Explain observations
- Constrain processing, whether in cueing or tipping-off or in managing fusion or management processes.

Architecture: Structure useful for creating solutions to a problem, which describes the parts composing a solution and how they are organized and related. Architectures can focus on different organizational aspects including physical/processes distribution and topology.

Framework: "A conceptual structure intended to serve as a support or guide for the building of something that expands the structure into something useful" [4]. A framework tailored for a specific domain (e.g., IF) may include specific components fitting a broad range of applications in that domain.

Middleware: Software layer placed on top of another component. It provides higher level, domain-specific functionalities that improve the usability of the base component by services, applications and libraries.

Fusion Node: Abstraction of a generic fusion process that can be thought as composed by four consecutive steps (Uncertainty Characterization, Common Referencing, Data Association and State Estimation). It defines an interface for exchanging information (input and output) and managing its internal state and configuration.

Problem Space Characterization: The description of a generic problem (e.g., tracking) as an observable set of variables that need to be known, and how they are related. With this information, an intelligent algorithm manager can select from a repository the best algorithms that solve a fusion problem.

It can be noticed that these definitions are very generic and may resemble human judgment to integrate context knowledge in evaluating situations. For this reason, the aim of this proposal is highlighting the separation between context inputs and information sources from an architectural approach, avoiding particular solutions where context representation and exploitation is dependent on the application.

B. Context Adaptive Architectures

Ideas introduced by Llinas in [4] and Gomez-Romero et al. [11] established the basis for context-aware architectural designs. In their work, CI can be fully static or dynamic, possibly changing along the same timeline as the situation. Furthermore, authors argue that full characterization and specification of CI may not be able to be known at system/algorithm design time. Therefore, an "a priori" framework, that attempts to account for the effects on situational estimation of that CI that is known at design time, was introduced. Llinas et. al. also consider that CI may, like observational data, have errors and inconsistencies itself. Accommodation of these errors in data fusion processes leads to development of hybrid algorithms for "a posteriori" context exploitation. "A posteriori" in comparison to "a priori" includes checks of the consistency for a current situational hypothesis with the newly discovered CI. Both architectures assume the existence of a "middleware" layer which will be not only able to sample CI data and shape it into a suitable form for fusion processes, but also discover new CI. Our vision on how to realize such a middleware is presented in the next sections.

III. Architecture Design

This section describes the proposed architecture to integrate context sources in Information Fusion (IF) processes in a general way, so that any fusion system in which contextual knowledge is available can be developed following this architecture. The approach does not make assumptions or puts restrictions about specific fusion processes or information and context sources, but it will be defined at an abstract level, so that specific algorithms and applications can be developed based on the proposed architecture. In the first place, the types of context sources are commented, and a general mechanism to access context from fusion processes is proposed, following a middleware paradigm. Then, the adaptive IF framework is explained. The key idea is the exploitation of context knowledge to adapt the IF processes in order to optimize their performance.

A. Context Sources

A fusion system may access a number of different sources of contextual knowledge depending on the specific domain. In many applications, it is available in static repositories such as maps, GIS databases, representations of roads, channels, bridges, etc.; in other cases, context comes through dynamic data, such as meteorological conditions. In this case, we talk about context variables, implying the need of context access and update processes running in parallel with the core fusion processes. Finally, sometimes CI cannot be observed directly, and only indirectly deduced from other sources (inferred context). In addition, we can distinguish physical and logical context. In the first case, we will have physical descriptions (like GIS files) or variables (like meteorological phenomena) which are measurable objectively. In the case of logical knowledge (such as entities engaged in a coordinated trajectory, traffic regulations, mission goals, etc.), context can come from knowledge, human reports, learned from data or result from indirect inference processes based on other pieces of information. This characterization of context sources is illustrated in Figure 1.

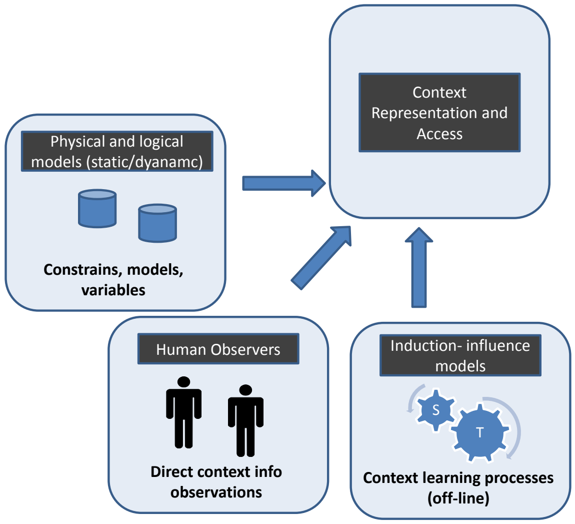

Fig. 1. Context source types

Therefore, contextual sources can classified in terms of the nature and way of accessing available information:

- Physical and logical structures:
  - Static datasets with information: roads, GIS databases, terrain characterization (navigation), urban environment, procedural information, normative, etc. In the maritime case, navigation routes or stationary areas are examples of context data sets, and some times they can be learned from historical data, as in the case of patterns of life reflecting the real behaviour of entities of interest.
  - Contextual variables such as physical fields: weather, wind, sea state, clouds, etc. These variables are distribution of magnitudes, changing in space and time
- Observed relations. Dynamic reports, human messages, and other documents represent the explicit input to the fusion process about situation (normal, labor day, anomaly, emergency, etc.), time of the day or week (working, meeting, etc.). These variables usually take discrete values indicating different contexts, coming from direct observation. The instantiated relationships are input to the system as context in some way, such as a human observation directly input to system. In the maritime case the geopolitical situation can be an example of dynamic observed relation.
- Inferred relations. Context can be deducted as dynamic relationships. A possibility is employing an automatic inference process, which may lead to the idea of a parallel representation of context process with its own processes and sources available.

B. Middleware

A way to systematically address advanced and generic context-based IF design deals with a context access and management system, in charge of providing useful context information about the entities as a transversal independent module. As mentioned, context services supporting fusion processes can be very heterogeneous, including, for example, access to reference databases, meteorological information, image repositories, GIS systems, texts, Internet, etc.

Accessing such heterogeneous information represents a challenge. The middleware approach can alleviate this problem by placing a component between context data and its consumers. This solution is a popular choice in context-aware computing applications, as analyzed in the survey [12].

IF processes access contextual resources through the interface exposed by the middleware. So, the context middleware acts as a transversal independent module in charge of deciding which context information is relevant, as illustrated in Figure 2

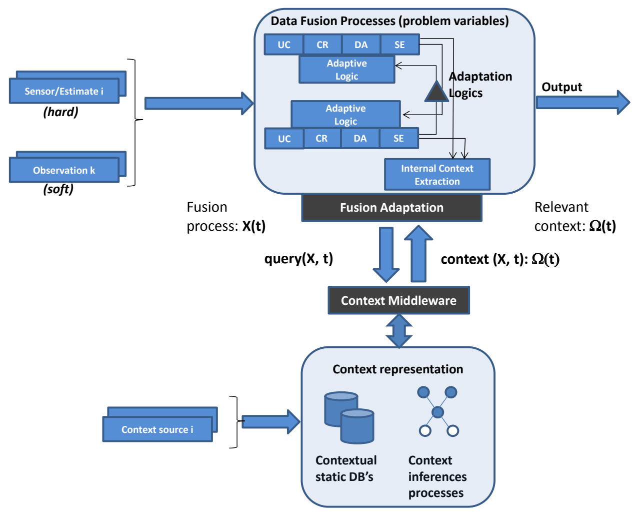

Fig. 2. Context middleware mechanism

The basic mechanism proposed follows a query-response process: the middleware returns the selected relevant context information from the available sources, according to the values inferred and hypotheses raised by fusion processes. Two basic elements can be identified in both sides:

- At the context side, the middleware is responsible for collecting, updating and making context knowledge usable by fusion processes.
- At the fusion side, the fusion adaptation logic uses the contextual inputs, so all processes and modules need to be described in terms of context input and interconnections to apply the adaptation.

In Figure 2, and also in the architecture presented in the next subsection, all IF processes are abstracted as nodes consisting in four main basic functions applied to the data. This abstraction is taken from [4], but including Uncertainty Characterization as part of the fusion process. In general, any fusion node accepts either sensor data from some input source or an estimate (fused or otherwise formed) from some prior processing node. In this characterization, processing operations involve four basic functions:

- **Uncertainty Characterization (UC)**: uncertainty associated to the information provided by the source, exploiting available models and related information.
- **Common Referencing (CR)**: normalization operations, such as coordinate or units transformations, to align data from information sources to be fused.
- **Data Association (DA)**: the multiple inputs (estimates or measurements) are examined in order to determine which (hypothetical) entity that the system believes to exist they are associated to or come from.
- **State Estimation (SE)**: computation of attributes (e.g., kinematic properties, classification attributes such as color, identity, inferred relationships, etc), exploiting the associated data together with prediction models in estimation/inference processes.

So, context middleware is responsible for providing "usable" context:

- Relevance: search for relevant pieces of context;
- It must provide up-to-date context. This means that it must integrate on-line information appropriate and potentially useful for the fusion processes;
- Granularity: it implies adaptation to the needs of fusion algorithm. For instance, in the access to wind representation, it can be 2D but needed 3D. Some aggregation or interpolation may be required to adapt the scales at both sides;
- Characterize the uncertainty in the contextual information provided, considering both the intrinsic uncertainty in contextual information and that propagated due to uncertain in query (for instance uncertainty in the location to index spatial context).

The operations to be done by the context middleware services are indicated below:

- Regarding search of applicable context to the fusion query:
  - Search of context relevant to the situation: physical (roads, bridges, channels, etc.), operational rules, etc.
  - Compatibility: validate the collected information as appropriate for query and check its compatibility (map, number of objects, etc.). In some cases, context maybe is not applicable (off-road, operational rules not met, etc.)
- Regarding transformation and normalization in the context response:
  - Context correlation and alignment with fusion process. This is especially relevant for use of real-time dynamic contextual sources, i.e. meteorological services;
  - Spatial alignment: fundamental for efficiency: search with appropriate representation and algorithms (maps, GIS, roads, etc.);
  - Time alignment (prediction functions): necessary when context is dynamic: simple temporal indexing, extrapolation models, etc.

With respect to context relevance, as commented in [10], a big challenge is determining the selection of context variables. In general, such selection should be based on previous knowledge of relations among context variables and problem variables. A possibility could be the development of an ontology based on relevancy of contextual variables to problem variables and their consistency. A context variable can be called relevant to a set of problem variables defining the reference items and relations between them, if the values of these problem variables change with the value of the context variable under consideration. Another criterion for determining a particular context as relevant may be the increase in information as the result of utilizing that context variable for estimation and/or inference. Finally, the problem of selecting context variables is more complex since relevance is often time-variable. Situations of interest are often dynamic, such that the availability of any sought data may also be time-variable. Even the mission-driven information needs and fusion processes can be also dynamic, making the utility of information given by context pieces also time-variable. Therefore, the middleware is proposed as an approach to generalize the context access and exploitation by fusion processes, organized as a set of operations done over the information available in different sources. The context middleware manager is responsible for searching and providing the relevant and updated information in the expected format and scale, considering the needs and requirements of the fusion node, so that fusion operations can take into account the context, independently of the specific strategy adopted. The service-oriented architecture is the key to develop a general perspective in the design and avoid particular solutions depending on the specific types and nature of the contextual sources available.

C. Architecture

The adaptive fusion architecture presented in this section is depicted in Figure 3, as an extension of [4]. Raw input data, covering both hard (electronic, physics-based) sensors and soft (human observers) sources, undergo detection, semantic labeling, and flow control composite functions. Once the best-qualified detections have been achieved, there is the question of assigning them to the various Fusion Nodes to be processed and generate the desired outputs.

The key to keep interaction with the contextual sources, through the middleware interface presented in previous subsection, is a function module called Problem Space Characterization below the detection operations. To adaptively manage a system with a library of alternative algorithms that address a generically-common problem space (e.g., object tracking problems), knowledge of the performance bounds of any library algorithm in terms of an observable set of parameters needs to be known. With such knowledge, an intelligent algorithm manager (part of the InterNodal Adaptive Logic) can terminate and invoke the best algorithm for the current

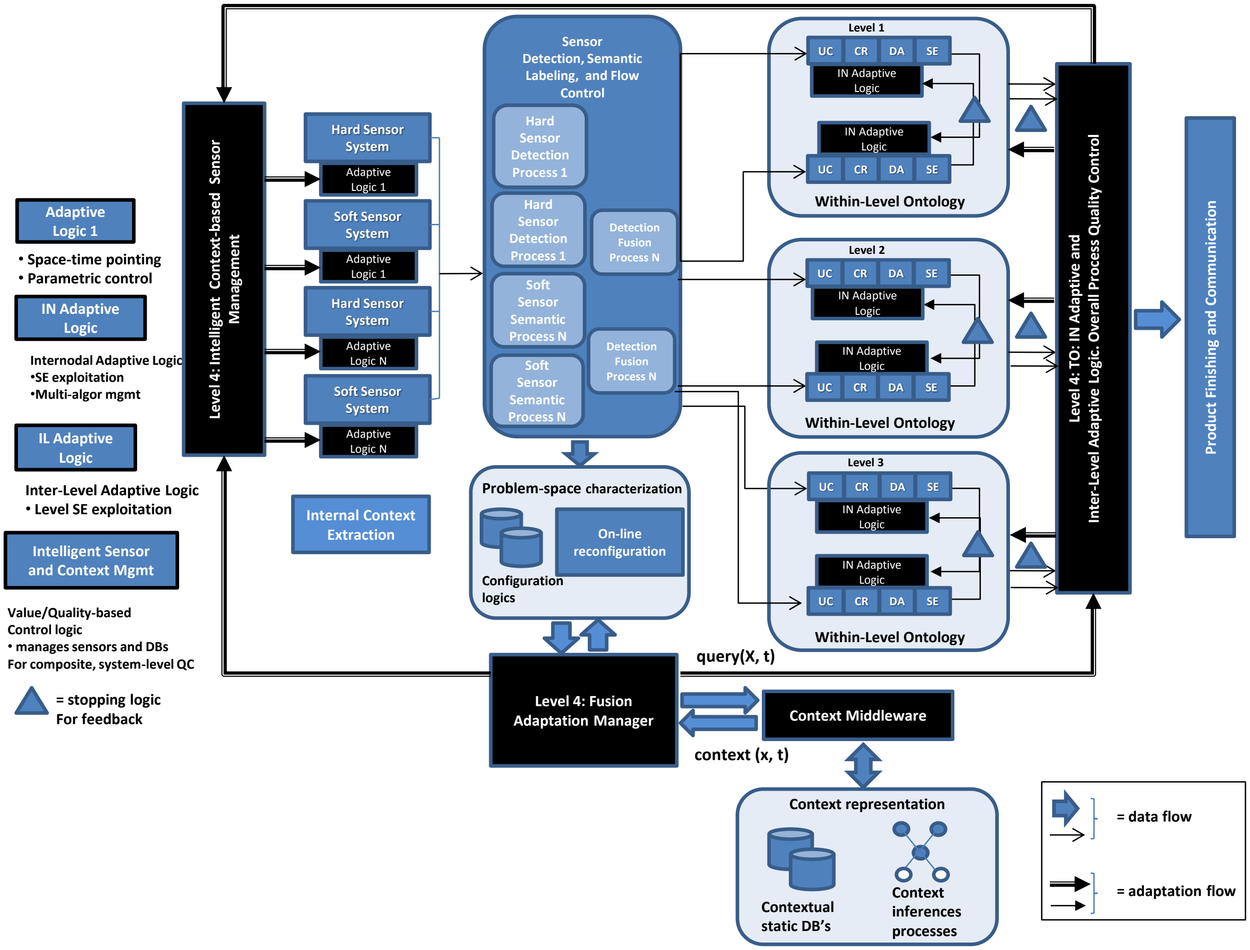

Fig. 3. Proposed adaptive Information Fusion framework

problem-space condition. An important point here is that the problems-space complexity parameters need to be observable by the system sensor set. Besides, we may distinguish a static configuration logic, describing all problem-space variable and inter-relations, and the possibility of dynamic adaptation. A typical example is the set of categories in a classification problem, which may change dynamically accordingly to the operative conditions or available context. This knowledge may also be contextually-dependent, so we have CI also feeding this knowledge base and control logic. The context middleware presented in previous subsection is in charge of providing the appropriate context pieces accordingly to the fusion variables state. This context is delivered by the adaptation manager to the different adaptive processes defined along the architecture, including the specific processes at the sources, the functions composing each individual fusion process (IN Adaptive logic boxes) and the inter-level processes, depending on the type of solution developed.

By definition, all the adaptation processes (highlighted in black in Figure 3) are part of JDL Level 4, which is one of the basic goals of the architecture: exploiting the context in order to refine and adapt the different fusion processes (including data sources). Feedback as adaptation is a fundamental aspect: the framework should show adaptive behavior such as inter-nodal feedback to allow (or perhaps require) that the Nodes share and exploit information if possible. One can see this in traditional Level 1 fusion for tracking and identification usually done in two separate Fusion Nodes; kinematics are of course helpful for identification, and identification is helpful for example to know an objects feasible dynamic motion. In turn, an adaptive Inter-Level feedback process is also shown, allowing situational estimates to feedback their estimates to other levels; an example of this would be a situational estimate that would suggest that maneuvering behaviors could be expected, informing Level 1 object tracking logic to open the tracking gates and capture the diverging measurements occurring upon the maneuver, i.e., as a maneuver-anticipation strategy instead of the (generally too-late) post-maneuver detection strategies often employed in tracking systems. As already mentioned in [4], all control loops need to define stopping criteria that terminate the otherwise-endless looping; that requirement is shown by triangles in Figure 3.

IV. Maritime use case

In this section, an example of instantiation of the proposed architecture is provided within a maritime use case. This use case is part of a selection of other use cases developed at CMRE to emphasize maritime security challenges and facilitate the collaboration and integration of different communities [13]. We identify the elements of context possibly considered, driven by the user's needs to take the decision.

A. Contextual Information

Significant portions of the world population live in coastal areas, and many large cities directly border the water. The maritime environment is complex, directly connecting the world via its waterways, with relatively limited regulation and a mixture of traffic ranging from large container vessels to smaller fishing boats and pleasure craft. Coastal areas are vulnerable to threats arriving from the maritime environment, as was seen in the Mumbai hotel bombings in 2008[1]. Civil authorities are responsible for monitoring harbor areas and protecting ports and critical infrastructure from threats arriving via maritime routes. Generally, some form of surveillance will be in place for major port areas, with any suspicious activity monitored, according to current threat levels and typical types of activity in the port. In heightened levels of threat, all unauthorized vessels approaching the port would be detected and monitored, with an assessment made of its behavior and intent assessed in order to allow early intervention if required. Intelligent systems making of use data and information fusion technologies are certainly an asset for harbor protection (e.g., [14], [15], [16]) and as an example, the fusion architecture presented in Section III is instantiated within the following use case.

The scenario takes place in a port loosely based on the port of La Spezia (IT), due to the variety and complexity of its activities. Some *physical contextual* information directly related to harbour zones characteristics is available such as water depths, channels, restricted areas, fishing areas, borders, harbours (fishing, recreational, etc), shipping lanes, ferry lane, military and LNG (Liquid Natural Gas) anchorage areas. A fair degree of *Pattern of Life* (PoL) is known about the area from experience and automated traffic pattern extraction routines [17]. There is significant fishing in the area and fishing vessels' behavior and fishing areas is generally understood. There are also several regular smaller passenger vessels for local tourism and private yachts and small boats. Other large vessels' including cargo vessels, tankers, and cruise ships operate normally in the area. Large passenger ships are required to report their Estimated Time Of Arrival and AIS information to the port authorities but smaller vessels do not have formal reporting requirements.

In this scenario, it is peacetime, there is no specific terrorist threat, but we are still in a post 9/11 security environment with a risk of potential malicious acts, from a variety of motivations. There is also an increased resentment after a recent wave of illegal immigration caused by political and economical instability of neighbourhood regions. Thus, the *geopolitical context* is relatively quiet and the Harbour Protection Level (HPL) is set to ONE over a scale of three levels[2]. For the *environmental context*, the meteorological conditions are clam (the weather is clear, sunny, there is no fog, the sea state is at the lowest level) within the port.

B. Response event

The use case presented here is a civil harbor protection response where the national authorities have just alerted the local authorities of a possible recent or imminent Improvised Explosive Device (IED) drop within the port [13]. After the notification, the local security coordinator executes the pre-planned response to confirm or disconfirm the credibility of the threat, including actions such as: (1) Elevate HPL to level TWO, (2) Increase local security measures (e.g., divert traffic and classify all real-time small vessel traffic), (3) Notify the investigation team, (4) Request for the deployment of Autonomous Underwater Vehicles (UUVs), to check the sea bed within the port and clear the area. The investigation team will conduct historical analysis of the electronic media and data (radar, SAR imagery, video from a Pan-Tilt-Zoom (PTZ) camera, phone traffic, AIS messages exchange, twitter, etc) of the last hours, interview local witnesses (e.g., harbour pilots, local fishermen, etc), looking for any suspicious or abnormal event missed during routine surveillance. An event of interest may have been missed because of the surveillance team was unaware of the threat at that time. The UUVs will adapt their search path based on any finding of the investigation team (e.g., localisation of a suspicious activity).

The *user context* is defined by the user's needs to take his decision: Based on the information provided by the investigation team together with the UUVs team, the local security coordinator will decide whether the threat is real or not and then, to step up the level of security or to return to normal security posture respectively [18]. The local security coordinator evaluates the risk regarding the probability of the threat (was it a hoax or not), the vulnerabilities of the port (e.g., the LNG terminal, ferries, container terminal, etc) and the consequences of the event (e.g., loss of life, economical). Based on some prior intelligence information, the evaluation of the threat by the investigation team first focuses on small vessels (fishing boat, pleasure craft, etc). Immediately, real-time small vessel traffic is to be classified by type. Further, among other aspects of the investigation, the captured data from the previous 24 hours will be reviewed and revisited in the light of the new threat declaration to possibly detect any suspect behaviour from small vessels.

C. Instantiated fusion architecture

Table I provides exemplar tasks to support the local security coordinator across the different levels of the JDL model [9].

[1] http://en.wikipedia.org/wiki/2008_Mumbai_attacks

[2] http://www.portlandharbor.org/Marsec%20Levels.htm

In rows, are listed the JDL levels 1 to 3 (level 0 is not considered here) while the four main fusion functions of Uncertainty Characterization, Common Referencing, Association and State Estimation and Prediction (see Section III) are listed as columns. To emphasize that the refinement process (level 4) applies at each level, it is added as a last column. Problem variables (observational, decisional and contextual) are also mentioned for each level.

Let us denote by $\mathbf{x}$ a vector of measurements (or observations) about different attributes (e.g., position, speed, heading, length, type) provided by several sources such as the coastal radar, the PTZ camera, the SAR imagery or AIS if available. Let us also denote by $\mathcal{X}_A^{(s)}$ the domain of the variable associated with attribute A for a given source, distinguishing between possible different domains across the different sources.

Level 1*:* The *State Estimation* $\hat{T}$ of the type of the vessel corresponding to a suspicious track (i.e., a small vessel) is performed, based on the vector of measurements $\mathbf{x}$. As the type is a perennial property, no prediction is required. The *Association* assigns any new declaration or measurement from the sources to the suspicious track. The different sources report over different attributes (e.g., the `vessel_width` and/or `vessel_length` for the SAR analyst or for the camera analyst, the `vessel_length` for the radar) and over different domains $\mathcal{X}_T^{(s)}$: Fishing vessels vs cargo vessels vs tankers vs service ships for the SAR imagery analyst, specific types of fishing vessels for the camera analyst). The *Common Referencing* at this level aligns for instance (but not only) the different type scales to a common one, as being suggested by the user context driven by his mission goal. The *Uncertainty Characterization* identifies some uncertainty origins such as the source's reliability, or the measurements' likelihoods and transforms the uncertainty into a suitable mathematical model of a dedicated mathematical framework (addressed in [19]).

Level 2*:* The behaviour analysis of each detected small vessel aims at detecting any behaviour such as "Speed too high for the type of vessel", "Fishing pattern while not in a fishing area", "Loitering in the port area", "Rendezvous". The anomaly detection task can rely on several *State Estimations* for a further global State Estimation (e.g., Normal vs Abnormal). Anomaly detection essentially compares the estimated attributes at level 1 (`vessel_speed`, `vessel_heading`, `vessel_type`) to expected ones as represented by pre-defined patterns of life of routes or dedicated areas [17]. The *Common Referencing* aligns the spatial scales of the different sources (AIS, radar, SAR), regarding the `vessel_position`. In addition to the *Uncertainty Characterization* of contextual information (routes) in routes' representation (contextual knowledge), the UC at this level is essentially similar to UC at level 1 and some likelihood functions may be elicited from past AIS records. However, other dimensions such as the uncertainty derivation (objective vs subjective), may be characterized as well for a better interpretation of uncertainty representation by the user. The *Association* identifies any piece of information contributing to the task and being possibly related to the vessel's behaviour. For instance, an phone or radio call associated to the vessel may be used.

Level 3*:* In case of the detection of an abnormal behaviour, the impact is assessed involving some risk analysis elements such as the cost of (relevant vs non-relevant) intervention need to be considered. The *State Estimation and Prediction* is the classification of the vessel as the threat (i.e., the one dropping the IED) which considers both its behaviour and static information. The *Association* ensures that all ID statements from concern indeed the suspect vessel. The *Uncertainty Characterization* includes some aspects of threat assessment (probability of abnormal behaviour) from Level 2 but also the assessment of the vulnerability and cost of critical assets in the area, for a further risk assessment at the *Prediction* task. The *Common Referencing* aligns the identification statements of the different sources to the standard categories applied by the local Harbour Protection team.

Level 4*:* The refinement step influences each of the three above JDL levels, to adapt to some contextual change:

- Level 1 The classification is refined based on new user's needs: At a first instance, the local security coordinator was interested in distinguishing between small and large vessels as represented by $\mathcal{X}_T^{(\text{User1})}$. A finer assessment was then required to discriminate between different types of small fishing vessels and pleasure craft, as represented by $\mathcal{X}_T^{(\text{User2})}$;
- Level 2 the anomaly detectors' performance is directly impacted by the speed estimation. An updated meteorological information requires to adjust sensors' parameters for an updated assessment of `vessel_speed` and an improved anomaly detection (see Section IV-D below);
- Level 3 the path planning of the UUVs may be adapted and modified on the fly based on the past location of a suspect vessel.

D. Dynamic Parameter Adaptation

The system can exploit contextual information for adapting the sensor parameters. A possible way of performing the dynamic parameter adaptation is to establish a relationship between the context variables and the parameters of the sensors. Given the context variables in Table I, they can be represented as quadruple $< T, r, a, l >$, where T is the vessel type, r the expected route, a the designed area, and l the HPL. A set of different context instances can be obtained by combining their values: $< T, r, a, l > \rightarrow \{C_1, ..., C_n\}$. For example, in case of $< ferry - boat, to - Slickville, ferry - lane, TWO >$, the associated context C_i can be labeled as "ferry boat of 5 pm". Given a particular context, a set of parameters for the sensors can be established though a relationship $C_i \rightarrow \{p_1, ..., p_n\}$, where p_i is a single parameter of a sensor in the system. In the case of the ferry boat, the position of the PTZ camera can be set to point on the ferry lane, with a zoom level adequate to the estimated distance of the boat from the camera site. As another example, in the case of a possible threat coming from a small boat, radar parameters can be changed to be more

TABLE I
EXAMPLE OF FUSION NODE FUNCTIONS ACROSS THE JDL LEVEL FOR THE USE CASE.

	Variables			Fusion Node functions				
	Observation	Decision	Context	UC	CR	DA	SE	**Level 4 Process Refinement**
Level 1 Object Assessment	$vessel\_length$ $vessel\_width$	$vessel\_type$	$\mathcal{X}_T^{(\text{User1})}$	track split and merge	$\mathcal{X}_T^{(\text{SAR})} \leftrightarrow \mathcal{X}_T^{(\text{Rad})}$	$\mathbf{x} \mapsto$ Track	$\mathbf{x} \mapsto \hat{T} \in \mathcal{X}_T^{(\text{User1})}$	$\mathcal{X}_T^{(\text{User1})} \to \mathcal{X}_T^{(\text{User2})}$
Level 2 Situation Assessment	$vessel\_speed$ $vessel\_type$ $vessel\_length$	$vessel\_behaviour$	$route\_set$ $designated\_areas$ sea_state	route extraction	Grid alignment SAR, AIS, radar	$\mathbf{x} \mapsto V$	$V \mapsto \{Normal; Abnormal\}$	Adjustment of camera's parameters
Level 3 Impact Assessment	$vessel\_behaviour$ $vessel\_flag$	$vessel\_identity$	HPL	threat statistics and costs	$\mathbf{x} \mapsto$ Standard categories	$\mathbf{x} \mapsto V$	$\mathbf{x} \mapsto \hat{ID} \in$ Standard categories	Detailed intervention plan

sensitive for the detection and tracking of small vessels. The SAR imagery parameters can be adapted by estimating the speed of the suspect vessel.

In the same way, the parameters of the fusion nodes can be updated. For example, if the context variable tuple contains $fishing - area$ as designed area, the parameters of the vessel route analysis process can be set to account for non-linear trajectories, since fishing vessels are expected to perform circular trajectories.

This example illustrates the potential use of context information to adapt fusion processes in a maritime scenario. However, in order to implement the proposed framework some steps are needed to obtain a full functionality. First, the context middleware should access the available sources, represented in a convenient way in order to provide the relevant and updated context. Second, it is necessary to develop appropriate interfaces to access the real fusion processes and adapt their parameters based on available context inputs, and manage the adaptation flows from the context middleware to each data processing node.

V. CONCLUSIONS

This paper addresses several concepts and issues to be taken into account in developing a context-aware fusion system. We have discussed an architecture for dynamically exploiting context at different levels in a fusion engine. The solution adopts a middleware approach which provides a convenient way of designing an interface level between data/information sources and the fusion functions, brokering all relevant contextual data sources to the correct data sinks. The concept has been applied to a port protection use-case and will be further developed as part as an international collaboration investigating the role of CI in fusion systems.

ACKNOWLEDGMENT

This work was supported by ONRG Grant N62909-14-1-N061, and partly by project MINECO TEC 2011-28626-C02-02 and NATO Allied Command Transform (ACT).

REFERENCES

[1] L. Snidaro, J. García, and J. Llinas, "Context-based information fusion: a survey and discussion," *Information Fusion*, vol. 26, pp. 16–31, September 2015.

[2] L. Snidaro and J. García, "Context-based information fusion," *Information Fusion*, vol. 21, pp. 82–84, 2015, Guest Editorial.

[3] J. Llinas, L. Snidaro, J. García, and E. Blasch, "Context and fusion: Definitions, terminology," in *Context Enhanced Information Fusion*. Springer, ch. 1, (in press).

[4] J. Llinas, "A Survey and Analysis of Frameworks and Framework Issues for Information Fusion Applications," in *Hybrid Artificial Intelligence Systems*. Springer Berlin Heidelberg, 2010, vol. 6077, pp. 14–23.

[5] A. N. Steinberg and C. L. Bowman, "Adaptive context discovery and exploitation," in *16th International Conference on Information Fusion*, Isanbul, 2013, pp. 2004 – 2011.

[6] D. L. Hall and J. Llinas, *Multisensor Data Fusion, Second Edition - 2 Volume Set*. London: CRC Press, 2001.

[7] C. L. Bowman, "Process assessment and process management for intelligent data fusion & resource management systems," in *AIAA Space Conference & Exposion*, Pasadena, 2012.

[8] A. N. Steinberg, C. L. Bowman, and F. E. White, *Revisions to the JDL data fusion model*. NATO/IRIS, 1999, vol. 3719.

[9] J. Llinas, C. L. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White, *Revisiting the JDL Data Fusion Model II*, San Diego, 2004.

[10] A. Steinberg and G. Rogova, "Situation and context in data fusion and natural language understanding," in *11th International Conference on Information Fusion*. Cologne: IEEE, 2008.

[11] J. Gomez-Romero, J. Garcia, M. Kandefer, J. Llinas, J. M. Molina, M. A. Patricio, M. Prentice, and S. C. Shapiro, "Strategies and techniques for use and exploitation of Contextual Information in high-level fusion architectures," in *13th International Conference on Information Fusion*, Edinburg, 2010, pp. 1–8.

[12] K. E. Kjær, "A survey of context-aware middleware," pp. 148–155, Feb. 2007. [Online]. Available: http://dl.acm.org/citation.cfm?id=1332044.1332069

[13] A.-L. Jousselme, K. Bryan, and M. Ilteris, "Selection of use cases to support maritime situation awareness algorithmic development," NATO Centre for Maritime Research and Experimentation (CMRE), La Spezia (IT), Techincal Memorendum, 2015, in review.

[14] E. Shahbazian, G. Rogova, and M. J. DeWeert, "Findings of the NATO workshop on data fusion technologies for harbour protection," in *Proc. of SPIE, Photonics for Port and Harbor Security II*, vol. 6204, 2006.

[15] J. Garcia, J. Gomez-Romero, M. A. Patricio, J. M. Molina, and G. Rogova, "On the representation and exploitation of context knowledge in a harbor surveillance scenario," in *Proceedings of the 14th International Conference on Information Fusion*, Chicago, Illinois, USA, July 5-8 2011.

[16] J. Gómez-Romero, M. A. Serrano, J. Garcia, J. M. Molina, and G. Rogova, "Context-based multi-level information fusion for harbor surveillance," *Information Fusion*, vol. 21, pp. 173–186, January 2015.

[17] G. Pallotta, M. Vespe, and K. Bryan, "Vessel pattern knowledge discovery from AIS data - A framework for anomaly detection and route prediction," *Entropy*, vol. 15, no. 6, pp. 2218–2245, 2013.

[18] L. Snidaro, I. Visentini, and K. Bryan, "Fusing uncertain knowledge and evidence for maritime situational awareness via markov logic networks," *Information Fusion*, vol. 21, pp. 159–172, January 2015.

[19] A.-L. Jousselme and G. Pallotta, "Dissecting uncertainty-based fusion techniques for maritime anomaly detection," in *Information Fusion (FUSION), 2015 Proceedings of the 18th International Conference on*, 2015.