ROCZNIK NAUKOWY WYDZIAŁU ZARZĄDZANIA W CIECHANOWIE 1-4 (XI) 2017

Rafał Tkaczyk, Mateusz Bonecki, Szymon Bohdanowicz,
Maria Ganzha, Marcin Paprzycki

ZŁOŻONE PRZETWARZANIE ZDARZEŃ W SYSTEMACH CYBER-FIZYCZNYCH: PERSPEKTYWA PRAGMATYCZNA

[**słowa kluczowe**: projekty badawczo-rozwojowe, złożone przetwarzanie zdarzeń, systemy cyber-fizyczne]

Streszczenie

W artykule omówiono rozwiązania dotyczące złożonego przetwarzania zdarzeń (Complex Event Processing), zastosowane w dwóch projektach badawczo-rozwojowych (ACCUS i DEWI). Celem tej pracy jest porównanie założeń złożonego przetwarzania zdarzeń i systemów cyber-fizycznych (Cyber-Physical Systems), wraz z zastosowanymi w nich rozwiązaniami.

* * *

COMPLEX EVENT PROCESSING IN CYBER-PHYSICAL SYSTEMS: A PRAGMATIC PERSPECTIVE

[**keywords**: R&D projects, complex event processing, cyber-physical systems]

Abstract

This paper discusses complex event processing solutions developed in two European R&D projects (ACCUS and DEWI). The focus of this work is a comparison of assumptions of complex event processing and cyber-physical systems with solutions actually applied in them.

Introduction

Nowadays, using complex event processing (CEP) in cyber-physical systems (CPS) is a common practice [3]. It is used in domains such as: financial [4], building automation [5], e- or m-Health [6], and is often combined with common technologies, such as RFID [7], WSN [8], or SOA [9]. This paper is based on experiences gathered during two European R&D projects: Adaptive Cooperative Control in Urban (sub) Systems [11] (ACCUS, which ended in January 2016) and Dependable Embedded Wireless Infrastructure [12] (DEWI, project which is in the final stage, after the pilot presentation). The ACCUS project developed an Integration and Coordination Platform for distributed urban systems, and was focused on energy, mobility, and buildings. The DEWI project, on the other hand, focuses on use of intelligent wireless embedded systems in: aerospace/space industry, car production, smart buildings and railway systems. While these projects have many common requirements and goals, “small differences” in their approaches make their results very different. Thus, we use hands-on experience of the ACCUS and the DEWI projects to illustrate breadth of possible solutions in multi-domain cyber-physical systems, using dedicated complex event processing modules.

Abbreviations

This paper contains many abbreviations of popular and new technology-related terms. Here, we introduce all names (in alphabetical order), used in the paper, in order clearly define their content.

- ACCUS [11] – Adaptive Cooperative Control in Urban (sub) Systems, EU project described, in details, in section “*Scope and objectives of ACCUS project*”.
- ACCUS API – ACCUS Application Programming Interface, based on HTTP protocol and REST standard (using JSON format).
- ACS – Access Control System. A system that was developed and deployed in the DEWI project pilot.
- CCTV [19] – closed-circuit television.
- CEDE – Complex Event Detection Engine. A module in the ACCUS platform architecture described, in details, in section “*Scope and objectives of ACCUS project*”.

- CEP [1][20] – Complex Event Processing.
- CPS [2][3][21] – Cyber-Physical Systems.
- CSDB – City State Database (related with the ACCUS project). A central storage component in the ACCUS platform architecture described, in details, in section "*Scope and objectives of ACCUS project*".
- D2RQ Platform [16] – system for accessing relational databases as virtual, read-only RDF graphs.
- D2RServer [16] – tool for publishing relational databases on the Semantic Web.
- DEWI [12] – Dependable Embedded Wireless Infrastructure [12]. EU project, described, in details, in section "*Scope and objectives of DEWI project*".
- DEWI Bubble – group of nodes, gateways and users within a restricted network (described in section „Scope and objectives of DEWI project").
- ICP – Integration and Coordination Platform (related with the ACCUS project), described, in details, in section "*Scope and objectives of DEWI project*".
- IPS [10] – Indoor Positioning System. A system, developed for and deployed within the DEWI project pilot.
- JBoss Drools [22] – Business Rules Management System solution.
- JENA [14] – free and open source Java framework for building Semantic Web and Linked Data applications.
- JSON [23] – JavaScript Object Notation, a lightweight data-interchange format.
- JSON-LD [24] – lightweight Linked Data format, based on JSON, provides a way to help JSON data interoperate at the Web-scale.
- MVEL [17][25] – MVFLEX Expression Language – hybrid dynamic/statically typed, embeddable Expression Language and runtime for the Java Platform.
- OWL [26] – Web Ontology Language. A Semantic Web Language designed to represent rich and complex knowledge about things, groups of things, and relations between things.
- R&D (projects) [27] – Research and development. Activities in connection with corporate or governmental innovation.
- RDF [28] – Resource Description Framework. A standard model for data interchange on the Web.
- RFID [29] – Radio-frequency identification. Uses electromagnetic fields to automatically identify and track tags attached to objects.

- SOA [30] – Service-Oriented Architecture. A style of software design where services are provided to other components by application components, through a communication protocol over a network.
- SPARQL [31] – Protocol and RDF Query Language. Language to make queries in data represented in the RDF format.
- TRISTAR [32] – Intelligent Transportation System. It was created to optimize traffic management within Polish Tricity (Gdańsk, Sopot, Gdynia).
- WDA – Wireless Data Aggregator. A module in the DEWI Bubble architecture described, in details, in section "*Scope and objectives of DEWI project*".
- WSN [33] – Wireless Sensor Network.
- ZigBee [18] – Specification of data transmission protocols in mesh wireless networks (cluster tree).

Scope and objectives of the ACCUS project

The Adaptive Cooperative Control in Urban (sub) Systems (ACCUS) project developed an Integration and Coordination Platform (ICP) for distributed urban systems, dedicated to energy, mobility, or building automation systems. The architecture of the project implies the central role of the ICP, which provides, among others, data bus functionality. Urban subsystems are connected to the platform and expose data, which is used, for instance, as an input to event processing. They are integrated with the ICP through adapters, which parse and semantically map the retrieved information onto the ACCUS ontology of the urban domain. As a result, data gathered from distributed subsystems is available through a common, unified data model and stored in a central storage component (City State Database; CSDB). The CSDB consists of an SQL relational database and a Virtuoso triple store repository. Since ACCUS assumes central role of the CSDB data storage, atomic events available for further processing are sourced directly from it. The event processing service, Complex Event Detection Engine (CEDE), is subsystem-data-agnostic as it operates on gathered, centrally stored, and semantically organized data, accessible through an information-brokering service (Infobroker). The Infobroker is a single access point to enable information retrieval from all ACCUS data resources, i.e. databases, subsystems or sensor networks.

Atomic events, stored in the CSDB, originate from devices, sensor networks, subsystems, etc. ACCUS works with data feeds concerning weather, traffic congestion, noise, pollution, etc. Molecular (complex) events are produced by the CEDE, as a result of processing atomic events, and may be composed of entities originating from different subsystems. Therefore, CEDE is a cross-domain service, capable of enhanced situational awareness that cannot be built on the basis of any domain-specific subsystem alone.

To use semantic data processing, the ICP employs a D2RQ Platform and a D2RServer [16]. As a result, data stored in the relational (SQL) database can also be accessed using a semantic query language (SPARQL [31], in the case of the project). Specifically, from the point of view of the client, data is accessed as it was stored using RDF [28] (the D2RQ Platform supports read-only RDF graphs). Overall, both SQL and SPARQL queries are performed by the Infobroker.

It should be noted that CEDE does not facilitate strict real-time guarantees for event processing. It is designed for soft real-time applications, where missing deadlines do not interrupt critical processes. The CEDE component, responsible for event processing, is the Persistence Monitor, based on JBoss Drools [22]. Event occurrence is a consequence of rule-based mechanism, where events are created, deleted, or modified, when certain conditions are met. Event definitions consist of ontology units (e.g. road congestion) and their properties (e.g. congestion level, road location). To compose an event, logical expressions (e.g. *AND*, *OR*) are used to indicate relevant ontology units, and mathematical comparators (<, >, ==, !=) to indicate their properties. Complex event definitions may also include temporal constraints (e.g. a composite event occurs when atomic event A occurred within 15 minutes after event B) and event validity period, after which the Persistence Monitor checks if complex event conditions are still met. Definitions are written in the MVFLEX Expression Language (MVEL) [17]. Event distribution is covered by a dedicated CEDE component (Event Broadcaster) available through the ACCUS API as a REST web service. Event Broadcaster can be queried by event consumers (applications, services, subsystems). A list of events is returned in JSON format [23]. Event processing in ACCUS is outlined in Figure 1.

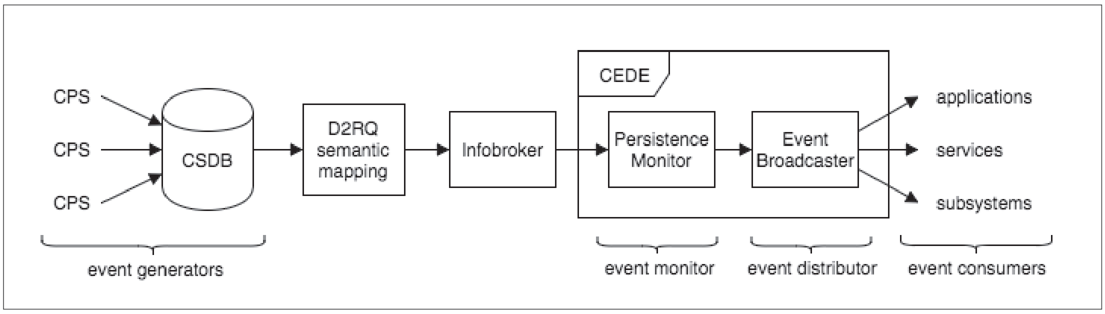

Figure 1. Complex Event Detection Engine as a CEP solution

The underlying assumption of the project is that processes, managed by respective systems, are intertwined, while the ICP architecture does not support data exchange and coordination of their capabilities. To this end, the ICP implements adaptive and cooperative control architecture, to optimize combined performance of urban subsystems and their processes. For example, air quality is a result of traffic, weather conditions (e.g. wind gusts), operations of combined urban heat and power plant, and requires middleware to exchange information between involved ICP subsystems and to coordinate control actions. Thanks to the ICP, it is possible to compose higher level functions, using services or data resources offered by constituent systems. In one of ACCUS applications, dedicated event detection service, within the ICP, was able to recognize complex events composed of atomic events sourced from: (i) traffic congestion data originating from the Intelligent Transportation System "TRISTAR" [32], and (ii) temperature measurements arriving from the Weather Monitoring system that aggregates weather readings from sensors distributed in the city. As a result, a coincidence of high traffic congestion (measured by TRISTAR) and subzero temperature (measured by the weather system) was identified at the ICP level as "difficult road conditions" and sent, as a message, to a "management cockpit" application used by the municipal services.

Scope and objectives of DEWI project

The Dependable Embedded Wireless Infrastructure (DEWI) project involves use of intelligent wireless embedded systems in: aerospace/space industry, car production, smart buildings and railways. Its main goal is to integrate nodes running, and cooperating, in the same environment, i.e. in a separate subnet – called DEWI Bubble (see Figure 2). A single Bubble handles one domain, e.g. a building. The purpose of such solution is to facilitate seamless cooperation of many systems (access control, positioning, emergency, security, lighting, CCTV, etc.) and devices (terminal, smoke sensor, camera, sensor, etc.), typically. instantiated in a single building.

Cooperation is based on two mechanisms. First, use of the Wireless Data Aggregator (WDA), a device with a complex functionality comprising: access control manager, area controller, etc. Moreover, the WDA is responsible for data fusion, aggregation and distribution. The WDA is one of the nodes in the net (it can be an independent device or an element of a system, e.g. the Access Control System). It enables direct communication between nodes (regardless of technology), since it supports multiple communication interfaces (wired and wireless, e.g. Wi-Fi, ZigBee [18], Bluetooth, etc.). The main reason for creation of the WDA is integration of different systems within a single "domain" (for instance a single building). The WDA deals with: (1) distribution of data, relevant from the point of view of different subsystems, (2) registration of nodes in the WSN network, as well as services they provide, (3) defining business rules associated with the gathered data, fulfillment of which is to trigger specific actions (e.g. alert detection), and (4) as quickly and efficiently as possible, obtaining information (e.g. in the Access Control System).

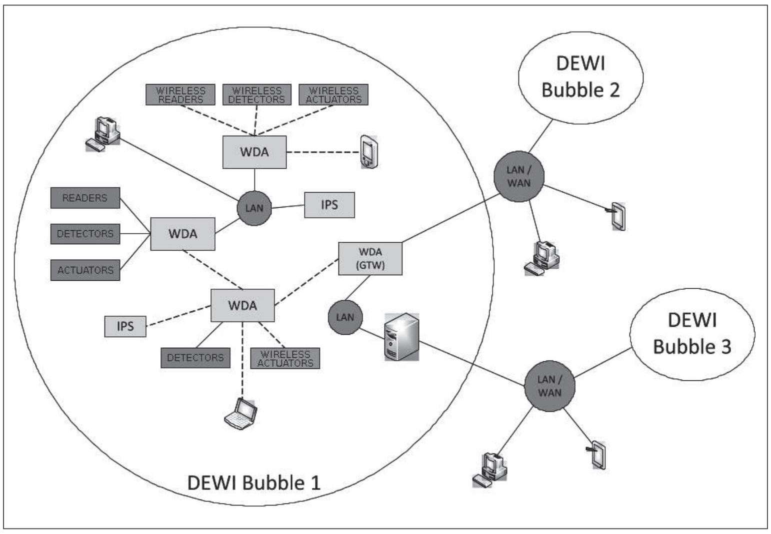

Figure 2. Topology of DEWI wireless sensor subnetworks (Bubbles)

The domain of the DEWI Bubble is formally captured in an ontology, on the basis of which the graph database (OrientDB [15], used in the project) structure is defined. This allows use of the WDA for data fusion. Specifically, data that is important from the point of view of other (cooperating) nodes, or of data analyzing modules (e.g. a Business Rule Engine) is gathered in this graph database. Data subscription module allows receiving the actual data on-the-fly. Note that in the DEWI, messages between nodes are in the JSON-LD format [24], because the overall data structure is based on an ontology and describes OWL [26] individuals.

An example of WDA use is a pilot deployment in which cooperation between (I) Access Control System (ACS), (ii) Indoor Positioning System (IPS) [11], and (iii) a prototype device mobile terminal takes place. The ACS, via the WDA, is obtaining information about positions of objects supported within the restricted area. Here, objects are entities like person, thing, device, equipped with a Jennic module (a ZigBee [18] wireless microcontroller module with a built-in antenna) or RFID [29] cards. Then, the IPS can obtain important information that affects the verification of results of applied algorithms (e.g. card read events, in ACS devices, can verify the estimated object position). Another scenario is to use a mobile terminal (with an embedded Jennic module) inside the building. With the WDA, the user can request his/her visit to a restricted area (using the ACS) and get a permission (or be refused). Moreover, based on the data from the ACS and the IPS, the terminal can locate the position of an object, which is sought by the user, e.g. person, thing. For instance, it can identify tool or device not belonging to the Bubble infrastructure, device belonging to the Bubble infrastructure, or a place (e.g. a specific room). The result of this search is not only the information about the position of the object, but also a map displayed on the screen, including a directions how to get to the "target".

The second approach to cooperation is a Big Data Architecture solution. For this purpose, two directly cooperating modules: (1) Data Fusion System, and (2) Context Aware And Reasoning Module were developed. It is a centralized approach, focused on gathering and processing large amounts of data in a single module. Data Fusion System enables gathering data from all WSN nodes and provides tools for processing. A detailed diagram of this architecture solution is shown in the Figure 3. Data are collected in the TDB storage [13] (a JENA module [14]), as RDF triples. This allows unification of data using ontologies representing the domain, and use of semantic data processing mechanisms, e.g. SPARQL queries. Prepared data is ready for processing by the Context Aware And Reasoning Module. It allows the execution of: (1) verification of complex conditions, (2) statistics, and (3) learning (as, nowadays, it is extremely important that the system can learn and apply knowledge, instead of using rigid patterns specified by its developer).

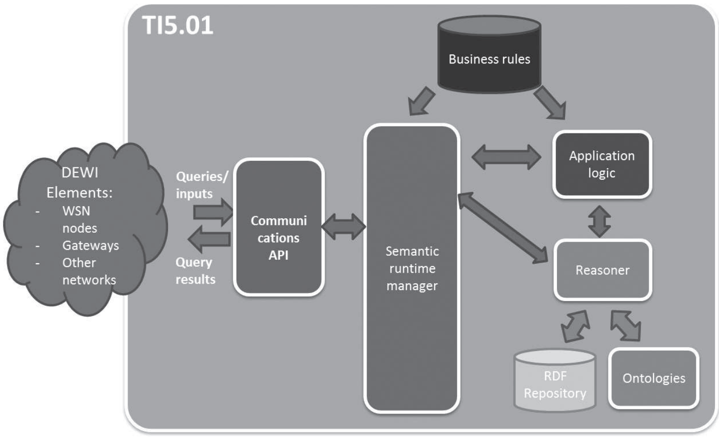

Figure 3. Data Fusion System components

Comparison – differences and similarities

ACCUS and DEWI projects have similar assumptions and handle similar domains. However, analyzing their implementation, it is easy to see that they result in different solutions. Two different approaches are also used within the DEWI project, responding to different requirements. Let us summarize key differences.

Architecture. In the ACCUS project, there is a Centralized platform, which uses the ICP to connect systems and applications. This project focuses on cooperation of high level entities: subsystems and application. The DEWI project, on the other hand, has two approaches: (1) centralized architecture, which uses the Data Fusion System module to collect and exchange data, or to analyzes in the Context Aware And Reasoning Module (CEP); and (2) partially centralized architecture, which uses the Wireless Data Aggregator (WDA) for cooperation of network nodes. Decentralization involves use of multiple WDA devices (theirs number depends on the number of nodes in network). These devices are autonomous, but they work together (cooperate), and (in the case of failure) can take tasks of the failed ones.

Communication. In ACCUS, since entities integrated by the ICP are high level structures, REST communication with JSON messages is used. In DEWI, the main communication style is also REST (with JSON). However, DEWI is intended to integrate high level (systems, applications, etc.) and low level nodes (devices, sensors, etc.). Therefore, multiple communication methods are allowed (and have been implemented).

Domains. In ACCUS, every system and application is from one, very general, domain: smart city. DEWI, in theory, works for any domain. In practice: aerospace/space industry, car manufacturing, smart buildings and railways, are being considered. Each domain is a separate DEWI Bubble, where each Bubble can cooperate with the others through gateways (here, note that the WDA can be also act as a gateway).

Main use cases. In ACCUS, the main goal is integration of systems and applications that run in a smart city environment. The aim of the project is to facilitate collaboration of systems/applications (e.g. by using ICP modules as a Complex Event Processing Engine). Whereas, in DEWI, the core idea is integration of nodes within the same subnet (DEWI Bubble). A node may be a simple entity such as sensor or device, or complex, such as system, application, or even a subnet.

Let us now consider similarities and differences between ACCUS and DEWI, compared as examples of integration layer for cyber-physical systems. Table 1 shows how both projects address requirements imposed on cyber-physical systems, according to [2]. Both ACCUS and DEWI have been compared taking into account the general architecture of their respective developed solutions, their readiness to support time-constrained computing, data exchange and communication methods, and semantics.

Table 1. CPS characteristics in DEWI and ACCUS

Characteristic	**DEWI**		**ACCUS**
	Overall approach		**Overall approach**
	Centralized (Fusion Module + Context Aware And Reasoning Module)	**Decentralized (Wireless Data Aggregator)**	**Centralized (CSDB + CEDE)**
Architecture	Data Fusion System and Context Aware And Reasoning Module, which gathers and processes large amounts of data.	WDA device which allows nodes in the WSN (devices, systems, etc.) to cooperate.	Complex Event Detection Engine as an event processor. Integration and Coordination Platform, which connects urban subsystems and applications.
Time constrained computing	Time-critical applications, real time guarantees. Also, non time-critical: e.g. long term statistics in order to prediction.		Soft real-time and soft real-time applications only.
Communication/distribution	Via REST Web Service, using RDF, and SPARQL queries.	Via REST Web Service, using JSON-LD and SPARQL queries.	Via REST web service, using JSON format.
Semantics	Unified ontology for specific domain of every WSN, and general ontology to unite all WSNs.		Centralized approach. Data unified in ACCUS ontology model. SPARQL queries into Virtuoso triple repository or to the SQL database using D2RQ Platform and D2R server.

The DEWI project proposes a dual-architecture that accounts for both centralized and distributed computing. Here, the Data Fusion System and the Context Aware And Reasoning Module, are responsible for centralized processing of larger volumes of data, while the WDA is deployed at the WSN level, and carries local computing tasks. The ACCUS project, on the other hand, delivers a centralized architecture, where CPSs are integrated through the ICP as a middleware. Here, event processing is provided as a service within the ICP.

DEWI supports soft real-time computing and is designed to enable time-critical applications. In ACCUS, to some extent due to the specificity of scenarios and use cases planned for the project, the ICP does not provide real-time guarantees and thus is suited only for soft real-time and near-real time applications.

Let us now analyze both approaches in terms of basic features of CEP technologies [1]: event generators, event rules (definitions), event monitors, event distributors, and event consumers. Summary of findings is presented in Table 2.

Table 2. CEP characteristics in DEWI and ACCUS

Characteristic	**DEWI**		**ACCUS**
	Overall approach		**Overall approach**
	Centralized (Fusion Module + Context Aware And Reasoning Module)	**Decentralized (Wireless Data Aggregator)**	**Centralized (CSDB + CEDE)**
Architecture	WSN nodes (sensors, controllers, applications, systems, etc.) as sources of data. Dedicated modules to process events. Applications and systems as consumers.		CPS as sources of data. Complex Event Detection Engine to processing events. Applications, services and subsystems as consumers.
Events gathering	Centralized triplestore TDB (semantic approach).	Decentralized OrientDB graph database, clustering of many WDA devices, distributed, (semantic approach).	Central database at platform level (City State Database) for all integrated systems. Not necessarily semantic (both SQL and RDF support).
Events sources	WSN nodes (sensors, controllers, applications, systems, etc.) as sources of data.		Urban CPS as data sources.
Types of events	Atomic events: Events produced by devices (e.g. sensor) and simple data, which arrive from systems (e.g. database records). Molecular events: events produced by processing modules (result of processing atomic events).		Atomic events produced by subsystems (e.g. sensor) and simple data, arriving from subsystems (e.g. database records). Molecular events created by the CEDE service – result of composition of atomic events under defined conditions (event definitions).
Semantics	Unified ontology for specific domain of every WSN, and general ontology to unite all WSNs.		Centralized approach. Data unified in the ACCUS ontology model. SPARQL queries to Virtuoso triple repository, or to SQL database, using D2RQ Platform and D2R server.

In case of data exchange, both projects display similar features, as they use web-services and support SPARQL queries. Furthermore, in both projects, the JSON message format is used.

Both projects use semantic models to formally represent conceptual structure of their domains. In ACCUS, there is a central middleware, i.e. one common, unified urban ontology was proposed that integrates data models of each of the ICP-connected subsystems. In this way, ACCUS achieves semantic interoperability of urban systems. On the other hand, in DEWI, there is one, central ontology provided to integrate the interacting DEWI bubbles, while domain ontologies have also been proposed (and implemented) to handle the specific DEWI bubble environments.

In the architecture, very important, from the point of view of the CEP, are sources and consumers of data. In both cases, similar solutions are used, where consumers are systems and applications, while sources are spread across the web environment.

In ACCUS, there is only one central database. In DEWI, there are two possible solutions that differ in two aspects. First, the WDA devices lead to the decentralized architecture, because they are a set of cooperating devices. While supporting tasks distribution (e.g. gathering data from various nodes), they are sharing the database (a clustering mechanism). Second, their data history will not be persisted, because (1) there is no need for it, from the point of view of the systems operation, (2) this is not possible, due to limited resources of devices. To unify gathered data, this approach also uses an ontology, describing the domain, in which the WDA is running. However, for earlier described reasons, it was decided to use a graph database – OrientDB [15]. It allows to store RDF triples as a graph and perform SPARQL (semantic) queries. This is important, because of the possibility to use semantic data processing. In addition, this database has built-in clustering and data replication mechanisms. Hence, devices can use distributed, but "the same" database is used, and they share parts of the backup. In this case, the business rule engine does not use very complex mechanisms, such as statistics or learning and gathering knowledge, but focuses on checking the present conditions of frequently changing data, descending from different nodes. Research is ongoing, concerning choosing the optimal technology for this purpose. However, most likely, built-in OrientDB mechanisms will be used, allowing application of hooks (a trigger-like mechanism) that will be triggered immediately when a specific set of business rules will be fulfilled. Other approaches under consideration are: (1) use of rule-based engine Drools, (2) creation of a new solution (module) on the basis of SPARQL queries and scripts, for appropriate regulations.

Atomic events in DEWI are mainly sensor-like events (most of them are fed by sensors and devices), while in ACCUS they are produced by subsystems. In both cases, molecular events are the effect of analysis performed in the CEP module.

In both projects, semantic representation (a single ontology or multiple ontologies – "central" and domain) has been used to formally describe the complete domain environment. This is caused by the need for unification of things like: types of events, devices, systems. Therefore, all nodes in the net can be "understandable" to other nodes and vice versa.

Concluding remarks

The key differences in complex event processing solutions, demonstrated in ACCUS and DEWI projects, result from the two-fold architecture of the latter project. In DEWI, apart from a centralized server-based architecture, a framework for distributed network of sensor nodes has been proposed. Higher-level analytic and processing features are available in central servers, whereas embedded systems like the WDA offer similar functionality "closer" to the border of the "physical reality". Hence, it may be concluded that, due to the wireless data aggregation solutions, DEWI project accounts for edge-computing style solution (solution from a fog-computing domain). Further, which is typical for edge-computing, execution of computational processes "closer" to sensors (data sources) and actuators supports real-time guarantees, as local (fog-level) computation is less vulnerable in terms of latency, reliability, network failure, etc.

In that regard, the ACCUS project remains at the centralized, "old style" server-based higher level reference model. This has also potential advantages as what concerns possibility of efficient processing of data stored in a single repository.

Acknowledgment

Work has been completed within the framework of the EU COST action IC1404, Multi-Paradigm Modelling for cyber-physical systems (MPM4CPS).

A part of the research leading to these results has been conducted within ACCUS project (FP7/ARTEMIS-JU, 2013-2016, grant no. 333020) and DEWI project (FP7/ARTEMIS, 2014-2017, grant no. 621353).

References

Books:

1. Etzion O., Niblett P. (2010). *Event processing in action*. Stamford, CT: Manning Publications.

Chapters in publications:

2. Lee E.A. (2008). *Cyber Physical Systems: Design Challenges*. Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, Technical Report No. UCB/EECS-2008-8.

Articles:

3. Talcott C. (2008). *Cyber-Physical Systems and Events*. Software-Intensive Systems and New Computing Paradigms: 101-115.
4. Adi A., Botzer D., Nechushtai G., Sharon G. (2006). *Complex Event Processing for Financial Services*. SCW'06. IEEE: 7-12.
5. O'Donnella James., Corryb E., Hasanc S., Keaneb M., Curryc E. (2013). *Building performance optimization using cross-domain scenario modelling, linked data, and complex event processing*. "Building and Environment". Volume 62: 102-111.
6. Yaoa W., Chao-Hsien C., Zang L. (2011). *Leveraging complex event processing for smart hospitals using RFID.* "Journal of Network and Computer Applications". Volume 34, Issue 3: 799-810.
7. Zang C., Fan Y., Liu R. (2008). *Architecture, implementation and application of complex event processing in enterprise information systems based on RFID*. "Information Systems Frontiers". Volume 10, Issue 5: 543-553.
8. Wang W., Sung J., Kim D. (2008). *Complex Event Processing in EPC Sensor Network Middleware for Both RFID and WSN*. 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC): 165-169.
9. Zang C., Fan Y. (2007). *Complex event processing in enterprise information systems based on RFID.* "Enterprise Information Systems". Volume 1, Issue 1: 3-23.
10. Górski K., Groth M., Kulas L. (2014). *A multi-building WiFi-based indoor positioning system*. 20th International Conference on Microwaves, Radar and Wireless Communications (MIKON).

Internet:

11. ACCUS Homepage. http://www.projectaccus.eu (access date – 2016.12.01).
12. DEWI Homepage. http://www.dewiproject.eu (access date – 2016.12.01).
13. TDB Homepage. http://jena.apache.org/documentation/tdb/index.html (access date – 2016.12.01).
14. JENA Homepage. http://jena.apache.org (access date – 2016.12.01).
15. OrientDB Homepage. http://orientdb.com (access date – 2016.12.01).

16. Accessing Relational DBs as Virtual RDF Graphs. http://d2rq.org (access date – 2016.12.01).
17. Mike Brock and Various Contributors: MVFLEX Expression Language. https://github.com/mvel/mvel (access date – 2016.12.01).
18. ZigBee Homepage. http://www.zigbee.org (access date – 2017.06.22).
19. CCTV definition. https://en.wikipedia.org/wiki/Closed-circuit_television (access date – 2017.06.22).
20. CEP definition. https://en.wikipedia.org/wiki/Complex_event_processing (access date – 2017.06.22).
21. CPS definition. https://en.wikipedia.org/wiki/Cyber-physical_system (access date – 2017.06.22).
22. JBoss Drools Homepage. https://www.drools.org (access date – 2017.06.22).
23. JSON Homepage. http://www.json.org/json-en.html (access date – 2017.06.22).
24. JSON-LD Homepage. https://json-ld.org (access date – 2017.06.22).
25. MVEL definition. https://en.wikipedia.org/wiki/MVEL (access date – 2017.06.22).
26. OWL Homepage. https://www.w3.org/OWL (access date – 2017.06.22).
27. R&D definition. https://en.wikipedia.org/wiki/Research_and_development (access date – 2017.06.22).
28. RDF Homepage. https://www.w3.org/RDF (access date – 2017.06.22).
29. RFID Homepage. https://en.wikipedia.org/wiki/Radio-frequency_identification (access date – 2017.06.22).
30. SOA Homepage. https://en.wikipedia.org/wiki/Service-oriented_architecture (access date – 2017.06.22).
31. SPARQL Homepage. https://www.w3.org/TR/rdf-sparql-query (access date – 2017.06.22).
32. TRISTAR Homepage. https://www.tristar.gdynia.pl (access date – 2017.06.22).
33. WSN definition. https://en.wikipedia.org/wiki/Wireless_sensor_network (access date – 2017.06.22).