An event-driven platform to manage agility

Behavior adaptation in delivery context

Matthieu Lauras
Ecole des Mines d'Albi – Centre de Genie Industriel
Route de Teillet – 81000 Albi – France
matthieu.lauras@mines-albi.fr

Roland Stühmer
FZI Forschungszentrum Informatik
Haid-und-Neu-Str. 10-14 – 76131 Karlsruhe – Germany
roland.stuehmer@fzi.de

Yiannis Verginadis
Institute of Communications and Computer Systems,
National Technical University of Athens –
Athens – Greece
jverg@mail.ntua.gr

Frederick Benaben
Ecole des Mines d'Albi – Centre de Genie Industriel
Route de Teillet – 81000 Albi – France
frederick.benaben@mines-albi.fr

Abstract—**The European PLAY project aims at defining a federated-open-trusted platform able to deal with events emitted by any device. That platform should be able to combine a publish/subscribe mechanism with complex event processing in order to use the huge amount of produced events as a shared knowledge. The PLAY platform may so be able to (i) inform users (or devices) with relevant events and (ii) suggest appropriate behaviors to users (or devices). In the context of "city logistics", such platform could be useful to deal with vehicles' itinerary and priority management.**

Keywords- event-driven architecture; complex event processing; adaptation; agility; service-oriented architecture.

I. Introduction

In the last fifteen years, many researchers [1], [2], [3] and [4] have proposed the idea of "city logistics" to solve urban freight transport problems associated with congestion, negative environmental impacts such as air pollution and noise. But the problem of "city logistics" is expanded today. As for any part of the supply chain, the freight transport has to cope with a lot of uncertainties and companies must be able to respond quickly to these short-term changes in demand or supply. According to [5] and [6], this ability to respond quickly and adequately to short-term changes can be defined as "agility".

Based on the definition of [3], we can qualify "city logistics" as the process for optimizing the logistics and transport activities by private companies with the support of advanced information systems in urban areas within the framework of a market economy considering one or several following criteria: traffic environment, its congestion, safety and energy savings, uncertainties on supply or demand.

There are numerous types of city logistics schemes that have either already been implemented or are proposed such as: Advanced information systems, Co-operative freight transport systems, Load factor controls or Underground freight transport systems.

The purpose of this paper is to propose an agile vehicle routing procedure based on a federated – open – trusted (FOT) platform for event-driven interaction between services.

The paper is split-up in three sections. The first one presents the architecture of the proposed platform. The second one develops the service of adaptation we propose to improve the agility capability. The last one proposes a use case based on delivery of drugs to French pharmacies (including the description of relevant events and the dedicated business rules).

II. PLAY Platform

The field of city logistics is continuously evolving and new generation of projects appear. Most of them focus on the use of new technologies able to support coordination of shippers, carriers, and movements in an integrated logistics system.

The PLAY platform contributes to reach this goal. This is a Web-oriented structure to combine events from many sources with the goal of connecting and orchestrating services, devices and people.

The platform has emerged as an event marketplace, a place that brings together the senders and receivers of events and provides numerous services on top of them. To that end PLAY combines several technologies to deal with delivery, processing and storage of events as real-time information. We will briefly outline these technologies by introducing the components of the platform in Figure 1.

The Distributed Service Bus (DSB) provides the Service-oriented Architecture (SOA) and Event-driven Architecture (EDA) infrastructure to connect components, devices and end user services. The DSB enables the federation of separate SOAs through the formation of domains, which can be allowed to exchange events. Thus, distributed sources of events can be combined in the platform.

The Event Cloud provides storage and forwarding of events so that interested parties can be notified of events according to content-based subscription. The storage operates as an event history to fulfil queries for older events, which do not need real-time results e.g., when generating statistics. The Event Cloud is comprised of a peer-to-peer system of storage nodes organised in a CAN network [7].

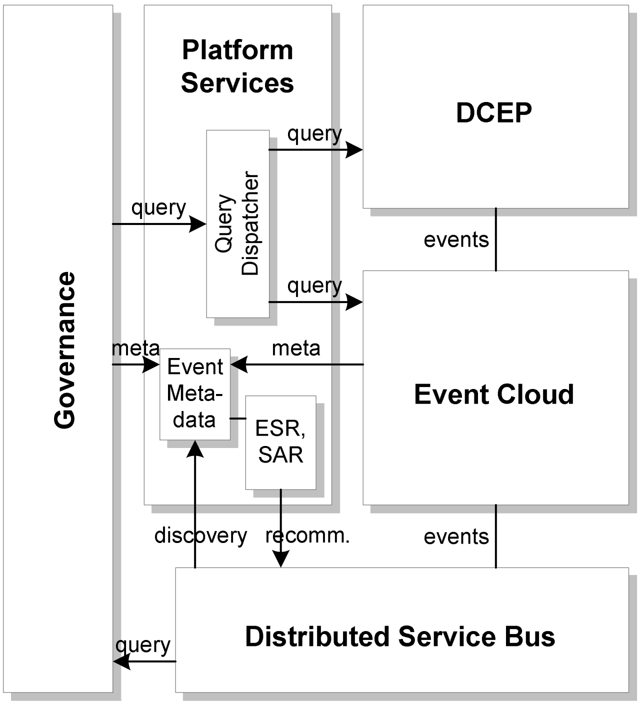

Figure 1. Conceptual Architecture

The Distributed Complex Event Processing (DCEP) component has the role of detecting complex events and do reasoning over events in real-time. Events per se might not be meaningful, but meaningful events can be derived from available, simpler events. The platform can readily detect such derived events, because it has knowledge of all events and applies event patterns, as described in [8], to the input events.

Finally, the Service Adaptation Recommender suggests changes (adaptations) of services' configurations, composition or workflows, in order to overcome problems or achieve higher performance. This component is described in more detail in the dedicated Section III.

Events in PLAY may originate from diverse devices, services and users such that a versatile event format and matching query language are required. To deal with this heterogeneity we propose an event format based on Resource Description Framework (RDF) with a matching SPARQL-based event pattern language syntax. Both base-technologies RDF [9] and SPARQL [10] are currently used on the Web as general methods for conceptual modelling (and querying, respectively) of information. We are adapting them to enable a real-time Web based on well-known foundations i.e., RDF and SPARQL.

Figure 2. shows an example event which we describe in detail below to explain the most important features and design decisions for our event format:

1. An event is **using quadruples** in TriG syntax [11]. The graph name (a.k.a context) before the curly braces is used as a unique identifier per stream e.g., to enable efficient indexing of contiguous triples in the Event Cloud for historic events.
2. There is an **event ontology** from which type `GpsPosition` is inherited. This ontology can be extended by any user of PLAY by referencing the RDF type `Event` as a super class.
3. An event can **link to other events** in different streams (not shown here). These events could have further linked events themselves. This allows modelling of composite events.
4. The event can link to static **Linked Data** where further context for the event can be retrieved e.g. from a vehicle registration database and a drivers' directory.
5. The event links to a **stream** where current events can be obtained as they happen.
6. There is a distinction made between the event `5534987067802526:event` and the Web resource `5534987067802526` to separate the URI for the **thing *event*** from the *event information* describing the event. The fragment identifier `#event` is used to identify the *event*.
7. There is an overlap of the namespaces. Unfortunately, stacking of namespaces like ~~`e:5534987067802526#event`~~ is not allowed, so we are using the absolute URI which avoids defining a new namespace for each unique event.
8. For geo-referencing events (where necessary) we are following the recommendations of NEOGEO [12] who will recommend standard ways of how to reference even complex geographical shapes from RDF.
9. The namespace `event-processing.org` was chosen as a generic home for this schema.

Figure 2. Example RDF Event

III. Adaptation Mechanism

In City Logistics area, several authors such as [3][4] identified issues related to the dynamic adjustments of vehicle and terminal operations within Information Systems. Nevertheless, [3] explain that they are not aware of any specific contribution to this topic.

The objective of the Adaptation Mechanism is bridge this gap by suggesting changes (adaptations) of services' configurations, composition or workflows, in order to overcome problems or achieve higher performance. Based on recognized situations, this mechanism will be able to define and detect adaptation "opportunities", proposing adaptation actions to the end users of service based applications involved in workflows. At the same time, it will undertake the responsibility of revealing to them the reasoning process that led to the adaptation recommendation, providing them with the capability of accepting or rejecting the proposed alterations.

In this section, we present requirements that apply for such mechanism called Service Adaptation Recommender. One of the basic capabilities for this mechanism is the situation awareness and detection functionality. This corresponds to the ability to sense situations relevant to service objectives and operation and the ability to track transitions between situations, by processing events and contextual service information. We consider complex events, detected in real time, as a way to signify situations that may require adaptation and we plan to enrich them with contextual information for defining dynamically what modifications are needed. Based on that, we will be able to make intelligent recommendations for service adaptations. This will improve service performance by detecting problems that need to be resolved (e.g. underperforming services, or suboptimal service workflows for the given situation) and by providing adaptation advices to be implemented in the appropriate workflow places.

We have defined some general requirements that apply here. Firstly, it is important that service adaptation recommender will be able to register for simple events as well as for complex events that carry combined information (e.g. the last 20 minutes in a pharmacy, there was a 20% orders increase that are related to a specific drug requested by the Wholesaler). Both simple and complex events need to carry semantics that will allow further processing and reasoning. The Event Cloud component in our platform must also guarantee that we will receive all events that they have been subscribed to. Failure to deliver an event to our adaptation mechanism may lead to the non-detection of a new situation. For reasoning purposes domain knowledge is needed, which may be comprised of an event ontology, a context and a situation model. Scalability is another general requirement that applies here, as the ability to cope with and extract valuable information from a "burst" of events, is imperative for detecting interesting situations. Finally, since our platform will be federated, it is important to detect global situations across the several service busses.

Service Adaptation Recommender is considered as a core part of our FOT platform and aims to provide recommender services for reacting at the right time and in the right way. This service adaptation mechanism will involve suggested changes of services configurations in order to overcome problems or improve service performance by:

- retrieving and analyzing service preferences/context
- analyzing service composition information
- deducing the situational model for the service using its semantics, composition, and preferences
- defining where in process flow we might need an adaptation
- evaluating and defining several alternative solutions of adaptation based on service adaptation strategies
- identifying and recommending and storing service adaptations as reaction to a current situation
- exploiting collective intelligence by deriving adaptation recommendations based on previous ones in similar situations

Service Adaptation Recommender will be Java based software component that will depend on the events acquired from our federated platform along with their semantics and the contextual information gathered from the related services. Adaptation recommendations may include: removal of a problematic service, replacement of an underperforming service, addition of a new service, alteration of the process flow. The latter is discussed in section IV of this paper, where events published by pharmacies (e.g. urgent orders, increased number of sales of a drug that was already requested from the Wholesaler) and also by other sources (e.g. traffic jam) are combined together in order to consider adaptation opportunities (i.e. altering wholesaler's vehicle routes in a real time).

In order to address the specific requirements of adaptation in dynamic event driven environments, we need a modeling formalism in order to describe the desired and meaningful system reactions to situations that demand for adaptation. The core notion of our approach is to use powerful models in order to enforce the fulfillment of goals. We adopted a goal model able to track the fulfillment of goals at run time. This model is called Situation-Action-Networks (SANs) and is a modeling framework that can be used for defining systems' reactions to situations that can occur before the goal is fulfilled. The main underpinning of SANs is to allow for high level definitions of goals, interesting situations and corresponding reactions at design time and to provide a reasoning mechanism that will be able to recommend at run time adapted reactions. SANs are tree structures whose nodes carry specific semantics used to model goal decompositions, enriched with flow control and planning capabilities. They provide a means to recursively decompose goals into subgoals, subgoals of subgoals, down to primitive actions. They furthermore provide plans for seeking and achieving the high-level goals. A plan is the order in which primitive actions should be taken, and the way they should be combined, in order to fulfill a goal. Such a modeling framework will be the backbone of our adaptation mechanism that we propose.

IV. "CITY LOGISTICS" USE-CASE

In this paper we develop the case of French drug deliveries to pharmacies. Actually, in the French context drugs and Over The Counter medicines are produced by different pharmaceutical companies and distributed through a two-levels network composed of wholesalers and pharmacies (see. Figure 3). The French law imposes a minimum of stock for each product in a wholesaler's warehouse. Consequently, in our study we consider no shortage at wholesaler level.

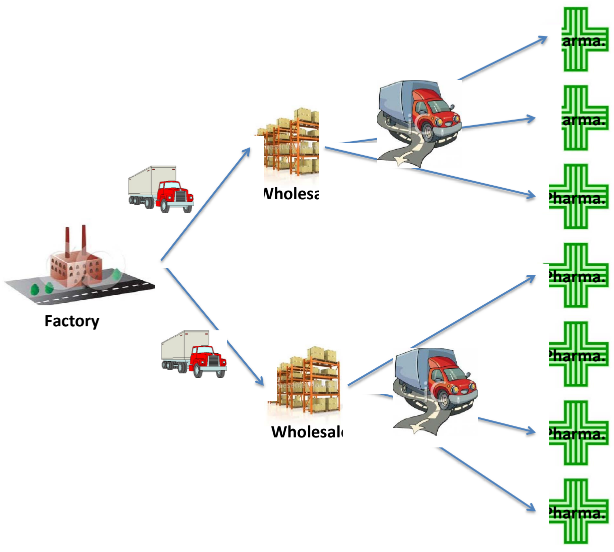

Figure 3. Considered distribution network

In this use case, we focus on products distributed from wholesalers to pharmacies. Each wholesaler supplies pharmacies of a given geographical zone, several times per day. Traditionally, these operations are organized through a set of scheduled rounds. Our idea consists in detecting some key information in order to be able to adapt the wholesaler's vehicle routes in a real time. These data can be considered as "events". All these events could be defined through taxonomy of event used to identify the different kind of events. This taxonomy of event is presented in TABLE I. For each kind of event, we define a structure based on WS-Notification and RDF. This structure will be used to identify the information contained in the event.

TABLE I. EVENT TYPES

Type of event	Description
Situational	This kind of event is used for information about the situation.
Consequences	This kind of event is used to transmit the result of one or a sequence of activities.
Activities	This kind of event is used for information about the state of service. A service can be done, in progress or waiting.
Resources	This kind of event is used for information about the resources. The resources can be information or objects.

In our case, events should include at least:

- GPS positions (situational event): to assess the traffic situation and/or potential problems during deliveries;
- Pharmacies' inventories (resource event): to assess the priorities in terms of supply;
- Patients' orders (consequence event): to assess the emergencies in terms of demand;
- On-going preparations at wholesaler's warehouse (activities event): to adapt time delivery schedule;
- …

Based on these events, we have to define some rules to be able to adapt the behavior of the system within the platform described previously. In our application case, these rules could be:

- If there is an important traffic jam 1km around a pharmacy then proceed to next pharmacy;
- If the route has to be modified then inform all the pharmacies for the delay;
- If there is an emergency for a drug which is inside the vehicle then proceed to this delivery first;
- …

The considered scenario concerns one vehicle that has to deliver to 5 different pharmacies (P) (see. Figure 4). The round schedule is the following:

1. Start from wholesaler's warehouse at 9:00 am
2. Arriving to P n°53 at 9:15 am and leaving at 9:25 am
3. Arriving to P n°22 at 9:45 am and leaving at 9:55 am
4. Arriving to P n°34 at 10:00 am and leaving at 10:10 am
5. Arriving to P n°11 at 10:30 am and leaving at 10:40 am
6. Arriving to P n°14 at 10:50 am and leaving at 11:00 am
7. Go back to warehouse at 11:20 am

At 8:55 am the vehicle is ready. Its round starts at 9:00 am.

Figure 4. Localisation of the different plants

At 9:13, the driver arrives to P n°53 and leaves at 9:21.

While he is driving to P n°22, he received an alert about a traffic jam due to a car accident just near the P n°22. The system then modifies the round schedule and proposes an alternative route to the driver in order to avoid the traffic jam and to access to P n°34 instead of P n°22. The system also sends information to P n°22 in order to inform that the delivery should be effective around 11:15 am. Finally, he arrives at P n°34 at 9:48 am and leaves at 9:58 am.

The new schedule indicates to the driver the route to go to P n°11. While he is driving to P n°11, he receives an alert about an urgent demand received by P n°14 and for which the products are in the delivery order he is transporting. Consequently, the system suggests modifying his route in order to deliver first P n°14 rather than P n°11. Finally the driver arrives at 10:28 at P n°14 and leaves at 10:43. The drug was finally delivered to the patient without any delay. The driver continues normally his route and arrives at P n°11 at 10:54 am and leaves at 11:05 am. We can remark that the Pharmacist received at 10:08 am information from the system to report about the move back of the delivery (from 11:30 am to 10:55).

The driver goes to P n°22 following the route suggested by the system. He arrives at 11:15 am and leaves at 11:25.

Finally, the driver goes back to the warehouse at 11:35 am.

This example shows some of the basic abilities of the PLAY platform and how event management could be useful in every day's life.

V. Conclusion

The PLAY platform provides an event management environment, which allows users to be dynamically connected (through topic and content based subscriptions). One very interesting aspect of that event-market place is the fact that users are not supposed to know each other or even to select the ones they want to get the events from. It is an overall structure of information sharing that selects the right information, for the right person at the right time. This is more or less the definition of the perfect information system. Consequently, there are some required constraints such as (i) quality of event patterns (business rules) and more generally mechanisms that allow the PLAY platform to filter, aggregate, deduce events from received ones, and (ii) trust, privacy and security.

Concerning the first point, one have to admit that "being informed" is nowadays one crucial issue. This is mainly because information exists and is probably available (which was not the case some decades ago) but it is quite difficult to find it (especially because of the very huge amount of available information…). One can simply think about emails and how difficult it is to deal with the number of email received per day. Event management, and particularly business rules that may be used to deal with these events in order to produce the right information, can be seen as one very promising opportunity. Defining these business rules and event patterns that should be used by the CEP engine to operate events is a really business-oriented activity. Consequently, event patterns editors and their accessibility, user-friendliness, are critical for the effectiveness and thus adoption of the PLAY platform.

Concerning the second point, there are currently some research work to integrate some social filter on the CEP in order to evaluate events and event providers [13]. This social filter should be in charge of defining the level of trust of any event provider (based on different considerations such as previously sent event, frequency of event sending, etc.). Again, similarity with email management may be noticeable (anti spam algorithm for instance).

On a technical point of view, one noticeable innovation concerns the RDF event format, which is versatile and uses open standards. Compared to related approaches, our RDF event format does not only model events but also comes with an execution model of how these events are consumed in a real CEP engine, namely DCEP. The final, main innovation is about EP-SPARQL, a query language, which goes together with the RDF data model and more specifically our event format.

Finally, the PLAY platform aims at providing a technical environment to deal with an anonymous, shared and universal knowledge. Perspectives with such a platform are immense: from telecommunication application, to crisis management [14], [15], from social network to transportation and supply-chain. In this article, the authors did focus on transportation and logistics uses of the PLAY platform. The provided example is quite simple but in a more complex one (especially involving complicated orchestration and choreography of workflows), benefits would be very tangible. Finally, such a platform is a solution to deal with complex processes management by ensuring coordination tasks (including adaptation tasks). By this way, human beings (and there decision skills) are free from tasks that do not require their human being skills, and could focus on decision tasks.

Regarding the city logistics problem statements, a real application case should be implemented with one of the three main French drug wholesalers in order to validate our proposition.

VI. Acknowledgment

The PLAY project *(Pushing dynamic and ubiquitous interaction between services Leveraged in the Future Internet by ApplYing complex event processing)* is being funded by the European Commission under Seventh Framework Program (Grant FP7-258659).

The authors would like to thank the project partners for their advices and comments regarding this work.

VII. References

[1] W. Ruske, « City logistics—solutions for urban commercial transport by cooperative operation management ». OECD Seminar on Advanced Road Transport Technologies. Omiya, Japan. 1994.

[2] U. Kohler, « An innovating concept for city-logistics ». 4th World Congress on Intelligent Transport Systems, Berlin, Germany.1997.

[3] E. Taniguchi, R. G. Thompson, T. Yamada and R. van Duin « City logistics—network modelling and intelligent transport systems ». Pergamon, Oxford. 2001.

[4] E. Taniguchi and H. Shimamoto, « Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times » Transportation Research Part C: Emerging Technologies, Volume 12, Issues 3–4. 2004.

[5] A. Charles, M. Lauras and L. Van Wassenhove, « A model to define and assess the agility of supply chains: building on humanitarian

experience » International Journal of Physical Distribution & Logistic Management, Vol. 40, N°8/9. 2010.

[6] H. L. Lee, « The triple – a supply chain » Harvard Business Review, Vol. 82 No. 10. 2004.

[7] I. Filali, L. Pellegrino, F. Bongiovanni, F. Huet, and F. Baude, « Modular P2P-based Approach for RDF Data Storage and Retrieval » Proceedings of The Third International Conference on Advances in P2P Systems. 2011.

[8] O. Etzion, and P. Niblett, « Event Processing in Action » , Manning Publications Co. 2010.

[9] G. Klyne, G. and J. J. Carroll, « Resource Description Framework (RDF): Concepts and Abstract Syntax », W3C Recommendation. 2004. http://www.w3.org/TR/rdf-concepts/

[10] S. Harris, and A. Seaborne, « SPARQL 1.1 Query Language ». 2010. http://www.w3.org/TR/sparql11-query/

[11] C. Bizer and R. Cyganiak, « The TriG Syntax », Technical report, FU Berlin. 2007.

[12] J. M. Salas, A. Harth, B. Norton, L. M. Vilches, A. D. León, J. Goodwin, C. Stadler, S. Anand, and D. Harries, « NeoGeo Vocabulary: Defining a shared RDF representation for GeoData », Technical report, FRLP, Universidad Tecnologica Nacional. 2011. http://geovocab.org/doc/neogeo.html.

[13] Hasan, O., Brunie, L., Damiani, E., « A Privacy Preserving Reputation Protocol for Web Service Provider Selection », 3rd IEEE International Workshop on Security Aspects of Process and Services Engineering (SAPSE 2011), in conjunction with the 35th Annual IEEE Computer Software and Applications Conference. 2011.

[14] S. Truptil, A. M. Barthe-Delanoe, F. Bénaben and R. Stuehmer, « Nuclear crisis use-case management in an Event-Driven Architecture ». Event-Driven Business Process Management Workshop, in conjunction with BPM Conference. Clermont-Ferrand. 2011.

[15] A. M. Barthe-Delanoe, F. Bénaben and M. Lauras « Improving the Design of Interoperable Platform Through Structured Case-Study Description Approach : Application to Nuclear Crisis ». ISCRAM conference. Portugal. 2011.