An Event-Driven System for Business Awareness Management in the Logistics Domain

Babis Magoutas[1], Dominik Riemer[2], Dimitris Apostolou[1], Jun Ma[2], Gregoris Mentzas[1], and Nenad Stojanovic[2]

[1] Information Management Unit,National Technical University of Athens, 9 Iroon Polytechniou str., 157 80 Zografou, Athens, Greece
{elbabmag,dapost,gmentzas}@mail.ntua.gr
[2] FZI Research Center for Information Technologies, Haid-und-Neu-Str. 10-14, 76131 Karlsruhe, Germany
{riemer,ma,nstojano}@fzi.de

Abstract. Modern organizations need real-time awareness about the current business conditions and the various events that occur from multiple and heterogeneous environments and influence their business operations. Moreover, based on real-time awareness they need a mechanism that allows them to respond quickly to the changing business conditions, in order to either avoid problematic situations or exploit opportunities that may arise in their business environment. In this paper we present an event-driven system that enables awareness about the situations happening in business environments and increases organizations' responsiveness to them. We illustrate how the proposed system increases the awareness of stakeholders about the running business processes, as well as their flexibility by presenting a practical application of the system in the logistics domain.

Keywords: Business Awareness Management, Logistics, Situation-Action-Network.

1 Introduction

To thrive in today's competitive market, organizations need to be agile, responding quickly to changing market conditions and exceeding customers' demands. Achieving business flexibility is a necessary condition for the business development of organizations, especially nowadays due to the global market downstream. Business flexibility, however, implies flexibility in the underlying ICT infrastructure and business processes.

A major challenge of current business process management solutions is to continuously monitor on-going activities in a business environment [1] and to respond to business events with minimal latency. Recent advancements in event-based systems and complex event processing [2] enable faster response to critical business events by efficiently processing many events occurring across all the layers of an organization and identifying the most meaningful ones within heterogeneous business environments.

M. La Rosa and P. Soffer (Eds.): BPM 2012 Workshops, LNBIP 132, pp. 402–413, 2013.

A recent research stream focuses on monitoring and responding to business situations detected through event patterns. For example SARI [3] provides an event-based rule management framework, which allows modeling business situations and exceptions with sense and respond rules. To wholly realize the potential of this research stream, it is essential to allow business users to model and intuitively comprehend the appropriate responses to business situations by using concepts that are familiar to them, like milestones and goals. Goal-orientation is based on separating the declarative statements defining desired system behavior from the various ways to achieve that behavior, thus hiding from business users the details about low-level events.

Hence, there is a clear need for an event-driven goal-oriented system, which would provide recommendations for reacting to interesting or critical business situations, while it would increase the awareness of business users regarding the running business processes.

In this paper, we focus on the challenges of enabling awareness about the business situations happening in business environments and increasing organizations' responsiveness to them. We present an event-driven framework for business awareness management, which aims to manage, i.e. monitor and control over time business situations and business systems that support the execution of business processes.

2 Motivating Scenario

Our presented scenario is based in the business area of large, end-customer oriented logistic companies like parcel deliverers. Usually, those companies operate a fleet of delivery vehicles that transport parcels, starting from a central repository, to the customer. Today, it is widely accepted that a predefined, optimized (in terms of time) routing plan leads to remarkable reductions of expenses in terms of fuel consumption and time savings. However, previously computed optimizations also imply less flexibility in the execution of business processes. Even in the case that routes can be changed during the business day, this is often a manually triggered, exceptional process that does not fully exploit new business opportunities that may appear.

From now on, we consider the case of a pickup service, which is the pickup of parcels directly at the customer's location. A pickup request is usually triggered by a customer via phone and entered in a CRM system by the call center agent. After that, one of the two situations applies: a) For important customers ("gold customers"), the agent might try to find a vehicle nearby to the customer and call the driver to pickup the parcel or b) the pickup is scheduled for the next business day.

Taking into account the real-time movement of vehicles (gathered by GPS devices) and external conditions like traffic, it is possible to automatically determine and inform the involved parties if it makes sense to pickup parcels immediately at the same day. We argue that adding more flexibility to this process can be beneficial for both customers and deliverers. Customers might profit from such an "express pickup" service in terms of the earlier pickup of parcels itself, while deliverers profit from increased customer satisfaction and less opportunity costs.

In the following sections, we provide a solution for this scenario by making use of advanced event processing capabilities and a contextual decision framework.

3 System Architecture

This section provides an overview of the major components our system is comprised of. Technically, the system is implemented as an event-driven architecture that connects several external services via a publish/subscribe broker using events in order to communicate between those components in a loosely-coupled fashion. Fig.1 shows the overall architecture of the proposed system.

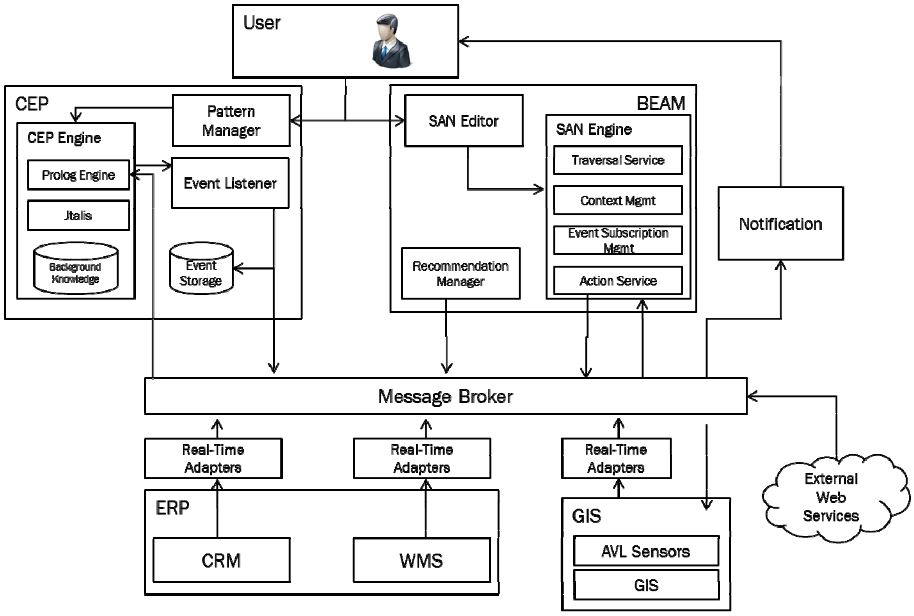

Fig. 1. Overall Architecture of the System

In general, we distinguish between event producers, software systems that provide the data needed for the presented use case, and event consumers, that subscribe to events in order to detect and trigger situations in real-time. As follows, we will describe the single components of the system architecture in more detail with an emphasis in the Complex Event Processing (CEP) and BusinEss Awareness Management (BEAM) components:

ERP. The ERP systems provide information about customers, orders and warehouse status. The systems have been extended with real-time adapters in order to directly capture status changes in the underlying systems. CRM produces events in case that new customers or orders have been created. Additionally, an adapter to a warehouse management system provides information about changes in a company's stock.

GIS. The Geographical Information System (GIS) consists of two parts. Vehicles are equipped with sensors that send information about the current position and fuel level to the system. The second part of the GIS is a service to provide information about routes.

CEP. Events in our scenario will come from multiple services as described above. These events are also known as atomic events, and they are instantaneous. Notifications about these occurred events together with their timestamps and possibly further associated data (such as involved entities, numerical parameters of the event, or provenance data) enter the event processing engine in the order of their occurrence.

Our engine is based on the logic-based ETALIS framework, which allows stream reasoning based on predefined background knowledge (e.g. duration of a working day). ETALIS correlates simple events in order to create complex (derived) events. The correlation is based on temporal, causal and semantic relations that can be established between events and possible background knowledge. An event query or pattern is a complex event description by means of which complex events can be specified as temporal constellations of atomic events. The complex events, thus defined, can in turn be used to compose even more complex events i.e., they can be turned back as input events. As opposed to atomic events, those complex events are not considered instantaneous but are endowed with a time interval denoting when the event started and when it ended. Event patterns in ETALIS are specified by a language for event processing [4]. Additionally, we provide a graphical editor for generating event patterns which abstracts from the technical prolog-based pattern language in order to allow a more user-friendly way of pattern definition as described in [5].

Finally – when detecting complex events – ETALIS may consult domain knowledge. This knowledge can be used to interpret events, data carried by events, as well as relations in which events and data are defined (e.g., subclass relationships etc.). ETALIS can evaluate the background knowledge on the fly, possibly inferring complex events that involve new implicit knowledge. This knowledge is derived as a logical consequence from event driven deductive rules, thereby providing the Stream Reasoning capability.

ETALIS internally comprises of three components. ETALIS Core engine is written in Prolog language, hence it requires a Prolog system to run (e.g., SWI, YAP, XSB etc.). Still to interface ETALIS Core engine with the message broker we provide a Java interface called jtalis. Event queries (patterns) are written in ETALIS Language for Events. ETALIS compiles these patterns into executable rules (written in Prolog). These rules may be accompanied with background knowledge to describe the domain of interest (as discussed above). Domain knowledge is expected to be expressed either in Prolog or as an RDFS ontology. Currently, we support usage of static domain knowledge defined in Prolog, more potential advantages of using domain knowledge within event processing that are planned to be integrated in the future are described in section 6. A backing store provides a log service for events. It records log data about current and past event interactions. This includes atomic and complex events as well as intermediate events that were generated during the detection of complex events. Providing a persistent storage for atomic, complex and intermediate events will be used to analyze an existing set of event patterns and to automate process of creation of new patterns.

BEAM. BEAM is based on the assumption that there exist specific goals that a business process and system should fulfill and proposes the adoption of a goal-directed model able to track the fulfillment of goals at run time. We utilize goal-directed modeling in which we follow a hierarchical goal decomposition model we have developed as part of our previous work, called Situation-Action-Network (SAN) [6]. In SANs, goals are related to situations that trigger their activation and reactions that should be performed towards achieving goals if certain conditions are met; for details about the SAN models the reader is referred to [7].

In BEAM, the detection of critical and/or interesting situations is performed by employing CEP capabilities. BEAM exploits the complex event patterns identified by the CEP engine in order to sense critical/interesting situations. Then, by taking into account the current business context, BEAM recommends appropriate responses with the aim to cope with problematic situations or exploit opportunities that may arise in the business environment. More details about our approach for context management can be found in [8].

The BEAM layer consists of several sub-components allowing the definition of goal-oriented situation-aware recommendations, the modeling of desired, meaningful reactions to interesting situations and the execution of the related SAN models. The SAN Editor is a graphical design tool developed in Adobe Flash/Flex. It is used for the development of SAN models represented in RDF (Resource Description Framework). SAN Engine undertakes the traversing of SAN trees stored in SAN Repository (implemented using Sesame 2.6.2) through the Traversal Service, as well as the subscription/unsubscription of complex event patterns in the Pub/Sub through the Subscriptions Management subcomponent. The Context Management subcomponent updates the current context and evaluates the necessary contextual conditions based on detected situations. Context changes are stored into a Context Repository. Action Service triggers actions in external systems and recommends actions to human actors by transmitting events through the message broker. Finally, the SAN engine is responsible for recommending actions which include notifications to users, subscriptions to other simple or complex events, and adaptations to running business processes.

Notification. A notification module receives recommendations from BEAM and informs participants of results, e.g. the driver to pickup some parcels or the customer about the time of pickup.

External Web Services. To enable the usage of external data like traffic information, our system provides adapters to integrate such services.

4 System Walkthrough

In the section we present a practical application of the proposed event-driven system for business awareness management in the logistics domain and more specifically in a Courier company. In order to understand how the proposed system can increase the awareness of managers about the running business processes, as well as their flexibility we discuss in the following an indicative pilot scenario. The pilot scenario is used for describing the practical role and use of the proposed system framework.

Consider the situation where a gold customer requests a pickup and provides his/her availability in terms of a time window. Information about the pickup including the location of the customer is incorporated into a pickup event generated by the CRM system. In ETALIS, a pickup event would be defined as follows:

```
pickupRequest(EventId, Timestamp, OrderId, CustomerId,
Latitude, Longitude)
```

Latitude and longitude have been calculated in advance by geocoding the address of the customer from the underlying CRM system. In the same time window that the customer has indicated his/her availability, one or more of the company's trucks are moving into a geographical area which is rather close to the customer. The following ETALIS pattern describes this situation (by looking for a vehicle that is within a range of 10km of the customer's location):

```
rule 1: nearbyVehicle(EventId, VehicleId, OrderId,
Distance, Latitude, Longitude) <- gpsEvent(EventId,
VehicleId, Latitude, Longitude, Speed) AND
pickupRequest(EventId, Timestamp, OrderId, CustomerId,
Loclat, Loclng) where getDistance(Latitude, Longitude,
Loclat, Loclng, Distance), Distance < 10
```

The getDistance function is a mathematical function that calculates the distance between two latitude/longitude pairs. After a nearbyVehicle event has been produced by ETALIS, it should be checked whether the specific vehicle actually moves to the right direction of the target location. Therefore, the following pattern is deployed:

```
rule 2: vehicleRecommendation(EventId, VehicleId,
OrderId, H, I) <- (nearbyVehicle(A, VehicleId, orderId,
D, E, F) SEQ nearbyVehicle(B, VehicleId, orderId, G, H,
I) where G<D)
```

Other rules that need to be added in order to ensure delivery within a certain time window or to avoid multiple recommendations of the same vehicle are not shown in these examples.

The aforementioned situation may be an opportunity for the company, as it may be beneficial to add dynamically an additional stop to the routing plan of one of its trucks, in order to allow it to pickup immediately the package. In this scenario the proposed software framework identifies interesting business situations like the one described above, investigates the feasibility and benefits of the various alternative actions and recommends the best one, while it also informs the employees of the company about the detected situation(s).

More specifically, BEAM starts the execution of the underlying SAN model (see Fig.2) when the aforementioned situation is detected by the CEP engine, i.e. when a truck is nearby the customer during the given time window (see point 1 of Fig.2), while it triggers a GIS query to compute a new route for the truck and an estimation of the time needed to get there (see point 2). It should be noted that traffic services are available for the GPS devices embedded in the company's trucks that provide them through FM signals with information about the traffic conditions. In this scenario, GIS generates a new event containing an estimation of the time needed for the truck to get to the customer and pick up the product, as well as the information that there is currently a traffic jam nearby the customer (see point 3).

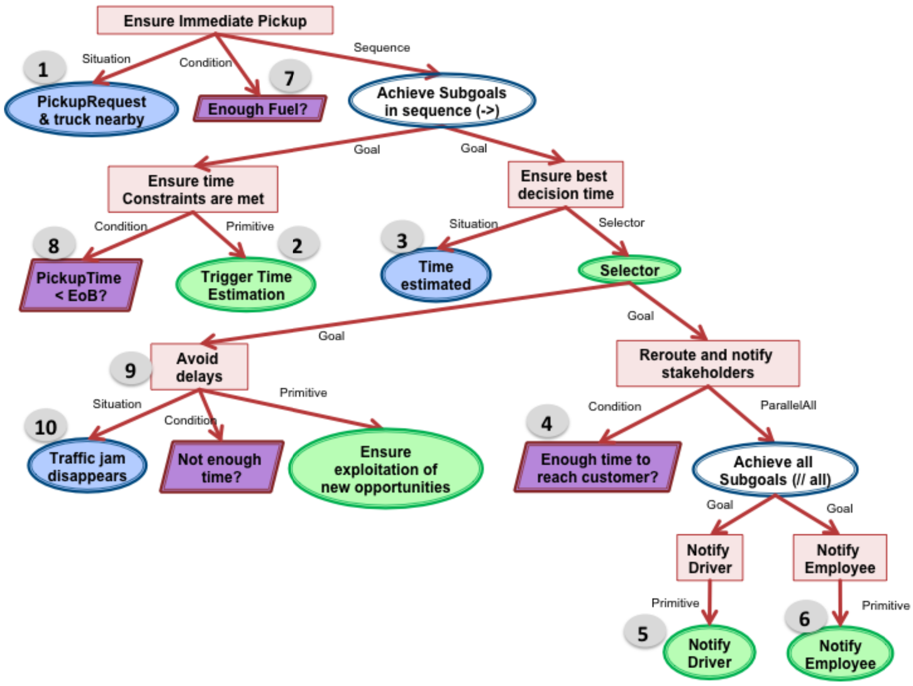

Fig. 2. Underlying SAN model of the Scenario

In case that the truck is close to the customer or it is estimated that the truck will be able to get to the customer in a rather short time (see point 4), while on the same time additional contextual parameters of the business environment allow it, BEAM decides to recommend the addition of an extra stop to the truck's route by triggering GIS through an appropriate event (see point 5) and notify the employee about that (see point 6). Regarding contextual parameters, these include specific business contexts of the Courier company for which it makes no sense to even investigate the possibility of recommending the addition of the extra stop to the planned route. Examples include the case that the truck has not enough fuel (see point 7), the case that the pick up request occurs close to the end of the company's business day (see point 8 - EoB stands for End of Business Day), the case that the truck is failing to execute the deliveries according to the specified time schedule (see point 4) and so on. So, in all of the above cases BEAM makes no recommendation.

On the other hand, in case that the business context allows it and the estimated time is above a predefined threshold (e.g. one hour), BEAM decides to "decide later", in the sense that it defines a new situation to be monitored by the CEP engine in terms of a complex event pattern (CEPAT), in which a decision about the package pickup can be made (point 9). The CEP engine monitors events from an online traffic service, as well as the GPS trails of the company's trucks and figures out - by detecting a CEPAT previously deployed by BEAM - that the traffic jam nearby the customer disappears (point 10) and another vehicle is approaching the customer, so it informs BEAM about the new opportunity. BEAM enters again the decision loop and this time recommends to the new truck to pickup the package from the customer, while it informs the customers about the estimated pickup time based on the calculations of GIS. As real time information is taken into account for estimating pickup time, the estimation is much better than the current state. Finally, BEAM generates an event that informs the business users about the successful pickup of the package.

5 Related Work

To enable Business Awareness Management, we examined and reviewed technologies that can enable awareness about changing circumstances that may require reactions as well as mechanisms for monitoring business activities. In this context, we consider relevant for our work approaches from the research fields of Situation Awareness, Business Activity Monitoring (BAM), goal-orientation and Complex Event Processing (CEP).

Situation awareness was introduced by Mica Endsley whose definition of the term is a generally accepted one: "Situation Awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" [9]. Since this original work, a lot of situation-related research has been carried out and has become a critical issue in domains in which there is the need to automatically and continuously identify and act on complex, often incomplete and unpredictable, dynamic situations; as a result, effective methods of situation recognition, prediction, reasoning, and control are required — operations collectively identifiable as situation management (SM) [10].

BAM describes the processes and technologies that provide real-time situation awareness, along with access to, and analysis of, the critical business performance indicators, based on the event-driven sources of data [11]. BAM is used to improve the speed and effectiveness of business operations by keeping track of what is happening now, and raising awareness of issues as soon as they can be detected. BAM applications may emit alerts about a business opportunity or problem, drive a dashboard with metrics or status, make use of predictive and historical information, display an event log, and offer drill-down features [12].

In principle, goal-orientation is based on separating the declarative statements defining desired system behaviour from the various ways to achieve that behaviour [13]. Rimassa and Burmeister [14] propose GO-BPMN, a visual modelling language for the specification of business processes, which is an extension of the OMG standard BPMN. This notation helps to add goals, activity plans, and their relationships to process models. The Tibco's approach for goal-driven BPM [15], [16] follows a process of sense-and-response incremental improvements, making possible the creation of the most dynamic, agile, and responsive processes. For the "sense" part of the aforementioned process, the approach exploits CEP in order to identify important business-worthy events and respond to them; i.e. the response is event-driven.

CEP is a very active field of research and is being approached from many angles. A multitude of languages are proposed to formulate complex event patterns and different event processing paradigms are proposed to match these patterns over events [2], [17].

The framework proposed in the context of this paper enhances existing BAM applications by incorporating concepts and technologies from the research areas of CEP, situation awareness and goal-oriented business modeling. According to [18], most key performance indicators (KPIs) in business activity monitoring and performance management scenarios are complex events (although not all complex events are KPIs). Although most process monitors implement a basic form of CEP in the sense that they apply rules and perform computations on multiple event objects to calculate what is happening in a business process, they are not general-purpose CEP engines, and they don't "listen" to events from outside the managed business process. The proposed approach links CEP with business process monitoring, allowing the provision of a broad, robust situation awareness capability that encompasses both internal process events and external business events. BAM on the basis of CEP shall improve the existing, often complained IT blindness, which is caused by thousands of low-level events per second without any semantics [19].

BAM usually sets up target values for each performance indicator. These target values are not the goals of the business processes, but the goals of the performance indicators. They usually lack meaning or purpose and are just values to be reached. They exist separately without relationships and hence it is hard to share a united vision of the monitored business processes. Therefore, there is a need to align the business processes with strategic goal architectures, which requires a change in emphasis from process to goal-oriented monitoring [20]. The proposed BEAM layer enables goal-oriented monitoring by using a process description oriented toward goals related to interesting situations, instead of using BPMN (or similar) process descriptions; for an example of the later approach please see [21], where the authors present a general framework for edBPM as well as a use case in the context of a large logistic company. Further, our approach incorporates a rule model that enables dynamic parsing of rules. Parsing or rules is dynamic because rules are hierarchically nested; rules lower in the hierarchy are activated only when a parent rule is fired. Finally, goal orientation of our approach enables users to effortlessly conceptualize critical business situations as well as to define and group hierarchically pertinent rules for monitoring critical situations.

6 Concluding Remarks and Further Work

Modern organizations need real-time awareness about the current business conditions and the various events that occur from multiple and heterogeneous environments. Moreover, the need for flexible processes is big in today's competitive environment as a lost customer, or a missed opportunity to recruit a new customer, may never be recouped. In this paper we presented a software framework dealing with such a need and an application of the framework in the logistics domain.

The adoption of the proposed approach would bring added value to all stakeholders compared to the current situation. For example, currently there are several problems in the case that the courier company utilizes predefined routing plans that cannot be changed during the business day. The customer has to wait until the next routing plan in order to have his/her package picked up. Moreover, while several drivers of the company having available space in their trucks may pass nearby the customer, both the drivers and the employees lack the necessary knowledge to take the right decision. These problems result in delays, additional costs for the courier companies in terms of personnel and transportation, unsatisfied customers and even lost customers in the very competitive courier industry.

Even in the case that the Courier company is using some sort of dynamic planning for specific customer categories (e.g. gold customers), the adoption of the proposed approach would bring additional added value as information is communicated to the interested stakeholders by following a “push” rather than a “pull” communication paradigm. For example, in the as-is situation employees of the Courier company take the initiative to get information (information pull paradigm) about the location of trucks when they receive a pickup request, in order to identify possible opportunities for immediate pickups. On the other hand, the proposed system identifies automatically and “pushes” to them information about the various opportunities, making the whole process less laborious and time-consuming. Moreover, currently dynamic planning is performed on a per employee basis, in the sense that each employee of the company is responsible to take a decision about planning based on subjective criteria when a pickup request arrives; therefore it is very difficult for the company to enforce a common strategy per customer segment.

In addition to enabling an “express pickup” service as discussed in the context of this paper, our vision for the proposed software framework is that it can facilitate the transition to a new business model of pickup and delivery services in the courier industry, where the price of the service varies in real-time depending on the market conditions and the user requirements (e.g. urgency of delivery). Today, most of the courier companies simply use flat rates for entire cities (depending on the size of the package, of course) for next-day or same day courier services. However, customers are demanding more from their courier partners. A shift where, more and more customers prefer to use the services of companies who are able to provide more flexibility and levels of service is being witnessed [22]. Therefore, some customers may be willing to pay more for getting specific features of a courier service like immediate pickup in less than one hour or exact estimation of the pickup time.

The proposed framework is able to support such features, which constitute a real advancement compared to what is possible today. In the new real-time pricing model the customer satisfaction will be given more importance as the customer himself will be able to track the order, get the perfect details of where his parcel is and when it will be delivered and at what time and cost. On the other hand the courier companies will be able to exploit the business opportunities in an ad-hoc manner, reduce inventory to a minimum, save substantial money while at the same time increasing their revenues.

To better support this new model we plan several extensions of the proposed system such as making use of static data, namely historical events as well as integration of existing ERP databases. By combining knowledge from the courier's database, (static) information about parcel sizes (that have been loaded into the vehicle as well as the size about already delivered parcels) and the current space left in a vehicle, pickup recommendations can be improved. Additionally, by using historical events processed by the system, it becomes possible to make predictions [23], e.g., if it makes sense to pickup a parcel, when a specific traffic situation exists. This requires the computation of partial patterns as proposed in [24] and historical events, in order to compute the degree that an event pattern has been fulfilled. If a pattern has been fulfilled by a certain degree, it is possible to estimate the probability of its complete fulfillment in the near future.

Finally, regarding the evaluation of the proposed system, we are currently deploying it into a logistics company in the context of an FP7 project, while its complete and thorough evaluation is expected until the end of the year. The evaluation will concern issues like the usability, scalability and technical performance of the system, while its added value will be examined by assessing the satisfaction of all stakeholders. Finally, in the context of the evaluation we will empirically calibrate some predefined parameters of our model (e.g. time thresholds)in order to increase the anticipated benefits for all involved stakeholders.

Acknowledgements. This work is partly funded by the European Commission projects FP7 SME ReFLEX(262305) and FP7 ICT PLAY (258659).

References

1. Sayal, M., Casati, F., Dayal, U., Shan, M.-C.: Business Process Cockpit. In: Proceedings of the 28th VLDB Conference, HP (2002)
2. Luckham, D.: The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems. Addison-Wesley, Reading (2005)
3. Schiefer, J., Rozsnyai, S., Rauscher, C., Saurer, G.: Event-driven rules for sensing and responding to business situations. In: Proceedings of DEBS 2007, pp. 198–205. ACM, New York (2007)
4. Anicic, D., Fodor, P., Rudolph, S., Stühmer, R., Stojanovic, N., Studer, R.: A Rule-Based Language for Complex Event Processing and Reasoning. In: Hitzler, P., Lukasiewicz, T. (eds.) RR 2010. LNCS, vol. 6333, pp. 42–57. Springer, Heidelberg (2010)
5. Sen, S., Stojanovic, N.: GRUVe: A Methodology for Complex Event Pattern Life Cycle Management. In: Pernici, B. (ed.) CAiSE 2010. LNCS, vol. 6051, pp. 209–223. Springer, Heidelberg (2010)
6. Patiniotiakis, I., Papageorgiou, N., Verginadis, Y., Apostolou, D., Mentzas, G.: A Framework for Situation-Aware Adaptation of Service-Based Applications. In: 1st International Workshop on Adaptive Services for the Future Internet (WAS4FI), in Conjuction with ServiceWave, Poland (2011)
7. Verginadis, Y., Papageorgiou, N., Patiniotakis, I., Apostolou, D., Mentzas, G.: A goal driven dynamic event subscription approach. In: 6th ACM International Conference on Distributed Event-Based Systems, DEBS 2012, pp. 81–84. ACM, New York (2012)
8. Verginadis, Y., Patiniotakis, I., Papageorgiou, N., Apostolou, D., Mentzas, G., Stojanovic, N.: Context Management in Event Marketplaces. In: 7th International Workshop on Semantic Business Process Management, Heraklion, Greece (2012)

9. Endsley, R.: Toward a theory of situation awareness in dynamic systems: Situation awareness. Human factors 37, 32–64 (1995)
10. Jakobson, G., Buford, J., Lewis, L.: Situation management: Basic concepts and approaches. Information Fusion and Geographic Information Systems, 18–33 (2007)
11. McCoy, D.: Business Activity Monitoring: Calm before the storm (2002), `http://www.gartner.com/DisplayDocument?doc_cd=105562`
12. McCoy, D., Cantara, M.: Hype Cycle for Business Process Management (2010), `http://www.gartner.com/DisplayDocument?doc_cd=200990` (accessed October 26, 2011)
13. Greenwood, D., Rimassa, G.: Autonomic Goal-Oriented Business Process Management. In: Proceedings of the Third International Conference on Autonomic and Autonomous Systems, p. 43. IEEE Computer Society (2007)
14. Rimassa, G., Burmeister, B.: Achieving Business Process Agility in Engineering Change Management with Agent Technology. Whitestein Technologies AG & DaimlerChrysler AG, Group Research (2007), Retrieved from `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.145.4626`
15. Tibco: Goal-driven business process management: Creating agile business processes for an unpredictable environment. Tibco Whitepaper (2006)
16. Tibco, Software, and Inc.: Tibco iprocess conductor (2007)
17. Etzion, O., Niblett, P.: Event Processing in Action, 1st edn. Manning Publications Co., Greenwich (2010)
18. Schulte, W.R., Sinur, J.: Use of Event Processing Increases Success of BPM Projects (2009), `http://www.gartner.com/DisplayDocument?id=993012`
19. Ammon, R.V., Silberbauer, C., Wolff, C.: Domain Specific Reference Models for Event Patterns – for Faster Developing of Business Activity Monitoring Applications. In: VIPSI 2007, Lake Bled, Slovenia, October 8-11 (2007)
20. Chen, P.: Goal-oriented business process monitoring: an approach based on user requirement notation combined with business intelligence and Web services. Carleton University Dissertation, Ottawa (2007)
21. Ammon, R.V., Emmersberger, C., Springer, F., Wolff, C.: Event-Driven Business Process Management and its Practical Application Taking the Example of DHL. In: 1st International Workshop on Complex Event Processing for Future Internet - Realizing Reactive Future Internet, FIS 2008, Vienna, Austria, September 28-30 (2008)
22. Scruz, A.: Study of the supply chain in courier industry. Dissertation Thesis, Christ University Institute of Management (2011)
23. Engel, Y., Etzion, O.: Towards proactive event-driven computing. In: Proceedings of the 5th ACM International Conference on Distributed Event-Based System, DEBS 2011, pp. 125–136. ACM, New York (2011)
24. Stojanovic, L., Ortega, F., Canas, L., Duenas, S.: ALERT: Active Support and Real-Time Coordination Based on Event Processing in Open Source Software Development. In: 15th European Conference on Software Maintenance and Reengineering, pp. 359–362 (2011)