

AGENT-BASED DIAGNOSTICS IN SUPPLY NETWORKS

Karl Kurbel, European University Viadrina, kurbel.bi@uni-ffo.de
Denny Schreiber, European University Viadrina, schreiber@uni-ffo.de

ABSTRACT

No matter how thoroughly supply chain operations are considered in the planning phase, disruptions and other unforeseen irregularities may occur in the execution phase. In many cases, fast reaction to such events is crucial, because they require immediate detection of the events as they occur. In this paper an approach for automatically detecting problems in supply networks with the help of software agents is discussed. Simulation runs were performed to evaluate system behavior. The architecture of the underlying multi-agent system and the results of the simulation are presented. Benefits and drawbacks of the approach are discussed.

Keywords: supply networks, software agents, supply chain event management, diagnostics

INTRODUCTION

Supply chains, or rather supply networks, are difficult to manage in today's globally distributed market economies. Planning the supply chains and the processing of particular orders may have been excellent, but what happens in reality when the orders go through the chains is a different story. Deviations from the plan, disruptions, or in the worst case, disasters make the life of supply chain managers difficult [13]. Their responsibility is to react quickly, or even better, to anticipate such situations. Hence, the concept of *supply chain event management (SCEM)* emerged, which synchronizes supply chain planning and execution. SCEM is the "application of statistical, process and technological identification and control solutions to standard and nonstandard events" [1]. Typical supply chain problems can result from shortage of materials, last-minute order changes by customers, ramping up and rollout problems, as well as problems with product quality and manufacturing [11].

A precondition for effective SCEM is transparency across the whole supply network. Supply chain visibility can be achieved with the help of RFID (radio frequency identification), Auto-ID (automatic identification), and mobile technologies [6].

In this paper, an approach to improve the detection of problems in supply chains is presented. Following the 'construction and building' approach as described by Backlund [2], we investigate the research questions:

- How can Auto-ID, mobile, and agent technologies be combined in SCEM applications to improve current SCEM solutions?
- What are the benefits of the approach?

As to answer these research questions, discrete event simulation was applied to evaluate SCEM system behavior with regard to the speed of detecting unplanned events.

The remainder of the paper is organized as follows. In the next section, the state of the art in supply chain event management and related work are outlined. Afterwards, the architecture and implementation of the **Mobile Agent-based SCEM System (MASS)** used in our work is described. Then the simulation approach to evaluate MASS behavior is presented. In the final section conclusions and an outlook to future work are given.

BACKGROUND AND RELATED WORK

Monitoring of goods and resources within and between enterprises can be accomplished with the help of tracking & tracing (T&T) systems and solutions such as Oracle Business Activity Monitoring and SAP SCEM Monitor. Zimmermann gives an overview of T&T and SCEM systems, stating that current systems lack proactive data gathering and do not sufficiently support network-wide monitoring [18].

While current systems are mostly based on target-state comparisons, our approach includes statistical analyses as well. State-of-the-art systems usually apply statistical methods only at the end of a process step (e.g. transportation), if at all [1, 18, 5]. Permanent monitoring of goods and machines throughout the process steps, e.g. during manufacturing or transportation, can effectively help to anticipate upcoming problems and treat them proactively. On-demand calculation of the values needed for the analysis is another crucial element.

Agent technology is a promising approach to the above problems because software agents exhibit favorable characteristics such as autonomy, proactivity and adaptivity [14]. Agent-based approaches for monitoring supply networks exist already, e.g. Dialog [12], CoagenS [8], and Agent.Enterprise [15, 18]. These research prototypes have made significant contributions regarding the detection and solution of problems in supply networks. However, more work is needed in order to realize permanent monitoring of goods and resources. Network-wide problem detection and resolution still have to be improved. Automating this process is also an open research question.

MOBILE AGENT-BASED SCEM SYSTEM

The rationale for the MASS system is to enhance SCEM with analytical methods and up-to-date technologies in distributed environments. One step in this direction is on-demand production of analytical data. With better and more precise measurements, accuracy of the diagnosis will also improve. Today’s technological means make it possible to automate the diagnosis and to initialize counter-measures in response to upcoming problems. Auto-ID systems based on RFID and sensor systems combined with mobile technologies can be used to track goods and resources wherever mobile networks are available [6]. Hanebeck & Tracey [9], for example, suggest using GPS and mobile triangulation for permanent tracking. This means that tracking is not limited to a fixed number of tracking points, but can be enabled whenever the process owner wants to.

Software agents in MASS collect and analyze data provided by monitoring systems from the entire supply network. For example, agents can compare the drivers' routes from the route planning system with tracking data provided by the T&T system and conclude what type of event occurred. Table 1 shows some examples of symptoms and underlying events that may occur on a network link in a transportation step. Conditions and possible conclusions are represented as a decision table.

Table 1 also shows that not everything is completely clear. For example, rule R4 is ambiguous. Automated measures cannot be taken since the cause of the problem may be either the driver or the vehicle. RFID-based technologies could help in this situation because they would allow to distinguish between the state of the driver and the state of the transport means. However, tracking humans is a sensitive topic due to privacy concerns [17]. Instead of automated

problem detection, the driver's feedback is needed to resolve the ambiguity.

Table 1. Diagnosis rules for transportation step

	R1	R2	R3	R4	R5	R6	R7
Symptoms							
Mileage	ok	ok	not ok	not ok	not ok	ok	ok
Path deviation	ok	ok	ok	ok	ok	not ok	ok
Speed	ok	ok	stop	stop	slow	ok	ok
State	ok	not ok	not ok	ok	ok	ok	ok
Timestamp	ok	ok	ok	ok	ok	ok	not ok
Diagnosis - possible events							
Faultless process	x						
Damaged goods		x					
Fatal crash			x				
Vehicle problem				x			
Driver injury				x			
Traffic jam					x		
Driver lost way						x	
Data out of date							x

Architecture of the System

The architecture of MASS is presented in figure 1. It is divided into four layers. In a bottom-up view, the first layer (data layer) contains the tracking data files and the databases of the underlying enterprise resource planning (ERP) system and the multi-agent system (MAS) hosting the software agents.

The main component of the architecture is represented by the *service layer*. This layer uses web services for exchanging data between the ERP system and the tracking & tracing system. Access to the database of the multi-agent system is achieved by so-called *wrapper agents*. The business logic of the MAS is divided into several separate tasks assigned to different types of agents. *User agents* mediate between human users and service-providing agents.

The *gatekeeper agent* authenticates the users of the system. A human user can trigger the *monitoring agents* to supervise the execution of jobs. If a problem occurs, a notification containing the detected

event is generated and sent to the user agent in charge. This agent can notify the human user via e-mail, SMS or other forms of alerts depending on the user's settings.

Resource agents are responsible for automatic problem solving. They interact with monitoring agents across the whole network. Resource agents are designed as mobile agents. They migrate to other MASS platforms in order to negotiate possible solutions with agents there.

Implementation of MASS

Our prototypical implementation of MASS uses Compiere ERP [7] as an underlying ERP system on top of an Oracle 10g database. Access to Compiere's business logic and to ERP data is achieved through a web service façade residing on a Jboss application server. T&T system functionality is also accessed through a web service façade in Jboss that uses a set of Java classes simulating a real T&T system.

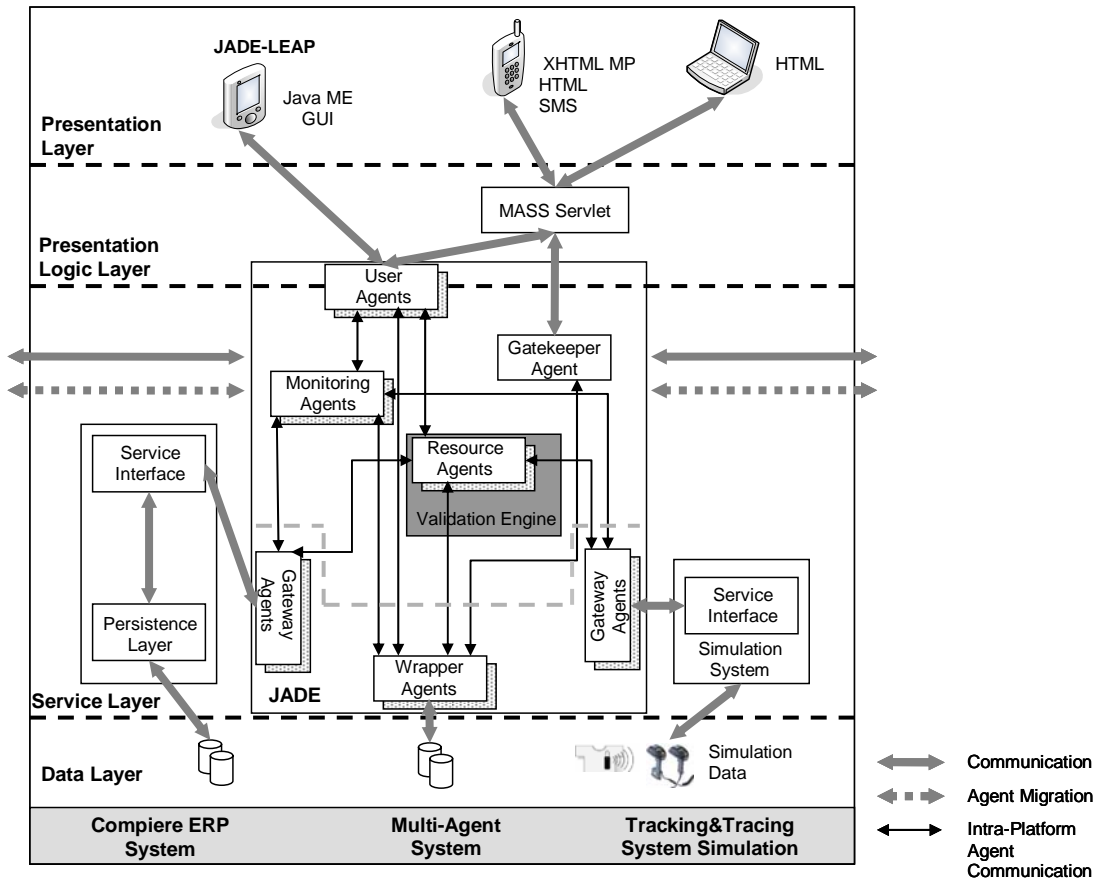


Figure 1. Architecture of the system

The top layers of the architecture were designed in such a way that users can access the SCEM system no matter what type of device they have. The *presentation logic layer* transforms content generated by the service layer into a human readable and device dependent form. The MASS Servlet acts as an additional intermediary between the user agent and the client's web-based device.

The *presentation layer* contains GUI software like browsers and Java GUIs displaying content on devices such as smart phones, PDAs, notebooks and others.

Tracking data are stored directly in the file system. A simulation module generates the tracking entries for active transportation orders.

Agents accessing the underlying systems and analyzing events are hosted on an agent platform, Java Agent Development Framework (JADE) [3]. Jadex, a reasoning engine for software agents [16], is used for implementation of the resource agents. The GUI on Java enabled mobile devices is provided by the JADE Lightweight Extensible Agent Platform (LEAP) [4]. Mobile devices with browsers that support HTML or XHTML are used when available.

SIMULATIONS

Discrete-event simulation was employed to evaluate the performance of MASS and to analyze its behavior. Computer based methods are usually evaluated with respect to the time needed for solving the underlying problem and the quality of the solution. In our case, time (i.e. the time to respond to a problem) and quality (i.e. the quality of the diagnosis) are very important criteria. When it takes longer to detect a problem, more follow-up problems are then likely to occur further down the supply chain. For example, in the Just-In-Sequence concept [10], unfinished goods must arrive right on time to guarantee the correct sequence of operations. Even a minor delay of one minute may require total or partial re-sequencing. Short response times from MASS are also appreciated in order to be able to monitor and analyze the impact of counter-measures taken.

Regarding the quality of the diagnosis, faultless detection of unforeseen events is important. In addition to target-state comparisons, we examined if statistical tests could further enhance diagnosis quality. In the ERP system, optimal routes starting from the warehouse to the delivery destination are stored. In reality, road conditions may require re-routing, and inaccurate data from mobile tracking systems may show deviations from the optimal routes. These differences between actual and optimal routes were then simulated. Here, statistical tests can further be used to assess the significance of the deviations. A chi-square test was chosen to ensure that the deviations from the optimal route were distributed normally. If this was not the case, it could be concluded that the deviations are significant. This information could then be used to define thresholds for notifying the process owner.

The scenario underlying the simulation is from the bicycle industry. A fictitious supply network was modeled in which each node is represented by the partner's ERP system. In the simulation, agents automatically change transaction data through pertinent web services. A tracking simulation is assigned to each running transportation order. The simulation system generates a starting point and an endpoint for the order. Then it calculates the route and simulates the transport. Software agents access tracking data and pertinent ERP data to monitor and analyze the orders.

To allow comparisons of the simulations, the following preconditions were set:

- Each agent monitors the same order from the ERP system.

- A fixed dataset of 258 tracking points and a fixed route with 5 nodes are provided.
- All agents run on one computer (AMD Athlon XP 1800+ with 512 MB RAM). The ERP system and the T&T system are hosted on a server (Dual AMD Athlon MP 2400+ with 2 GB RAM) and accessed by web services.
- In one simulation run, all agents use the same algorithms for monitoring and analysis.
- Only “request – inform” speech acts are allowed; i.e. no further negotiations between agents take place.

Evaluating Time

In a supply network, many transport orders are active at the same time. The MASS has to monitor all orders concurrently and still provide the user with up-to-date information. Therefore, larger order numbers result in more agents running simultaneously in order to monitor the orders. The focus of the simulation was therefore on how the system performs with increasing number of monitoring agents. The time between the start and the end of the diagnosing process was measured.

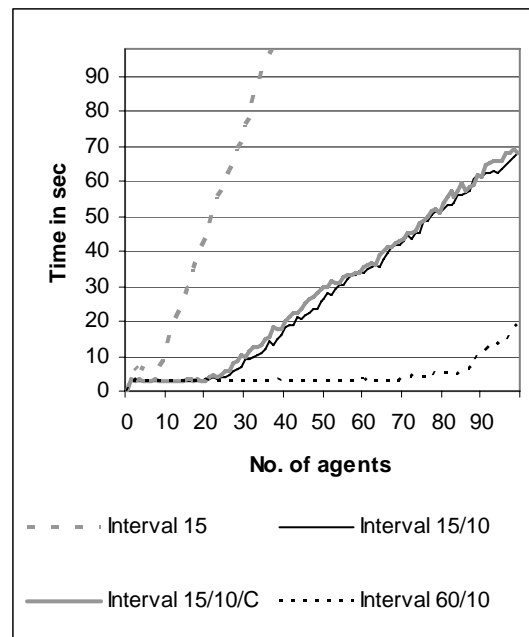


Figure 2. Response time vs. number of agents

Figure 2 shows the results of four simulation runs. Average response times are plotted against the numbers of agents involved. The term “response time” refers to the real-time delay the MAS takes for recognizing and diagnosing an event. (We do not mean the time the user has to wait for a response at

his or her computer. The user will get the response immediately.)

In the *Interval 15* curve, an agent renews data every 15 seconds. The curve shows a steady increase in response time from 9 agents on. When 25 agents access the underlying systems simultaneously, the process requires more than one minute, which is too long in many cases. The limiting factor is accessing the ERP system, in particular database calls to retrieve order details such as quantity and delivery dates.

Since ERP data do not change as rapidly as tracking data, better performance can be achieved by limiting the number of calls to the ERP system. This effect is shown in the *Interval 15/10* curve. ERP data are checked only every tenth time tracking data are read. 88 agents can now be active before the response time exceeds one minute.

In the *Interval 60/10* curve, the T&T system is checked every 60 seconds and the ERP system is accessed every tenth time the tracking system is checked. Here 152 simultaneous monitoring requests can be processed before the response time exceeds one minute. The increase is linear from the 84th agent on. Not shown in the figure is an exponential increase of response time from the 307th agent on. This is due to the limitation of the Java VM heap space.

Finally, the effect of simultaneously applying statistical tests (chi-square tests) in addition to the target-state-comparisons was measured. The results are shown in the *Interval 15/10/C* curve. Obviously the additional computing effort is minimal as compared to the *Interval 15/10* curve. This means that it is worth trying to enhance the results by chi-square tests.

Quality Aspects

The quality of a diagnosis is high if the initial problem is correctly determined. Since we try to detect problems automatically, the question arises which types of problems can be found and treated in this way at all. Automation is possible for routine problems that can be treated through routine actions based on logical considerations. Beyond the scope, on the other hand, are problems involving irrational factors, emotions and intuition.

Based on the target-state comparisons, the thresholds can be customized without restrictions. Therefore, it is possible that deviations from the plan will not be recognized or that unnecessary notifications will be sent to the users. In our simulations, all disruptions

entered manually to the T&T module were detected. Nevertheless, some problems are hard to find. Examples are problems that build up when smaller problems have not been recognized, or problems for which sufficient data are not available (e.g. in the case of a disaster).

Another problem is the detection of measurement failures. Through the target-state comparisons it is possible yet cumbersome to eliminate single outliers. Being susceptible to outliers, the chi-square tests are more problematic. Since the chi value used is relatively small and two values (mean, standard deviation) are estimated, the null hypothesis of having a normal distribution is easily rejected. The conclusion is a major deviation from the planned route although in reality only a minor problem may have occurred. Therefore, better statistical tests should be applied in the future.

IMPLICATIONS

The presented approach for diagnosing events in supply networks has several benefits:

- It helps to improve transparency in networks.
- Information can be accessed through mobile devices from anywhere therefore supplying process owners with up-to-date information.
- Diagnosis is faster and therefore problems are detected quicker.
- Process owners save time because they do not need to detect the problems themselves.

Another advantage comes from the fact that the standard deviation has already been calculated. Likewise, the data can be given to controlling or performance measurement systems for variance analyses. For example, minimal differences between the planned and the actual routes do not lead to a notification of the process owner, but many small ones can add up and increase the transportation cost (e.g. gasoline consumption). With variance analysis, this can be detected and considered in future route planning.

As the simulations have shown, only a limited number of agents can run on a single computer. However, an advantage of agents is that they can be distributed (federated) over multiple hardware resources. This allows upscaling to thousands of parallel orders. User agents can distribute monitoring agents over available hardware resources (load balancing). Another option is varying the time intervals in which the T&T and ERP systems are accessed. For example, when the number of agents increases, the

number of queries to the underlying systems may be decreased.

One precondition for achieving the benefits of automated problem detection should be mentioned: All partners of the supply network must trust each other and be willing to share their data.

CONCLUSIONS AND FUTURE WORK

This paper discussed agent-based diagnostics in supply networks using new technologies such as Auto-ID and mobile communication. Through partial automation of the diagnosis, process owners do not need to examine all problems themselves. Software agents gathering data along the supply chains and checking planned data against actual data provide human-users with relevant information for problem recognition and problem solving. In the future, extending the number of measurements and further statistical analyses are expected to improve the diagnosis.

Current research activities are aimed at enhancing the system with additional therapeutic capabilities. For example, detecting a fatal crash could lead to a new transportation order or perhaps a new production order initialized by software agents.

While the technologies we used are readily available, security and privacy issues need further research. In order to apply the presented approach in an enterprise setting, privacy concerns regarding RFID and the trust issue regarding mobile agents have to be solved.

REFERENCES

1. Alvarenga, C.A. & Schoenthaler, R. C. (2003). A New Take on Supply Chain Event Management. *Supply Chain Management Review*, 7 (2), 28-35.
2. Backlund, P. (2005). On the research approaches employed at recent European Conferences on Information Systems (ECIS 2002 – ECIS 2004). *Proceedings of 13th European Conference on Information Systems* [on CD], Regensburg.
3. Bellifemine, F., Caire, G., Poggi, A. & Rimassa, G. (2003). JADE – a white paper. *exp - in search of innovation*, 3 (3), 6-19.
4. Berger, M., Rusitschka, S., Schlichte, M., Toropov, D. & Watzke, M. (2003). Porting agents to small mobile devices - the development of the Lightweight Extensible Agent Platform. *exp - in search of innovation*, 3 (3), 32-41.
5. Bretzke, W.-R. & Klett, M. (2004). *Supply Chain Event Management als Erfolgsfaktor für Outsourcing-Prozesse* (in German). Available: <http://www.bvdp.de/index.htm?/files/logistik/4E5B99263AEB46E4856CE31FCA5D653D.htm>
6. Bose, I. & Pal, R. (2005). Auto-ID: managing anything, anywhere, anytime in the supply chain. *Communications of the ACM*, 48 (8), 100-106.
7. Compiere (2007). *What Compiere Does*. Available: <http://www.compiere.org/about/whatitdoes.html>.
8. Dangelmeier, W., Pape, U. & Rütter, M. (2004). *Agentensysteme für das Supply Chain Management: Grundlagen, Konzepte, Anwendungen* (in German). Deutscher Universitäts-Verlag, Wiesbaden.
9. Hanebeck, H.-C. L., Tracey, B. (2003). The role of location in supply chain management. *International Journal of Mobile Communications*, 1 (1/2), 148-166.
10. Harrington, L. (2007). Change Drivers: Navigating the New Auto Supply Chain. Available: http://www.inboundlogistics.com/articles/features/0207_feature02.shtml.
11. Hendricks, K. B. and Singhal, V. R. (2003). The effect of supply chain glitches on shareholder wealth. *Journal of Operations Management*, 21 (5), 501-522.
12. Kärkkäinen, M., Främling, K. & Ala-Risku, T. (2003). Integrating Material and Information Flows Using a Distributed Peer-to-Peer Information System. *Collaborative Systems for Production Management*, Boston, 305-319.
13. Lee, H.L. (2004). The triple-a supply chain. *Harvard Business Review*, 82 (10), 103-112.
14. Luck, M., McBurney, P., Shehory, O. & Willmott, C. (2004). *Agent Technology Roadmap* [online]. Available: <http://www.agentlink.org/roadmap/roadmapreport.pdf>.
15. Nimis, Jens & Stockheim, Tim (2004). The Agent.Enterprise Multi-Multi-Agent System. *Multi-Conference on Business Information Systems (MKWI2004)* [online]. Available: <http://www.wiiv.de/publikationen/TheAgentEnterpriseMultiMulti713.pdf>.
16. Pokahr, A., Braubach, L. & Lamersdorf, W. (2005). Jadex: A BDI reasoning engine. *Multi-Agent Programming*, Springer, Berlin, 149-174.
17. Weber, K. (2006). The Next Step: Privacy Invasions by Biometrics and ICT Implants. *Ubiquity*. [online], 7 (45). Available: http://www.acm.org/ubiquity/views/pf/v7i45_weber.pdf.
18. Zimmermann, R. (2006). *Agent-based Supply Network Event Management*. Basel: Birkhäuser Verlag.