An Approach to Find Integration and Monitoring Points for Container Logistics Business Processes

Tugkan Tuglular, Dilek Avcı, Şevket Çetin
Izmir Institute of Technology
Izmir, Turkey
{tugkantuglular,dilekavci,sevketcetin}
@iyte.edu.tr

Gökhan Dağhan, Murat Özemre
Bimar
Izmir, Turkey
{gokhan.daghan,murat.ozemre}
@bimar.com.tr

Tolgahan Oysal
Yekare Consulting
Izmir, Turkey
tolgahan.oysal
@yekare.com

***Abstract*— An end-to-end intermodal container transportation usually requires business processes of various enterprises to work together in a fast, accurate, reliable and secure way. This can be achieved with integrated business services, which are monitored constantly for the above mentioned quality objectives. Although there is academic research currently available for the general topic, to the best knowledge of the authors of this paper, no environment is defined and implemented for container logistics business that provides an integration and monitoring framework for business processes of the sector under consideration. One of the steps in the specification and development of such a framework is to seek points for integration and monitoring. This paper discusses an approach to find integration points for container logistics business processes as well as monitoring those points of integration.**

Keywords-business process; integration; monitoring; container logistic.

I. INTRODUCTION

Although container logistics business is composed of well-defined services, such as loading/unloading containers from ships/trailers/trains, storing containers in depots, etc., which are handled by various companies, due to competition the very same goods traveling the same route may be stored in different depots or carried by different shipping companies. With intermodal transportation, the number of possibilities for carrying a container from one point to another increases dramatically as shown in Figure 1. With intermodal container transportation getting more support from the European Union, and therefore, introducing reduced costs, the pressure on enterprises to use different transportation modes, such as sea, land, and rail, is increasing constantly. Under these circumstances, the necessity for integrated business services is unavoidable for companies in logistics business and for customer satisfaction, which is usually defined with service level agreements (SLAs), monitoring of integrated business services becomes crucial.

Currently, integrations shown in Figure 1 are real and running among heterogeneous applications, which means that different processes in different business units are supported by different applications [1]. This indicates that different semantics for the data to be exchanged exist in the container logistics environment. For instance, every country has its own customs regulations. Even though they stem from trade union regulations, which introduce some standardization, it is almost impossible to find two countries having exactly the same regulations. In spite of this fact, in our case, all transportations are of container type and there are EDIFACT [2] message standards for containers, such as CODECO (container gate-in/gate-out report message) [3], COPARN (container announcement message) [4], COPRAR (container discharge/loading order message) [5], etc. That means some standardization exists at data integration level for some of the container logistics business processes. However, our observations show that a similar standardization is not valid for Enterprise Resource Planning (ERP) systems or financial systems that are part of container logistics business.

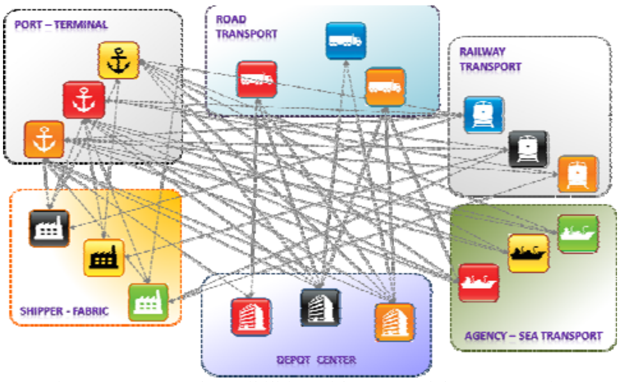

Figure 1. Increased Possibilities with Intermodal Transportation

With the realities explained above in mind, we propose a method based on "need to know" analysis and formal concept analysis supported by semantic similarity analysis along with ontology analysis to find integration points for container logistics business processes. After integration points are found and decision on integration implementation is made, we utilize audit, control and monitoring design patterns to achieve quality objectives set by SLAs. These two methods in action constitute the novelty of our approach. The next section explains our approach to find integration and monitoring points for container logistics business processes with a running example of port-agency integration of gate in-out, which coordinates container entry or exit to/from port.

II. APPROACH

In Bimar case, there are 180 integrations running on Microsoft's BizTalk server. These integrations are among different actors, such as COPARN integration between port and agency as well as between agency and depot. Business processes (BP) of these integrations are modeled using swimlane diagrams. These swimlane diagrams show the integration points for existing integrations, such as given in Figure 2, which shows integrations among Port, Agency, and Depot using EDIFACT message standards for containers. For instance, booking BP of agency (software) triggers and supplies necessary information in COPARN message format to booking BP of depot (software). This is an integration of agency-depot and booking is an integration point between agency and depot.

The “need to know” principle means that access to the information must be necessary for the conduct of one's official duties [6]. In other words, data or information should be entrusted to those who must have knowledge of it for its necessary usage. If you go backwards from usage to the need, then you may discover what is needed in order to start, continue or complete a business process. In case of the running example, port needs to know about each entering container, whether it is empty or not, whether it contains dangerous goods or not, whether it contains frozen goods or not. Depending on this information, which is supplied by the agency-port integration, port software will trigger corresponding business processes possibly through another integrations or web services.

The purpose of formal concept analysis (FCA) [7] is to support the user in analyzing and structuring a domain of interest. Such a method allows us to automatically obtain similarity scores without relying on human domain expertise [8]. Here, we plan to apply FCA to a domain knowledge, which includes inputs and outputs of each business process in that domain, to discover similarities of input and output, which will indicate an integration point. In case of the running example, imagine that containers with frozen goods are stacked in a special storage area at the port. The stacking BP and gate-entry BP as part of the port domain are represented in the port domain knowledge with help of corresponding ontologies. When FCA is applied to port domain knowledge, a high similarity score between “frozen goods” output in gate-entry BP and “frozen goods” input in stacking BP will be found an it may indicate a possible integration between to business processes. We call this operation as integration discovery.

We also plan to apply FCA to existing integrations to reduce number of integrations. Similarities of input among existing integrations will be searched. High similarity of inputs belonging to two or more integrations may mean that those integrations can be combined. In case of the running example, triggering stacking BP and inspection of frozen goods BP can be combined into a single integration with similar input. We call this operation as integration reduction.

It is assumed that an integration point is also a monitoring point using the anology of a local area network connected to Internet through a router (integration point) with a firewall on it (monitoring point). What to monitor, to what depth, and how frequently depends on the SLA defined for the integration. In case of the running example with agency-port integration, which informs the port about the entering trailers with containers, trailer stops for a while and this stop causes a trailer queue at the port entrance, which causes accumulation of empty containers at the port, which may also cause lack of space for unloading. Such an integration point should be monitored heavily, meaning that different views should be compiled from different sources and rules should be checking these views for anomalies.

The audit, control and monitoring design patterns (ACMDP) introduced by Trad and Trad [9] is based on flow-encoded switching design pattern and supported by Observer Design Pattern [10], Model-View-Controller Design Pattern [10], and the Decision Making Design Pattern [11]. ACMDP design patterns suggest factors and views. Factors determine characteristics and quality status of the phenomena under observation whereas a view is defined as a set of one or more factors. By using ACMDP approach and Microsoft’s Business Activity Monitoring (BAM) tool, we plan to add monitors and rules to the integrations, to observe integrations through BAM portal and to send e-mail and SMS notifications to the people on duty.

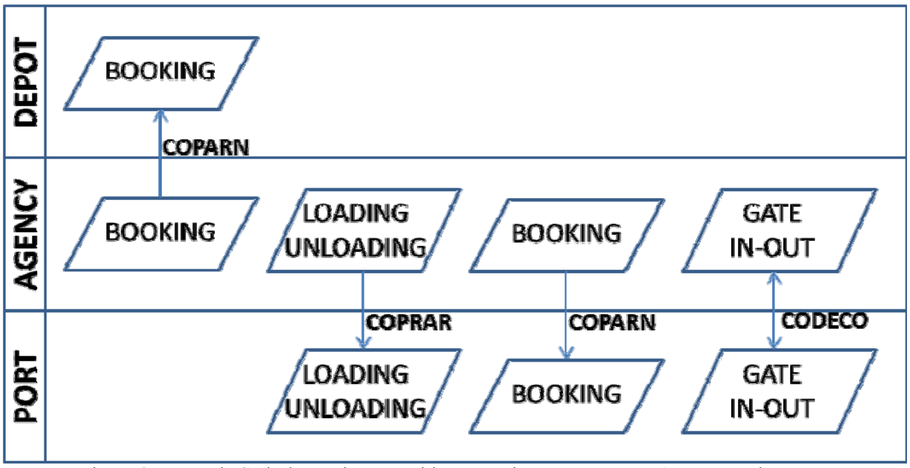

Figure 2. Example Swimlane Diagram with Integrations among Port, Agency, and Depot

There may be integration requests coming from customers, who know what is needed for the business goal and have a rough idea from which business process the necessary data can be taken. In this case, we plan to use the "need to know" analysis method proposed in this paper to validate the rough idea of the customer. In some other cases, the customer may not have an idea about what is necessary for the business goal and which business process can provide necessary data. For those cases, we plan to use "need to know" analysis and similarity analysis method as explained with a running example above. Those cases will show the benefit of the method.

III. PLANNED FUTURE WORK

At the moment, we apply this approach to the integrations developed and handled by Bimar to find integration points, and therefore, monitoring points. The next step will be finding similar integration points using FCA method, so that number of integrations can be reduced along with monitoring points. Furthermore, we plan to collect not only business data but also data exchanged among various levels of software, database, network, and operating systems, from monitoring points and analyze it to define largest possible set of monitors be used in the configuration of monitoring parameters as well as in the service level agreements.

Later, we plan to develop an integration and monitoring framework for container logistics business processes. Currently, all integration development and monitoring configuration are performed manually by drag and drops, connecting two points and writing expressions. The goal is to replace manual operation with code development, so that the process can be improved, for instance, by inclusion of unit testing. Since libraries, APIs, and frameworks are needed for code development, an integration and monitoring framework for container logistics business processes will be developed.

ACKNOWLEDGMENT

This work is supported by SAN-TEZ 00933-STZ.2011-1 Research Grant.

IV. CONCLUSION

A combined approach to find integration points and monitoring points for container logistics business processes is introduced. The proposed approach is composed of "need to know" analysis and formal concept analysis. After integration points are found and reduced respectively, the resulting set integration points are also accepted as monitoring points. Monitoring requirements will be determined by using the approach set forth by the audit, control and monitoring design patterns and agreed upon service level agreements. An integration and monitoring framework for container logistics business processes will be developed to enable coders to implement monitoring requirements efficiently.

REFERENCES

[1] T. Puschmann and R. Alt, "Enterprise Application Integration - The Case of the Robert Bosch Group", Proceedings of the 34th Hawaii International Conference on System Sciences, 2001.

[2] UNECE, "The United Nations rules for Electronic Data Interchange for Administration, Commerce and Transport", http://www.unece.org/cefact/edifact/welcome.html, last accessed on July 8th, 2012.

[3] UNECE, "UN/EDIFACT, Message Type: CODECO", http://www.unece.org/trade/untdid/d00a/trmd/codeco_c.htm, last accessed on July 8th, 2012.

[4] UNECE, "UN/EDIFACT, Message Type: COPARN", http://www.unece.org/trade/untdid/d00a/trmd/coparn_c.htm, last accessed on July 8th, 2012.

[5] UNECE, "UN/EDIFACT, Message Type: COPRAR", http://www.unece.org/trade/untdid/d00a/trmd/coprar_c.htm, last accessed on July 8th, 2012.

[6] Wikipedia, "Need to know", http://en.wikipedia.org/wiki/Need_to_know, last accessed on March 21st, 2012.

[7] B. Ganter and R. Wille, "Formal Concept Analysis: Mathematical Foundations", Springer, Berlin, 1999.

[8] A. Formica, "Concept similarity in Formal Concept Analysis: An information content approach", Knowledge-Based Systems, 21(1), pp. 80–87, 2008.

[9] A. Trad and C. Trad, "Audit, Control and Monitoring Design Patterns (ACMDP) for Autonomous Robust Systems (ARS)", International Journal of Advanced Robotic Systems, 2(1), pp. 25-38, 2005.

[10] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. "Design Patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley, 1995.

[11] A. J. Ramirez and B. H.C. Cheng, "Design Patterns for Developing Dynamically Adaptive Systems", Proceedings of SEAMS '10, May 2-8, 2010, Cape Town, South Africa.

 ISBN: 978-1-61208-215-8