International Journal of Advanced Computer Science and Information Technology (IJACSIT)
Vol. 2, No.1, February 2013, Page: 1-13, ISSN: 2296-1739

www.elvedit.com

Hierarchical Structural Driven Model for Integrative Information Management Architecture

Authors

Mohd Izzuddin Mohd Tamrin
Department of Computer Science,
International Islamic University Malaysia
izzuddin@iium.edu.my
Gombak, Malaysia

Tengku Mohd Tengku Sembok
Department of Computer Science,
International Islamic University Malaysia
tmts@iium.edu.my
Gombak, Malaysia

Mira Kartiwi
Department of Information System,
International Islamic University Malaysia
mira@iium.edu.my
Gombak, Malaysia

Abstract

Heavy reliance on the other nodes in a complex supply chain network and failure to deliver the specialized assignment by merely one node can caused disruption to the entire processes. The authors introduce the Integrative Information Management Architecture (IIMA) to generate semi-automated assistance against common supply chain deviations. Support models are used and they are made from tree-like structure in XML documents. The recommendations are stored in the leaf nodes whereas the intermediary nodes are filled with characteristics of the deviation. The IIMA employed the XPath together with a series of questions to navigate through the relevant path. We had developed the prototype of the IIMA but run it in simulated environments. The results showed that the group with IIMA support improved the number of completed processes compared to without the IIMA support.

Key Words

Intermediary processes, semi-automated case base reasoning, XML/XPath driven support models.

I. Introduction

The supply chain composed of important nodes in a complex network working together to deliver the required services to the client. The complex network of the supply chain required the good management in order to deliver the right services at the minimum costs. The lack of integration between the nodes can deterred the ability of the supply chain group to react appropriately in timely manner to the problems that impinged the current business supply chain activities. The integration issues are inaccurate reflection of physical world in the servers [1] and the visibilities on the operations of their partners are often blurry [2, 3]. The systems worked in isolation and there is limited communication between these systems across the supply chain which exacerbate the situation even further [4].

In this paper the author introduced an Integrative Information Management Architecture (IIMA) to provide semi-automated assistance to address these integration challenges. Firstly, the data retrieved from the RFID networks across the supply chain will be transformed into process related information to be matched against the models. Secondly, the support models are utilized by the IIMA to assist the group members in identifying the problems, searching through the solutions tree and adhering to the process constraints. The prototype of the IIMA is developed and tested in simulated environments for the evaluation of its performance.

A scenario application created for the IIMA which is based on supply chain group working with their client to deliver services [5] is incorporated in the simulation environments for evaluation purposes. The results show that the supports given by the IIMA is stable in two different settings for two different types of the supply chain problems. In the following section, we are going to describe the related works and the organization of the IIMA. Next, the formulization of the support models and the collaboration algorithm that governed the interaction between the components of the IIMA will be presented. Lastly, we show the evaluation of the IIMA prototype, results and analysis before proceeding with the conclusion.

II. Related Works

The mean time architecture [6] utilized data from the RFID network and extracted the time segment. Every entry and exit of the vicinities is marked and the time it took an item to move from these points are recorded for determine any bottleneck from the processes. This system does not reveal the activities of the supply chain. The customization system [7] in contrast employed a more complex analysis of the data by introducing the concept of network coefficients which could identify the type of activities from the inflow and outflow traffic. The model based monitoring system [8] on the other hand produce much finer details of the information at the process level with the creation of constraints based component model.

Our approach is to take a simpler approach but generate process level information by associating the data with customizable problem model based on the local context. Similar to this line of work, the traceability system architecture [9] also had utilized the data from the RFID network of readers and manipulated them to provide information about the relationship between tagged items with the complete history tracking. In comparison, our IIMA provide additional information on the constraints of the processes available from the information model.

The dynamic logistic process system [10] extracted data from the RFID network and employ software agent to identify the current activities and determine the standardize processes to be followed by the workers. Similarly, the real time collaboration architecture [11] utilized agent technology to negotiate on the behalf of members from the supply chain group for the execution of shared processes. The technique used by this system is quite similar to [10] but on a broader scope. The authors had taken simpler approach for the IIMA to assist the appointed management team in making decision. This is accomplished by giving recommendation and validation on cases through the case and information handler, two prominent sub components of the IIMA.

The service based shop floor architecture [12] employed services provided from the machines remotely in order to reconfigure the manufacturing processes at will and support for frequent requirement changes in the market. The collaborator system [13] in contrast provided the mean for collaboration between the supply chain members with integrated portal in much broader scope instead focusing on limited manufacturing processes. A global planning is created to coordinate the supply chain activities across the group and relevant web portlets are invoked. In comparison, the IIMA wrapped the functionalities from the existing systems and offered them as services instead of creating these services from scratch portal which is similar to the approach employed in [14].

The bio-disease tracking system [15] on the other hand, offered additional functionalities beside the functionalities from the existing systems such as the estimation of the tracked objects are affected by disease. The IIMA in comparison offers the necessary functionalities for detecting process deviation as RFID services by manipulating the data generated from RFID readers.

III. ORGANIZATION OF ARCHITECTURE

The IIMA initiate an intermediary processes between the RFID network of readers and the existing systems. The following describe the six main components of the IIMA [16]:

System Communicator: interacts with the local existing supply chain systems and provides their functionalities as services to the IIMA

RFID Service Provider: transforms basic data retrieved from the local RFID readers such as tag identification, time stamp and reader identification into process level information. Different types of data manipulation are provided as different types of RFID service.

Service Handler: serves as the middleman between the subsystems and the main system. The local services that are scheduled by the main system will have to be passed to the respective local service handlers for execution. Moreover, it is responsible to detect deviation in the current activities by comparing the events recorded by the RFID service provider against the instructions from the main system.

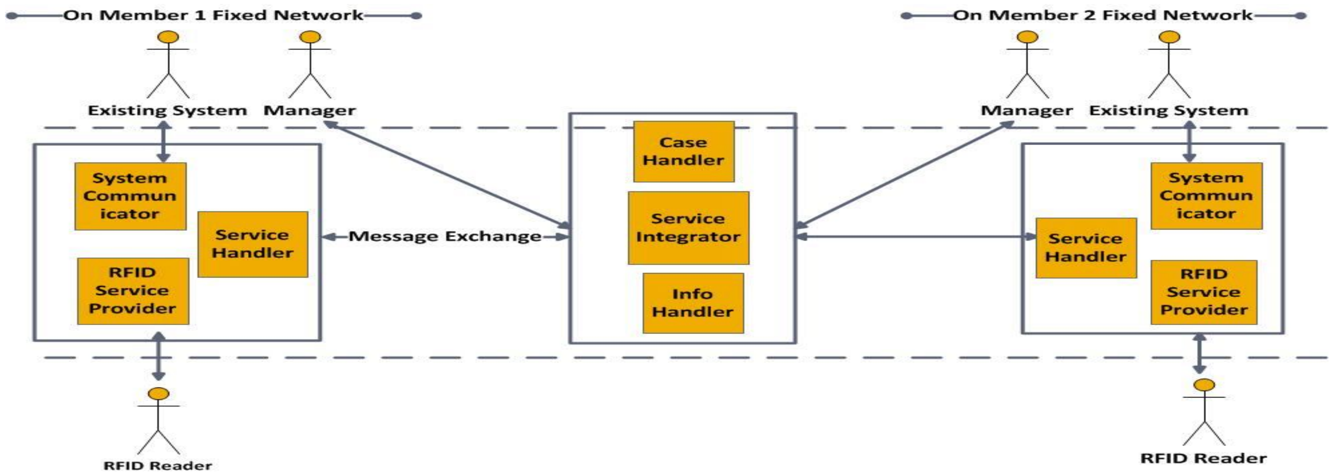

FIGURE I: ORGANIZATION OF IIMA.

Service Integrator: serves as the middleman between the main system and the local subsystems. It is responsible to initiate the process restructuring procedure. The case handler and the information handler are sought to assist the appointed team management with the alternative process and to check the process constraints before it can proceed for execution.

Case Handler: responsible to produce recommendation for the team management in response to the problems detected earlier. It holds many potential cases that are known to be the solutions for the common deviations in the supply chain activities.

Information Handler: responsible to check process constraints on the recommended case. Usually, the information handler performed a background check on the member that is responsible for the execution of the process. Mainly it involved checking the job scope of that member and accessibility of the information required by the services offered for execution.

IV. FORMULIZING SUPPORT MODELS

The support models encompass problem, case and information models utilized by the service handler, case handler and information handler respectively. In this section, we described the elements used in these support models and their hierarchical relationships.

Definition 1: Let IN, I, EV, L, TM, S, P, SL, DT, DL, G, M, R, A, IC be sets of elements existed in the support models. The following are the definition of the sets:

IN = {in | in is input retrieve by system}
I = {i | i is instruction from existing system}
EV = {ev | ev is event from RFID network}
L = {l | location from event}
TM = {tm | time from event}

S = {s | suggestion made by system}
P = {p | problem identify by system}
SL = {sl | solution identify by system}
DL = {dl | difference in the instruction and event location}
DT = {dt | difference in the instruction and event time}
G = {g | supply chain group}
M = {m | member from the group}
R = {r | relationship establish in the group}
A = {a | action of the member}
IC = {ic | information required by member}

Definition 2: The supply chain group comprised of the selected members. Each member established specialized relationship in the group by providing specific set of actions and required specific set of information:

$M \subseteq G \cap R \subseteq G$
$A \subseteq M \cap IC \subseteq M$

Definition 3: The system input comprised of event from the RFID reader and the instruction from existing system:

$EV \subseteq IN \cap I \subseteq IN$

Definition 4: The event and instruction are made up from location and time of recorded tag item and recorded instruction respectively:

$L \subseteq EV \cap TM \subseteq EV$
$L \subseteq I \cap TM \subseteq I$

Definition 5: The suggestion provided by the handlers comprised of problem identification or alternative solution:

$P \subseteq S \cup SL \subseteq S$

Definition 6: The problem comprised of difference in time and location between the event and instruction:

$DL \subseteq P \cap DT \subseteq P$

Definition 7: The solution comprised of alternative actions:

$A \subseteq SL$

V. COLLABORATION ALGORITHMS

The collaboration algorithm described the interaction between the IIMA components. These algorithms are designed to undergo six main phases. However, in this section the authors give more emphasized on the behavior of the handlers utilizing the support models. For this reason, only two related phases from the six phases are highlighted.

A. Deviation Phase

Two of the prominent sub-components of the service handler are the event manager and deviation alertor. Figure 2 described the high level algorithm between the service integrator and the local service handler. The algorithm started with event manager receiving list of events (LE) from RFID service provider to initiate the process of deviation checking in current processes. The event manager passed the LE and list of instruction (LI) to the deviation alertor. Based on definition 3, LE and LI correspond to the EV and I respectively. Both of these lists made up the system inputs, IN. The deviation alertor is responsible to follow the logic of the problem model via the "navigateModel" method call. Figure 3 shows the snippet of problem model. The authors have evaluated three approaches to retrieve the correct type of problem: content filtering [17], collaboration filtering [18] and hybrid filtering [19].

```
Variables: LE, LOR, LP, LI, clientMain, pw, hostName, portNum
SubSystem:
while(curEvent == false):
    evmObj = new EventMgr();
    LOR = evmObj.detDeviation(LE);
      if (LOR == true), then:
        clientMain = new Socket(hostName, portNum);
        pw = new PrintWriter(clientMain.getOutputStream(), true);
        pw.println(LP);
ServiceHandler:
problemDoc = docBuild.parse(in);
daObj = new DevAlertor();
  while (endFlag == false) :
    xPath = daObj.nagivateModel(LE,LI,val);
      if(xPath.equals("END"))
        endFlag = true;
     else:
        nIterate = XPathAPI.selectNodeIterator(problemDoc,xPath);
        nPoint = nIterator.nextNode();
        val =  nPoint.getFirstChild().getNodeValue();
```

FIGURE 2: HIGH LEVEL ALGORITHMS FOR DEVIATION PHASE

The authors adapted the content filtering technique to match the input from existing system (i.e., transform into LI) and RFID reader (i.e., transform into LE) against the characteristic of the problem in the form of differences in time and location. Based on the definition 6, the characteristic of the problem correspond to P which made up from the differences in time and location, the DL and DP. The problem model is created in XML documents because of the unique XML characteristic, namely platform independent, which ease with the modification of the models based on the context. The Document Object Model (DOM) needed to be created, the "problemDoc", which provide with the necessary API for XML manipulation. XPath is employed to navigate through the document and the authors had utilized the Xalan package to evaluate the XPath expressions.

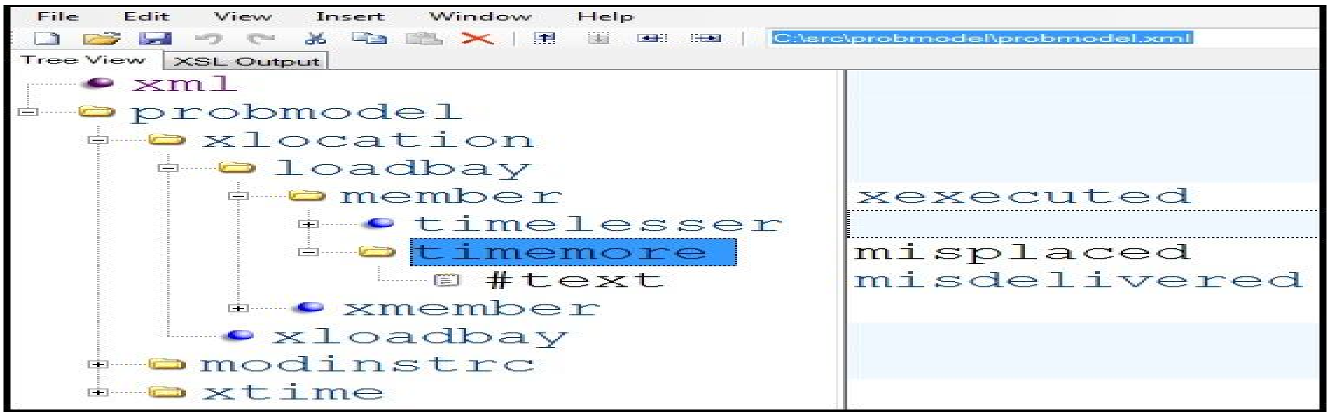

FIGURE 3: SNIPPET OF PROBLEM MODEL IN XML DOCUMENT

In the "navigateModel" method call, the deviation alertor began to compare the first children in the model with the input from LI and LE. It continues with the remaining siblings and stops until there is a match. From this forward point, it braches out to the next level from the matched node. In figure 3, the first level nodes are "xlocation", "modinstrc" and "xtime". Based on the value from LE and LI, there are differences in the location and for this reason, the service hander chosen the "xlocation" node instead of the other nodes from the first level. Because this is not a leaf node of the problem model, the algorithms initiate the "navigateModel" method iteratively to navigate to the second level from the "xlocation" node until the service handler find the leaf node which suggest the problem faced by the supply chain group. By matching the inputs from the system with the characteristic of the problems from the problem model, the problem may be known.

B. Validation Phase

In the alteration phase, an alternative solution to counter the misplace problem identified from the deviation phase is stored in the list of case (LC). The continuation from the alteration phase and which is the validation phase, the LC is passed back to the case interface to proceed with the validation phase by invoking "verifyCase" method in figure 4. Based on the results of the verification stored in the list of results (LOR), the algorithm completed this validation phase with positive verification or return back to the alteration phase with negative verification. The information manager, the sole sub-component of the information handler is responsible to do a background check on the member assigned for the execution of the alternative solution. In order to accomplish this, the information handler gets the properties of the LC via the "getRequirement" method call and these properties are match with the process constraints from the information model.

```
Variables: LC, LOR
MainSystem:
while(LOR == false):
// While loop is located before entering the alteration phase
  ciObj = new CaseIntr();
  LOR = ciObj.verifyCase(LC);
Info Handler:
infoDoc = docBuild.parse(in);
imObj = new InfoManager();
caseReq= imObj.getRequirement(LC);
  while (violateFlag == false):
    xPath = imObj.navigateModel(caseReq, val);
      if(xPath.equals("VIOLATE"))
        violateFlag = true;
      else:
        nIterate = XPathAPI.selectNodeIterator(infoDoc,xPath);
        nPoint = nIterator.nextNode();
        val =  nPoint.getFirstChild().getNodeValue();
```

FIGURE 4: HIGH LEVEL ALGORITHMS FOR VALIDATION PHASE

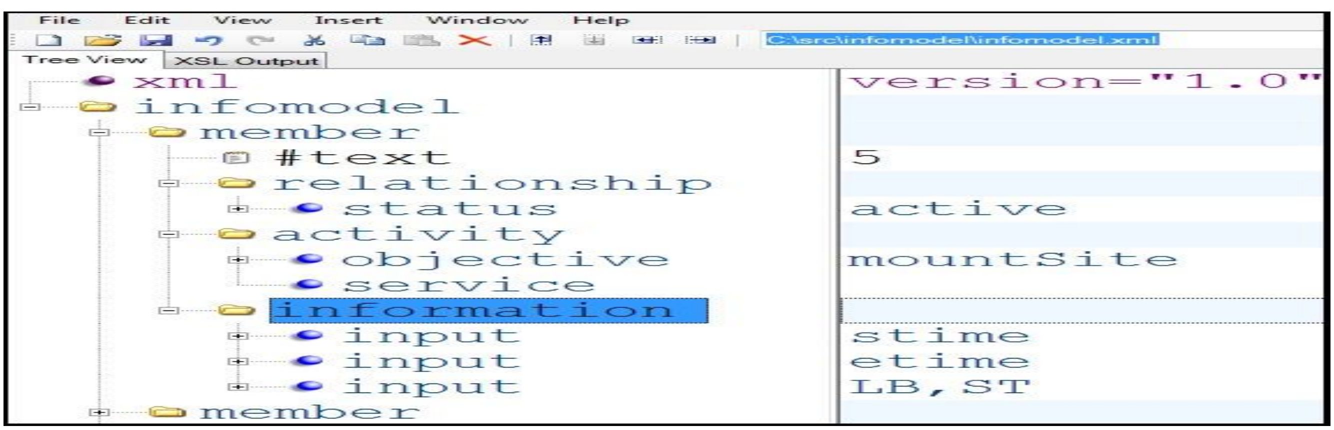

FIGURE 5: SNIPPET OF THE INFORMATION MODEL IN XML DOCUMENT

Based on definition 2, the process constraints correspond to the properties of the supply chain group in term of the selected members, their relationships, actions and information required. The "navigateModel" is invoke iteratively to navigate through the information model and information manager try matching the chosen alternative case requirements with the process constraints. If it matches for every node in the information model for member 5 as shown in Figure 5, positive results will be returned in the LOR. In this example, the information manager will check whether member 5 is still active. Next, it checks that the alternative solution is within the job scope of member 5, namely “mountSite”. Lastly, the information required for this alternative solution is within the permissible information restrictions for member 5 as defines in the information model, namely “stime”, “etime” and “ST”.

VI. SIMULATION

The prototype of the IIMA was develop and evaluated in simulated environments. For the subsystem, we have created two servers, the first server runs the RFID service provider and service handler whereas the second server runs the system communicator and service handler. These two servers are installed at the network of every member in the supply chain group. The total of servers for the subsystems across the group had reached up to 10 servers. In addition to this, we had created another two servers for the main system and the central database. The main system runs the service integrator, case handler and information handler whereas the central database stores the events generated from the local subsystems across the group. The simulated environments are described in the following:

Dimension 1: With the IIMA support. The capabilities of the IIMA have been described extensively throughout this paper.

Dimension 2: With alternative supports. Offers three other related systems to the IIMA for comparison. The first system has the capabilities to detect ongoing processes in the supply chain activities but not able to dictate the flows once the execution had taken place. In contrast, the second system has high control over the processes but created latency in the receiving the status on the ongoing processes. We had introduced three days and five days delay for system two.

Dimension 3: Customization processes. This is the processes undertaken by the supply chain group to produce e-commerce solutions to their clients. We had introduced three types of customization processes: basic, focus and full-fledge. For each type of customization processes, two different tracks are available which correspond to two different set of server capabilities.

Dimension 4: Process disruptions. We had designed two types of common process disruptions for the supply chain management namely “misplace items” and “transportation delays”. In order to test the supporting systems throughout the customization processes, these process disruptions are supposed to impinge on the early and ending stage of the processes respectively.

VII. RESULTS AND DISCUSSIONS

The number of processes completed by the support systems across three types of customization processes inflicted with the misplace items problem are not much difference; at most with merely one task as shown in figure 6. The reason for this is because the problem occurred at the end of customization processes and therefore number of task remaining for execution is down to just one task regardless of the support employed. The performance of the IIMA and system two with a three days delay are the same in term of the number of completed processes for basic and focus customization processes remained at 5 and 7 processes completed respectively. However, as the number of processes required to be executed increases as in the full-fledge customization processes, the number of completed processes by system two with three days delay reduced.

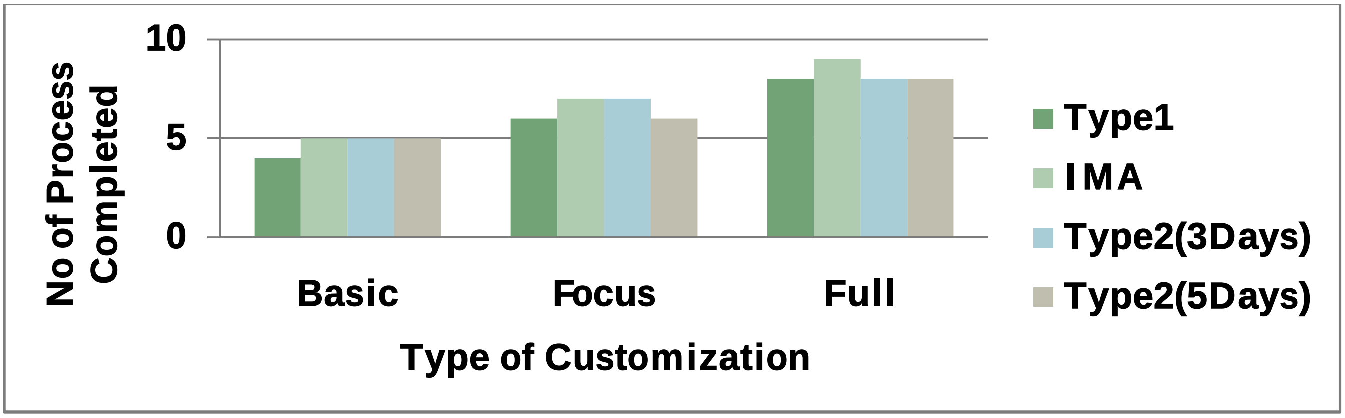

FIGURE 6: MISPLACE - NUMBER OF COMPLETED PROCESSES

This is because the safety buffer (in days) decreases proportionate to increment in the number of required processes. The situation exacerbated in system two due to the delay in reacting with the problems and cut down the number of safety buffer further depending on the number of delays. This is the reason system two with five days delay started to miss the required task earlier than system two with three days delay, in which it had taken place in the focus customization processes. Figure 7 demonstrates the percentage of outperformed processes by the IIMA over the other three systems. The IIMA performed better as the types of customization moved from basic to full-fledge; the number of systems outperformed by the IIMA increases from 1 to 3, although the number of completed tasks outperformed are small at 11 percent.

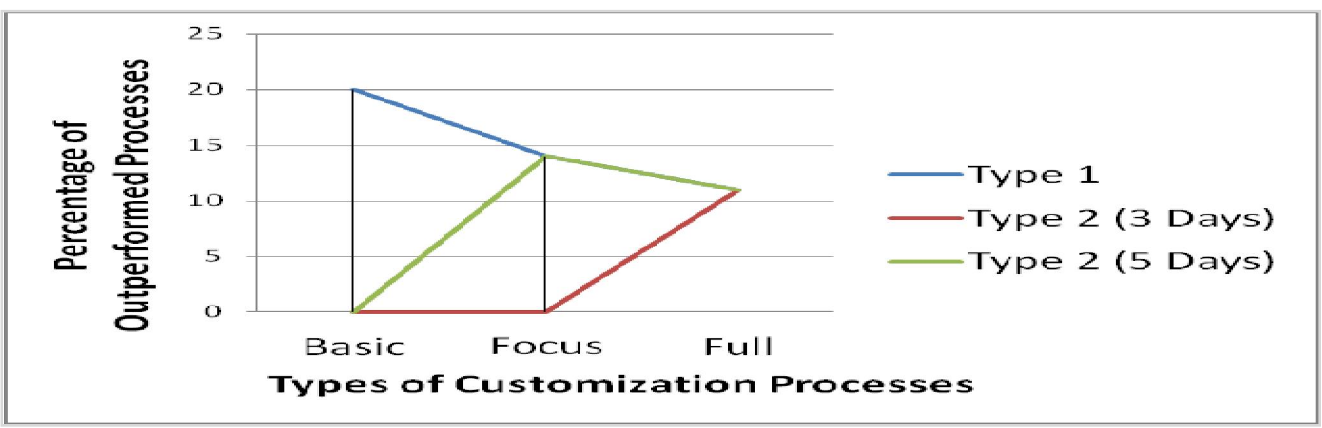

FIGURE 7: PERCENTAGE OF OUTPERFORMED PROCESSES BY IIMA

In contrast, the results look more promising for the delay transportation problem which inflicted at the early stage of the customization processes. The results will be discussed in details in our next paper which is a continuation from this paper. In order to statistically proves that the

IIMA significantly performed better than the other three systems, the Mann Whitney test was employed. The two samples for IIMA and type two with three delays based on the recorded number of completed processes in full-fledge customization processes were used in the test. Figure 8 demonstrates the results. The small p value which is less than 0.05 statistically proves that these two systems are significantly different from each other. In addition, the rank average of 30.5 and 10.5 respectively shows that the IIMA has the higher value of completed processes.

	Systems	N	Mean Rank	Sum of Ranks
Tasks	3.00	20	10.50	210.00
	4.00	20	30.50	610.00
	Total	40		

(a)

	Tasks
Mann-Whitney U	.000
Wilcoxon W	210.000
Z	-6.245
Asymp. Sig. (2-tailed)	.000
Exact Sig. [2*(1-tailed Sig.)]	.000[b]

(b)

FIGURE 8: MANN WHITNEY U-TEST BETWEEN IIMA AND SYSTEM 2 (WITH 3 DAYS DELAY) – (A) AVERAGE RANK (B) TEST STATISTIC

VIII. CONCLUSION

The intermediary processes offers by the IIMA worked silently in the background to provide supports to the customization processes against any disruptions. These problems creates financial burden to the group with very high unmet demands from the clients. The collaboration algorithm has the abilities to react accordingly and in timely manner against the common problems in the supply chain management. Together the RFID service provider and the service handler utilized the problem model to provide continuous monitoring of the ongoing supply chain processes. These are very important abilities as we have seen from the results of the simulation that the performance of system two had deteriorated as the number of required tasks increases. The situation became worst with the increase in delay from three days to five days.

Other important abilities are to assist the management team in process restructuring via the case and information handler to proposed alternative solutions and verify process constraints. The evidence from the performance of system 1 in the simulation had shaded light on the importance of these abilities. The number of uncompleted processes remained the same across different types of customization processes straight after the processes had been inflicted the common problems. Since these support models are implemented in XML documents and standard XPath expression can be employed to navigate through these models, we believed that other supply chain activities can be supported. Other activities can be supported by modifying these platform independent support models accordingly based on the needs of that specific context and later upload them into the IIMA.

REFERENCES

[1] Bowersox, D.J., Closs, D.J., & Drayer, R.W. (2005). The digital transformation: Technology and beyond. *Supply Chain Management Review,* 9(1), 22-29.

[2] Devaraj, S., Krajewski, L., & Wei, J.C. (2007). Impact of eBusiness technologies on operational performance: The role of production information integration in the supply chain. *Journal of Operations Management*, 25(6), 1199-1216.

[3] Gadde, L., & Hulthen, K. (2008). Improving logistics outsourcing through increasing buyer-provider interaction. *Industrial Marketing Management*, 38(6), 633-640.

[4] Cassivi, L. (2006). Collaboration planning in a supply chain. *Supply Chain Management: An International Journal*, 11(3).

[5] Lusch, R.F., Vargo, S.L., & O'Brien, M. (2007). Competing through service: Insight from service-dominant logic. *Journal of Retailing*, 1(83), 5-18.

[6] Delen, D., Hardgrave, B.C., & Sharda, R. (2007). RFID for better supply chain management through enhanced information visibility. *Production and Operation Management*, 16(5), 613-624.

[7] Cheung, C.F., Cheung, C.M., & Kwok, S.K. (2012). A knowledge-based customization system for supply chain integration. *Expert System with Application*, 39, 3906-3924.

[8] Huang, Y., Williams, B.C., & Zheng, L. (2011). Reactive, model-based monitoring in RFID-enabled manufacturing, *Computer in Industry*, 62, 811-819.

[9] Kelepouris, T., Pramatari, K., & Doukidis, G. (2007). RFID-enabled traceability in the food supply chain. *Industrial Management and Data Systems*, 107(2), 183-200.

[10] Chow, H.K., Choy, K., & Lee, W. (2007). A dynamic logistics process knowledge-based system: An RFID multi-agent approach. *Knowledge-Based Systems*, 20(4), 357-372.

[11] Soroor, J., Tarokh, M.J., & Shemshadi, A. (2009). Initiating a state of the art system for real-time supply chain coordination. *European Journal of Operational Research*, 196(2), 635-650.

[12] Ribeiro, L., Barata, J., & Colombo, A. Supporting agile supply chains using a service-oriented shop floor. Engineering Applications of Artificial Intelligence, 22, 6 (2009), 950-960.

[13] Cheng, J.C., Law, K.H., Bjornsson, H., Jones, & A., Sriram, R. (2010). A service oriented framework for construction supply chain integration. *Automation in Construction*, 19(2), 245-260.

[14] Tarantilis, C., Kiranoudis, C., & Theodorakopoulos, N. (2008). A Web-Based ERP System for Business Services and Supply Chain Management: Application to Real-World Process Scheduling. *European Journal of Operational Research*, 187(3), 1310-1326.

[15] Wu, C.C., Jian, M.S., & Chou, T.Y. (2011). Environment Affection Based RFID Potential Bio-Disease Tracking/Tracing System, WSEAS Transactions on Systems, 10(2), 38-48.

[16]Mohd Izzuddin Mohd Tamrin, Tengku Mohd Tengku Sembok (2012). Using Information Management Architecture to Support Flexible Supply Chain Management. International Conference on Advance Application and Technology, 26-28 Nov, ACSAT 2012.

[17]Pazzani, M.J., and Billsus, D. (2007). Content-based recommendation systems. In P., Brusilovsky, A., Kobsa, and W., Nejdl, (Eds.), The Adaptive Web, Springer, Berlin, 4321, 325-341.

[18]Schafer, J.B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In P., Brusilovsky, A., Kobsa, and W., Nejdl, (Eds.), The Adaptive Web, Springer, Berlin, 4321, 291-324.

[19]Ricci, F., Arslan, B., Mirzadeh, N., & Venturini, A. (2006). ITR: A Case Based Travel Advisory System. In S., Craw, and A., Preece (Eds.), Advances in Case Based Reasoning, Springer, Berlin, 2416, 613-627.