Analysis of supply chain policies in IT Company

Neeraj Kumar
(Associate Professor) Applied Science
IIMT COLLEGE OF ENGINEERING GREATER NOIDA UP. INDIA

Abstract :- An analysis of supply chain policies in IT Company is an information based study in which. I have concentrated on various issues. Such as Gap, aim and objectives in company, There seems to be same gap between academia and the business world concerning the treatment of the subject of SCM. The aim is to fill this gap by providing an analysis of lower level. The aim of the workflow engine is to simulate a business process model in order to check strategic decision. The man objective is to provide an execute version of a business process model (BPM). Our workflow engine does not provide validation or verification. According to need under the assumption that the provided BPM is correct. It should understand that 'start' and 'finish' junction signify and distinguish between 'and split' 'or-split' ''and joint' and 'or-joint' junction and execute them according to their definition. Various assumptions such as Simplistic Vs sophisticated treatment of process waiting time

Explicit time measurement and real time BPM execution V/s estimation of start and finish time are discussed by

me for realistic execution of BPM.

Keywords:- Business process model (BPM), Workflow engine, Start and Finish Junction, Simplistic Vs sophisticated treatment, Explicit time measurement, realistic execution.

1 INTRODUCITON

1.1 Motivation

The role of supply Chain Management (SCM) is becoming more and more important in today's business world. From a purely operational approach to SCM of the 1960s we have move to a more integrated and strategic approach. Hence, supply chain management is today considered as a source of competence and innovation. IN the modern business world, companies are competing not only through their product range and customer relations, but also through their supply chains.

In this, IT companies have been held as the "golden example" of Supply Chain Management. IT companies has achieved to become one of the most successful PC companies in the world, b y emphasizing and aligning its strategies with the design of its supply chain (SC)..

Therefore, the interest in investigating IT companies SC strategies is great, as it is expected to highlight more general and innovative issues of SCM.

12. Gap, Aim and Objectives

Even though several research efforts have examined IT companies supply chain strategies, most of the adopted approaches fall into the category of strategic and theoretical, abstract view of the subject. On the other hand, the business world is "starving" for examples and practical, realistic advice for strategies and operations.

So, there seems to be some gap **between academia and the business world** concerning the treatment of the subject of Supply Chain Management.

The **aim** is to fill this gap by providing an analysis of a lower level, thus use knowledge-based techniques to analyze and model IT companies business and Sc strategies. After examining these strategic, we will develop a business process model (BPM) for Dell that is strategic, business-goal-oriented and executable. To make the BPM executable we will create a workflow engine for BPM simulation and calculation of the total execution time and cost.

So, the **primary objective** of proposed work is to have an insight into IT companies supply chain strategies. The **secondary Objectives** include: i) the development of a BPM for IT companies that illustrates its SC strategies, ii) the creation of a workflow engine for BPM simulation that is business context sensitive, and iii) the simulation of the developed BPM using the workflow engine for further analysis of IT companies strategies.

We will review literature that is relevant to our work and that will be helpful to the reader to bear in mind throughout the report. Since our work will combines a business-oriented subject with a computer science methodology, it is meaningful to review literature of both sciences. So, we will first present some general background information about supply chain management, and then we will review literature the handles IT companies supply chain strategies. Then we will explain some topics that we sill base our work on, thus **FBPML** and the three-layered business process modeling approach.

1.3 Supply Chain Management

A supply chain consists of all parties involved, directly or indirectly, in fulfilling a customer request. In other words, a supply chain (SC) includes all organizations that collaborate in order to pro0duce and deliver a finished product to the final customer, as well as the customer himself. An example of a simple, direct SC would be the one for a bakery in Edinburgh, which contains one supplier, a distributor of the materials, the bakery and a customer.

Supply chains can differ in size, complexity of relations between the members and distribution of physical presence. Here two different types of channel relations can be seen: direct, where the SC consists of one supplier and one customer of an organization, and extended, where apart from the above, a supplier's supplier, a customer's customer, etc. are included. In general, supply chains are dynamic, and involve the flow of information, products and funds between different stage.

2 RESEARCH OBJECTIVES AND SCOPE

2.1 Aim & Objectives

The **aim** of the workflow engine is to simulate a business process model in order to check strategic decisions, in our case IT companies supply chain strategies. The main **objective** is to provide an executable version of a business process model, which will give us an insight into the actual behaviour of the BPM, thus offering a more complete and realistic view of the object of discourse. After all, the workflow engine is the "medium" that takes us from the logical layer to the implementation layer of the Three-Layered Business Process Modeling Approach, which we have adopted for our work and which has been explained in the second chapter. Another important objective of the workflow engine creation is to support the analysis and reasoning about business strategies; through explicitly measuring time and cost that is related to business process execution, the workflow engine will adopt a business context sensitive approach and facilitate comparison between different strategies, and hence different business process model conceptualizations.

2.2 Design conceptualization & Requirements

As we have already mentioned, the main use of our workflow engine will be to simulate a business process model in order to reason about the related business strategies.

Checking business strategies makes sense only when **checking the "normal case"** of business process execution, as simulating an exceptional or working business process model would have a poor contribution to arguing about strategic decisions. So. "checking the average case" means mainly two things:

First, that everything is expected to 'go right" in the business process execution, thus events take place at the right/usual moment, the initial state is correct and guarantees the process execution, actions execute at the pre-specified time point, etc. Second, this means that the duration and cost assigned to each process is the average value of them, thus the expected one.

Since we are only interested in the simulation of the usual and correct business process model, there is no actual need for validation or verification. After all, why check the correctness of a BPM if we already know it is correct? So, our workflow engine does not provide validation or verification, as there is no need for this, under the assumption that the provided BPM is correct.

Additionally, reasoning about business strategies has another impact on the use and design of the workflow engine: IN order to analyze and compare different strategies through BPM execution, one should "reduce" the business operations (and the corresponding time and cost) to the "**single case**". For example, if we want to compare the computer assembling procedure of two companies, such as IT companies like Dell and IBM, then it makes more sense to compare the time and cost related with assembling one computer. This design requirement has two implications: First, that the modeler should already know the cost and time of each "single case" business process and second, that there are not other needed variables for the workflow engine apart from the time and cost of each process. Hence, other variables like number of suppliers, or proportion of big and small customers, are beyond the scope of our workflow engine.

The biggest part of the workflow engine design conceptualization involves general and standard workflow engine issues. Since the workflow engine will be used to make a BPM executable, it will have to conform to some **general workflow engine requirements.** This means that it will have to be able to execute processes, keep track of the workflow state (e.g. know which processes have been executed so far),

understand the current world state (e.g. know which entities and data hold at each timepoint) and update it according to the actions executed, and give some feedback to the user about the business process execution results. To make this clearer, some general requirements for our workflow engine are the following:

i. Understand the business process model, hence understand and "execute" the different junctions of the model.

ii. Understand the definition of business processes and execute them according their special conditions (trigger conditions, preconditions, etc.) and the currents world state.

iii. Understand and update the current world state according to the actions and processes executed.

iv. Keep track of the workflow state, thus "remember" which junctions have been reached and which processes have been executed.

v. Inform the user about the business process execution status.

The first requirement means that junction definition has to be formally specified, so that it is understandable by the workflow engine. Since we have used FBPML for IT companies business process model, our workflow engine will also he based on FBPML for junction definition and execution. So, it should understand what the "start" and "finish" junctions signify, and distinguish between "and-split", "or-split", "and-joint" and "or-joint" junctions, and execute them according the their definitions.

Similarly, the second requirement means that processes have to be formally defined. This definition should include date important for their execution, such as trigger conditions, preconditions and actions they invoke.

In order to make the executable version of a BPM realistic, we should incorporate the description of the world in our workflow engine. Since the world changes according to the workflow state we are in (e.g. what processes and actions have been executed), our workflow engine should be able to update the world state accordingly.

The fourth requirement is an important "control mechanism" of a workflow engine, as it guarantees that we correctly move from one process to another instead of getting

stuck in some workflow state or re-executing processes. Also, keeping track of which processes have been executed is necessary for total cost measuring, and it is interesting information to give to the user as feedback.

The last requirement is actually imposed from the user-side, as the users of a workflow engine need to know real-time what is happening during business process simulation. So, after starting BPM simulation, it would be useful to provide information such as current time point and workflow state; it is also essential to inform the user when the BPM execution is completed and the total cost involved.

2.3 Assumptions

Now that we have made clear what our workflow engine is expected to do, we will discuss how to deal with some design issues and why relevant decisions have been made. In addition, assumptions that are based on design decision will be explained.

- Forward chaining vs. backward chaining algorithm

One of the first design decisions of the workflow engine algorithm is whether to adopt a forward chaining or a backward chaining approach. The backward chaining approach, even though not very popular for workflow engine implementation, may seem convenient for the chosen programming language, Prolog, because of its recursive “nature”. So, the reasoning for a simple BPM composed by two processes would be the following: The BPM execution is completed if the finish junction is reached, which holds if the last process is executed, which requires the previous process to have been successfully executed, and so on. However, our analysis and experimentation with a backward chaining workflow engine algorithm has shown that such a choice makes the estimation of process execution starting time quite complicated, and even incorrect in some cases. On the other hand, a forward chaining algorithm is a more “natural” and correct approach, as it can help us track the state of the BPM execution in each time point. Therefore. A **forward chaining algorithm is algorithm is chosen** for the creation of our workflow engine. The idea of a forward chaining algorithm is the following: The execution of BPM is completed if, starting from the start-junction and by successfully executing the following processes, we reach the finish-junction.

- Simplistic vs. sophisticated treatment of process waiting time

In daily business life it is quite common that processes are triggered later than expected or in not easily predicted time points, especially when the trigger condition has to do with external factors. When such processes are executed in parallel with other processes, then it may become difficult to calculate the starting time of some process following them. So, here arises the question of how we want to deal with such waiting time – in a simplistic or a sophisticated way? Since one of the basic requirement of our workflow engine is to measure time, we have decided to **adopt a more sophisticated and flexible approach.** This means that we will estimate process starting time by taking the corresponding waiting time, if any, into account, and not by neglecting it, as the simplistic approach would suggest.

- Explicit time measurement and real time BPM execution vs. estimation of start and finish time

Another crucial design decision is how to treat time in BPM execution: implicitly, by estimating each process's start and finish time, or explicitly, by representing the world and workflow state in each time point? Even though the second option may be more costly in the case of processes with long duration, it actually turns out that it guarantees a more precise and correct process start time estimation, especially in case we want to model the waiting time for some process. Since we have decided to treat waiting time in a more sophisticated way, we are obliged to **measure time explicitly throughout** BMP execution.

- Junction cases covered

As we have already mentioned, junction definition and differentiation in our workflow engine will be based on FBPML. In these cases a junction connects only processes between them; in fact, this relation may be either one-to-many or many-to-one, and our workflow engine is expected to model and execute both types. We have decided not to cover the case where a junction is connected with another junction, (e.g. an and-joint junction being followed by an or-split junction), as that would make the workflow engine algorithm quite complicated. However, we have recognized the need for modeline a start-junciton followed by an and-split or an or-split junction, and accordingly a finish-junction preceded by an and-joint or or-joint junction, as they are

to be seen in many of IT companies BPMs. These cases will be dealt by "inventing" a new junction which is actually a combination of the two, thus a "start/and", a "finish/and" and a "finish/or" junction. So, the corresponding assumption is that junctions connect only processes between them, except for the case where a start or a finish junction is connected to some other junction.

- Process instantiation

Most workflow engines require a process to be instantiated in order to be eligible for execution, and we will adopt this approach as well. We will regard a process to be instantiated when it is reached through the workflow state, thus when the junction preceding it has been reached and processed. Then, this process instance may be checked for the special conditions (trigger conditions and preconditions) that specify whether it can start execution. Note that for matters of ease we will assume that each process can be instantiated and executed only once, thus our workflow engine will not provide any loop-handling.

- Prior knowledge of events' occurrence

Like most workflow engines do, you workflow engine will relate trigger conditions of process with event occurrences. Even though in real business life it may not be known when events may happen, in our workflow engine, for simplicity, there will be complete prior knowledge about which e vents will take place and when, as opposed to when such info is provided in real-time. This is an assumption that lets us have some control over waiting time of processes. For simplicity, we will have prior information of all evens that will occur throughout the BPM execution, either internal or external, and even if some of them are a "product" of some process execution. This means that we will not include event occurrence as a post-condition (action) of processes, as such information will already be known from the event-occurrence list.

3. PROPOSED CONTENT OF THE THESIS

- Chapter 1 Introduction
- Chapter 2 an overview of literature
- Chapter 3 describes the developed Business Process model

- Chapter 4 Development of the workflow engine
- Chapter 5 Experiments we have conducted on IT Companied using our workflow engine
- Chapter 6 Evolution
- Conclusion

BIBLOGRAPHY

[1] Dell, M., Fredman, C.(2006). Direct from Dell: Strategies that Revolutionized and Industry, Collins.

[2] Dong, Y., Carter, C. R., Dresner, M. E. (2001). "JIT purchasing and performance: an exploratory analysis of buyer and supplier perspectives". Journal of Opoerations Management. 4(19): 471-483.

[3] Fugate=. B.S., Mentzer, J. T. (2004). "Dell's Supply Chain DNA". Supply Chain Management RJeview.p.20-24.

[4] Gattorna, J. (2006). "Supply Chains are the Business". Supply Chain Management Review, September 2006.

[5] Gunasekaran, A., Ngai, E.W.T.. (20050. "Build-to-order Supply Chain Management: a Literature Review and Framework for development". Journal of Operations Management, 23:423-451.

[6] Han, D. and S. Park (2006). "Exception-Based Dynamic Service Coordination Framework for Web Services". Workflow Management Handbook, Layna Fischer:81-96.

[7] Harrison, A., (2203). Lecture Notes of Supply Chain Management Course, "Competing Through Supply Chains", Department of Management Science and Technology, Athens University of Economics and Business.

[8] Jaeger, T., A. Prakash, et al. (1994). "A framework for automatic improvement of workflows to meet performance goals". Proceedings, Sixth International Conference on Tools with Artificial Intelligence.

[9] Kalpic, B. and P. Bernus (2002). "Business process modeling in industry – the powerful tool in enterprise management." Computers in Industry 47(3):299-318.

[10] Kapuscinski, R., R. Q. Zhang, et al. (2004). "Inventory Decisions in Dell's Supply Chain." Interfaces 34(3):191-205.

[11]Kavakli, V. and P. Loucopoulos (1999). "Goal-driven business process analysis application in electricity deregulation." Information Systems24(3):187-207

[12]Koehn, N. F, Michael Dell. (2201). "Winning on the Demand Side of the Information Revolution", Harvard Business School Case 801 363.

[13] Karemer, K., Deddrck, J. (2001). "Dell Computer: Usingh E-commerce to support the Virtual Company", Center for Research on Information Technology and Organizations, Globlization of I.T., (June 1, 2001), paper 236, www.crito.uci.edu/git/publication/pdf/dell_ecom_case_6-13-01.pdf

[14] Kraemer K. L, Dedrick, J., Yamashiro, S. (2000). "Dell Computer: Refining and Extending the Business Model with IT". The Information Society 16:5-21.

[15]Lawton, T.C., Michaels, K.P. (2001). "Advancing to the virtual value chain: learning from the Dell model". Irish Journal of Management 22:91-113, ISSN:1649-248X.

[16]Lee, H. L. (2000). "Creating Value through Supply Chain Integration" Supply Chain Management Review, September 2000.

[17] Lee, H. L., Padmanabhan, V., Whang, S. (1997). "The Bullwhip Effect in Supply Chains". Sloan Management Review 38(3): 93-102.

[18] Lindsay, A., D. Downs, et al. (2003). "Business processes - attempts