Simulating Business Processes – A Review of Tools and Techniques

Benny Mathew and Rajesh Mansharamani

***Abstract*—There is a great need for simulation of dynamic/temporal aspects of business processes. The current standard for modeling business processes is BPMN and in this review, the various techniques and tools that use BPMN for simulation are studied. The review also points to shortcomings in BPMN support for simulation and in the BPM tools claim to support simulation of business processes. We also present a case for general purpose DES tools to widen their scope and create components/plugins to convert BPMN output to DES input.**

***Index Terms*—BPM, BPMN; BPD; BPC; simulation; business process; discrete event; operations research; DES; DEVS.**

I. Introduction

Since the early days of computer simulation beginning in the 1950s, discrete-event simulation (DES) has been the most popular amongst various modeling techniques [1]. Consider simulating flow of a particular fluid in a chemical factory. If the flow rate is plotted against time it will be a continuous curve and hence not suited for DES. On the other hand, DES is suitable for problems in which variables change in discrete times. For example, if the inventory of a particular item in a warehouse is plotted against time, it will be a stepped one. So the increase or decrease of inventory is a discrete function. Apart from inventory management, DES has been found to be invaluable for simulation of hospital patient management, computer systems and manufacturing plants

The Business Process Modeling Notation (BPMN) is a standard to model business process flows and web services. Initially created by the Business Process Management Initiative (BPMI) and now handled by Object Management Group (OMG), the goal of BPMN is to provide a notation that is readily understandable by all business users. This includes the business analysts that create the initial drafts of the processes, to the technical developers responsible for implementing the technology that will perform those processes and finally, to the business staff who will manage and monitor those processes. Thus, BPMN creates a standardized bridge to address the gap between the business process design and process implementation.

II. Need for Simulating Business Process

In order to view impact of business process change (BPC), many companies are using tools or methods for static and dynamic modeling. While static modeling is more widely practiced, dynamic modeling usage is still quite low [2]. However, it has been established that the most important problems related to BPC projects are the inability to accurately predict the outcome of radical change and the inability to recognize the dynamic nature of the processes. Simulation of business processes has been suggested for use in BPC projects as it allows the essence of business systems to be understood, the processes for change to be identified, process visions to be developed, new processes to be designed and prototyped and the impact of proposed changes on key performance indicators (KPI) to be evaluated [3].

Various methods and techniques can be used for modeling business processes in order to obtain an understanding of possible scenarios for improvement. IDEF0, IDEF3, Petri Nets, System Dynamics, Knowledge-based Techniques and Discrete-Event Simulation are some examples of business process modeling (BPM) techniques [4][5]. To address the market requiring the improvement of BPM tools with the components for dynamic modeling, BPM software tools vendors are incorporating simulation modeling features usually using Discrete-Event Simulation (DES) [6].

Before examining the impediments in getting to DES from BPMN, let's examine both technologies in greater depth.

III. Overview of DES and BPMN

A. DES Overview

DES models systems as a network of queues and activities, where state changes occur at discrete points of time. Entities represent objects or individuals and they move through the system and generally are either receiving some sort of time-consuming service or are waiting to receive service [7].

Graphically the structure of a discrete event simulation model resembles a flowchart. The nodes are linked by directed arcs that represent the flow of entities through the system. The nodes correspond to resources where entities wait, receive service or get directed to the next resource. Movement of entities over the arcs can represent physical movement in the system or a change in status like from waiting for service to receiving service. Entities, as they move through a simulation model, frequently need to carry along with them pieces of descriptive information such as a part number, a customer preference, a classification for a package, and so on. These pieces of information, known as attributes, are used at various points in the simulation model to control the routing and handling of the entities.

Let's take an example of a supermarket. A customer enters the supermarket either directly arriving at the entrance or at the supermarkets car-park. Within the supermarket the

Manuscript received May 12, 2012; revised June 25, 2012.

B. Mathew is with Performance Engineering Research Centre of Tata Consultancy Services (email: benny1.m@tcs.com)

R. Mansharamani is currently a freelance consultant in IT Systems Performance Engineering based out of Mumbai This work was done when he was with Performance Engineering Research Centre of Tata Consultancy.

customer will shop for various items. All items are arranged on the supermarket shelves where customer directly picks them up except at the meat and the medical section. At the meat and medical sections the customer has to wait in queue for his turn. Finally the customer will pay at the checkout counters and leave the supermarket. The queuing network diagram that can be simulate with KPI being the end to end time taken by the customer from arrival at parking/entrance to check-out is shown in Figure 1.

Here, the arrival of entities (customers in this case) is generated using a *time based generator*. There are two directed arcs from the *generator* indicating that the generated entities can either directly go to shopping area or will go to shopping area via the parking. Probabilities are assigned to the directed arcs the generated entities are directed to the destination based on the probability values. No queuing happens at either the parking or the shopping area and an *infinite-server* model is used to emulate the time taken for parking and also for shopping. At the medicine and meat sections there are queues using *FIFO* discipline and there may be one or more persons serving at the counters. For the meat and medical section, *N-server* model is used. Entities arriving at the checkout counters choose from one of the many checkout counters, depending upon the queue size. This logic is embedded in the *router*. The router keeps track of the number of entities in the downstream queues and sends subsequent entities to the smallest size queue. Each checkout counter is represented by a *FIFO* discipline queue and also a *single-server* model. Finally the entities enter a sink where statistics for each entity can be collected.

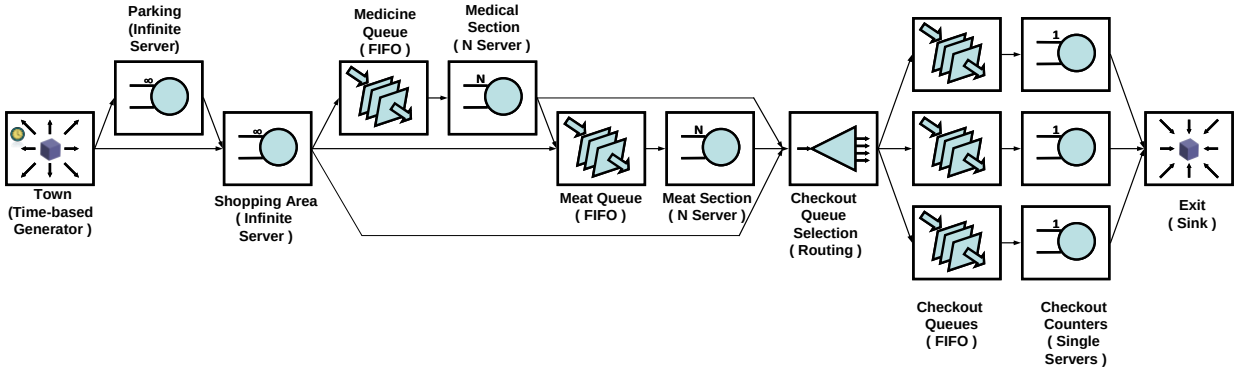

Fig. 1. Example of DES Network Diagram

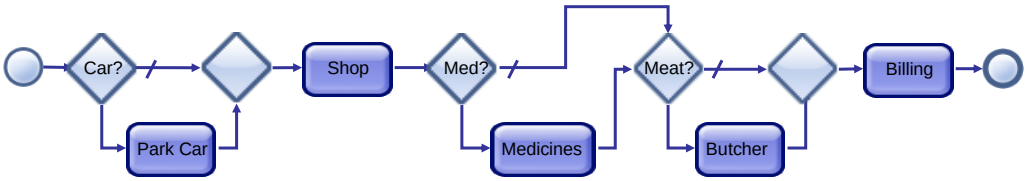

Fig. 2. Example of BPMN diagram

B. BPMN Overview

BPMN defines a Business Process Diagram (BPD), which is based on a flowcharting technique tailored for creating graphical models of business process operations [8]. A Business Process Model, then, is a network of graphical objects, which are activities (i.e., work) and the flow controls that define their order of performance.

A BPD is made up of a set of graphical elements which were chosen to be distinguishable from each other and to utilize shapes that are familiar to most modelers. For example, activities are rectangles and decisions are diamonds.

Usually simulating BPMN models is concerned with verifying correctness and accuracy of the sequence flow of diagrams. For this purpose a token is propagated from the Start Event of the diagram along Sequence flows, across Activities and Gateways, being duplicated and merged when necessary, until it is consumed by an End Event. By studying the path of the token one can verify if the modeled process flow behaves in the same manner that the decision maker wishes or what changes are required to obtain the desired flow. For each activity one can associate a resource or performer (human resource).

Taking the same supermarket example, the BPD of the supermarket will appear as given in Figure 2. This is a simplified example and usually a business analyst will model each of the activity in more details (see section VII).

IV. Gaps in Doing DES Using BPMN

In DES, entities move through a series of queues or buffers acquiring and releasing resources as they move through the model domain [9]. The simulation is driven by a sequence of discrete events that occur when activities are completed, and the entity movement that occurs as a consequence of these events occurring. A superficial view of BPMN diagrams suggests that this is quite similar to DES. Instead of transactions or entities moving between blocks, we have tokens moving between symbols. However it is apparent that this is not the case.

Let us examine some of these differences:

- Unlike in DES, in BPMN there is no equivalent of a queue. In DES, queues are resources where the transactions or entities wait before acquiring resources or messages.
- There is no provision of assigning priorities to a token.
- Once an activity is started there is no way another token

can pre-empt the activity

- Though each activity can be assigned a time for completion, there is no way of assigning inter-task times.
- In DES wherever branching occurs, probabilities are assigned to each branch. However in BPMN there is no provision to do so.
- There is no way of specifying routing policies like shortest number or round-robin in BPMN

V. Research Work on Bridging the Gap

There has been extensive research in the area of making business process models more conducive for simulation. Some of these approaches are even prior to BPMN becoming the standard for modeling business processes.

Syrjakow [10] converts a business activity and makes it compatible with DES. Each business activity is converted into event with begin and end as well as requiring resources. Their web-based tool converts business process output in XML format to Stochastic Petri Nets (SPN). Temporal properties are assigned post conversion to SPN.

Garćıa-Bañuelos et al. [11] have developed tool called OXProS that converts BPMN generated XML output to Colored Petri-Nets (CPN). It expects simulation parameters like arrival times and branching probabilities in with the XML using BPMNs extensibility mechanisms. OXProS provides necessary services for CPN simulation and produces a log file that is analyzed using offline tools to extract the key performance indicators.

Wagner et al. [12] suggests modifying DES specifications itself to support activity. Internally the DES till will treat business activity as a complex DES event having a start event and an end event.

Wong et al. [13] describe a relative-timed semantic model for BPMN in the language of Communicating Sequential Processes (CSP). The model is augmented by introducing the notion of relative time in the form of delays chosen non-deterministically from a range. The authors adopt a variant of two-phase functioning approach widely used in real-time systems and coordination languages like Linda.

Gagn éet al. [14] suggests extensions to BPMN and calls it Time-BPMN. Time-BPMN captures the temporal perspective of business processes. Time-BPMN allows for the specification and depiction of temporal constraints and dependencies within a BPMN diagram. With the extensions, BPMN becomes expressive enough to account for a very large set of real world business processes that are time dependent.

The Event-driven Process Chain (EPC) is a business process modeling language that captures the temporal and logical dependencies between activities of a business process. It was initially described by Keller et al. [15] and then formalized by different authors [16]. EPCs offer three element types: functions, events, and connectors. EPCs extended with data, resources, time and probabilities, called extended EPCs (eEPCs) are intensively used in commercial tools like ARIS and SAP R/3 [17].

VI. Business Process Simulation Tools

There are several BPM tools that support simulation and some of the tools are listed in Table 1. Usually BPM tools they have a diagramming environment where BPDs are created and properties assigned to the activities.

The tools also have a simulation environment where temporal properties are assigned. The BPM tools employee various techniques in order to support simulation which are outside the boundaries of BPMN.

Though the level of simulation support varies from tools to tool, the more sophisticated tools like L-SIM [9] employs the following techniques:

- Allow for probabilities to be assigned wherever token split at gateways. Though this appears simple, BPMN has different types of gateways depending on which type of gateway is used for splitting and merging appropriate mechanisms need to be used during simulation.
- Identify symbols where this is possibility of queuing and get inputs from user about the queuing discipline.
- Identify symbols where resource/performers are required and get time is required for the activity in the form of distribution like normal or exponential.
- Some tools have built in optimizers and expect user entered parameters to be a range of values. The optimizer automatically finds out the parameter values that optimizes the KPI.

During the simulation run, various statistics can be viewed at runtime and on completion of the run, report and various statistics of the simulation is generated.

TABLE I: Business Process Simulation Tools

Tools	Vendor	Whether open source
eClarus Business Process Modeller	eClarus	×
ARIS	IDS Scheer	×
Infinity Process Platform	Sungard	×
Process Modeller	itp-commerce	×
G2	Gensym	×
IBM Websphere Lombardi	IBM	×
Rational System Architect	IBM	×
Metastorm Provision	Metastorm	×
Cordys Business Process Management Suite	Cordys	×
ActiveVOS	Active Endpoints	×
Arena	Rockwell Automation	×
Intalio|BPMS	Intalio	✓
JBoss - jBPM Suite	-	✓

VII. Further Considerations for Business Process Simulation

While the simulation abilities of BPM tools have no doubt been useful, there are still a few concerns:

- Complexity of certain business process cannot be completely captured by BPD tools. In such case where the KPI is highly dependent on complex routing or business logic, there is no alternative other than use of general-purpose DES tools that allow programming in a computer language.

- A Business analyst will tend to model differently as compared to someone with specialized simulation training/experience like Operations Research (OR) analyst. In the supermarket example, most likely a business analyst will put the customer and supermarket into swim-lanes and model the interactions between them in more detail (Figure 3). For example at the checkout business person will model interactions like 'give credit card' and 'get receipt'. So desirable feature of the simulation environment is the ability to merge several activities into one.
- Customer activities that take time but have no interactions with supermarket personnel/systems, like parking car and shopping will tend to be ignored by a business modeler. Here it is essential for the business modeler to also keep in mind the simulation KPI while drawing the BPDs.
- Usually the business analyst will split the organizations entire business processes into smaller and easier to handle segments and draw separate BPD for each. For example an insurance company may sell life, automobile, and medical insurance and the modeler will tend to draw separate BPDs for each. However, the activities in these separate BPDs may share common resources/performers. If we simulate each BPD separately, the response time, utilization, wait time in queues for shared resources will be incorrect. This means that there should be mechanisms to import separate BPDs into the DES environment and simulate them as one.

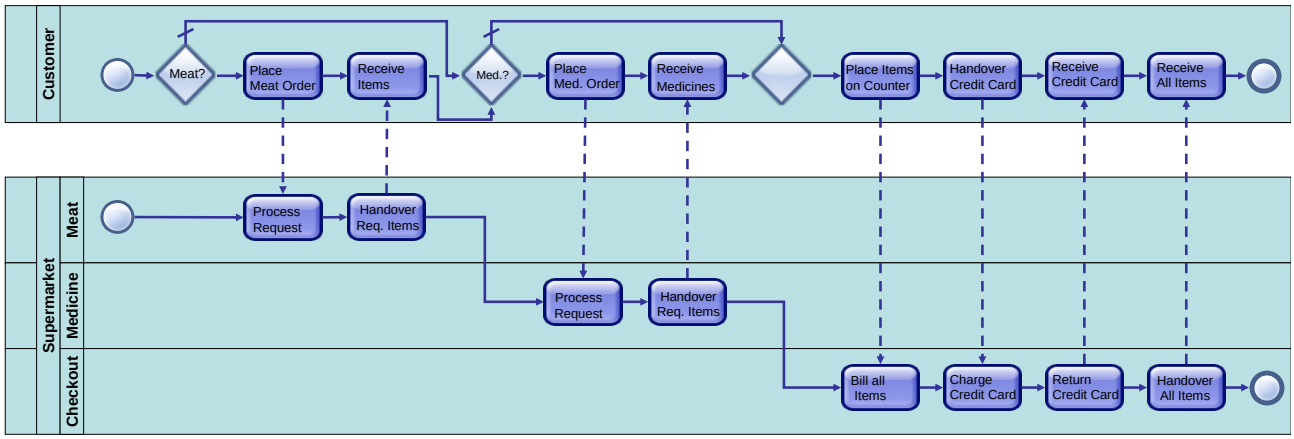

Fig. 3. Example of diagram using BPMN with swim-lanes

VIII. Future for Discrete Event Simulation

The current limitations of using BPMN for simulation can be overcome by one the means given below:

A. Extensions to BPMN

The next release of BPMN is going to be BPMN 2.0 and going by the draft specifications, there is not much of an improvement its support for simulation. The primary intention of BPMN was to provide a notation that is readily understandable by business analysts and resistance to add support for simulation is due to the worry that these new extensions will make BPMN too complex for business analysts. So the option of BPMN extensions for simulation will not be available at least in the near future.

B. Increase Sophistication of Business Process Simulation Tools

Section VI described how various tools operate outside the BPMN boundaries and provide simulation support. These tools can be expected to continue to add features and sophistications to address the current limitations. The drawback of this approach is that the person carrying out simulation will be required to have expertise in both BPMN as well as with Discrete Event Simulation.

C. Convert from BPMN to Discrete Event Specifications

Another option is to model and simulate complex business scenarios using general purpose DES tools. The building of models for business processes from scratch is not the ideal situation, since there is still great value in BPMN, and the efforts of the business analyst will get wasted. One way to avoid this is to have BPMN tool's output converted to standard DES specification like Discrete Event System Specification (DEVS). The techniques mentioned in Section s V and VI can be used for the conversion. An OR analyst can then import the model into general purpose DES tools with which he/she is more familiar with and can exploit all the features of DES tools. This approach avoids adding complicated extensions to BPMN, keeping business analysts happy and the work of OR analysts also gets simplified since they start off with a partially ready simulation model.

Though this approach may seem ideal, the drawback to this approach is that BPMN diagramming tools have not adopted a common standard for output. This means that the conversion tools will have to be aware of the different output formats of the BPMN tools. Despite this drawback, there is definitely value in DES tools to have additional plugins/components that will convert BPMN output to Discrete Event Specifications.

References

[1] S. Robinson, "Discrete-event simulation: from the pioneers to the present, what next?," *Journal of Operational Research Society*, 2005, vol. 56, pp. 619-629

[2] W. Currie and V. Hlupic, "Simulation Modeling: The link between change management panaceas," in *Proceedings of the 2000 Winter Simulation Conference*, pp. 2022-2028
[3] A. Greasley and S. Barlow, "Using simulation modeling for BPR: Resource allocation in a police custody process", *International Journal of Operations & Production Management*, 18(9/10), pp. 978-988
[4] J. Eatock, G. M. Giaglis, R. J. Paul, and A. Serrano, "The implications of information technology infrastructure capabilities for business process change success," *P. Henderson (Ed.) Systems engineering for business process change*, London, Springer-Verlag pp. 127-137
[5] A. F, Seila, V. Ceric, and P. Tadikamalla, *Applied Simulation Modeling*, Southbank, Australia, Thomson Learning
[6] C. Hall and P. Harmon, "The 2005 enterprise architecture, process modelling & simulation tools report," *Business Process Trends*, [Online]. http://www.bptrends.com
[7] Ed Hughes, "Modelling Systems and Resources with SAS Simulation Studio: An Introduction," *SAS Global Forum 2010*
[8] S. White, "Introduction to BPMN," *Object Management Group, Business Process Model and Notation*, [Onlie]. http://www.bpmn.org
[9] A. Walker, M. Clark, and L. Enstone, "L-SIM : Simulating BPMN diagrams with a purpose built engine," in *Proceedings of the 2006 Winter Simulation Conference*, pp. 591-597
[10] E. Syrjakow and M. Syrjakow, "Web-based Business Process Modelling and Optimization," *IMSA 2002: Kaua'i*, Hawaii, USA, pp. 124-129
[11] L. Garc´ıa-Ba˜nuelos and M. Dumas, "Towards an Open and Extensible Business Process Simulation Engine," *CPN Workshop 2009*, [Online]. http://www.cs.au.dk/CPnets/events/workshop09/
[12] G. Wagner, O. Nicolae, and J Werner, "Extending discrete event simulation by adding and activity concept for business process modeling and simulation," in *Proceedings of the 2009 Winter Simulation Conference*, pp. 2961-296.
[13] P. Wong and J. Gibbons, "Relative Timed Semantics for BPMN," *Electronic Notes in Theoretical Computer Science*, Volume 229, Issue 2, (July 2009), pp. 59-75
[14] D. Gagné and A. Trudel, "Time-BPMN," *IEEE Conference on Commerce and Enterprise Computing 2009*, pp. 361-367
[15] G. Keller, M. Nüttgens, and A. W. Scheer, "Semantische Prozessmodellierung," *Ereignisgesteuerter Prozeßketten (EPK)*, Heft 89, Institut für Wirtschaftsinformatik, Saarbrücken, Germany (in German)
[16] J. Mendling and W. van der Aalst, "Towards EPC Semantics based on State and Context," in *Proceedings of the 5th EPC Workshop EPK 2006*, CEUR. pp.25-48
[17] A.-W. Scheer, *ARIS : Business Process Modeling*, Springer-Verlag, Berlin, 2nd edition, 1998

B. Mathew is a member of IACSIT (80338770). He completed his B.E. in mechanical engineering from University of Pane, India in 1992. He then did M. Tech in reliability engineering from Indian Institute of Technology Bombay, India in 1996.

He is working as Senior Scientist at Tata Consultancy Services the performance engineering research centre since the year 2009. Previously he has worked with Tata Research Development & Design Centre, Pune as a part of performance engineering group and also with Veritas Software. In Veritas too, he was part of the performance engineering group. His current areas of research include discrete event simulation and modelling of computer systems, call centres and business processes. He has also worked in the areas of performance measurement, testing and tuning of file systems, databases and middleware.

Mr. Mathew is also a member of IAENG(118060).

R. Mansharamani completed his B.Tech. In computer science & engineering from Indian Institute of Technology-Bombay in 1988. He then went on to complete his MS and PhD in computer sciences from the University of Wisconsin, Madison, in 1989 and 1993, respectively.

He joined Tata Research Development & Design Centre, Pune, in 1994, where he started a performance engineering group. In 2006, he started the Performance Engineering Research Centre for Tata Consultancy Services in Mumbai, where he was a Vice President and Chief Scientist. He is currently a freelance consultant in IT Systems Performance Engineering based out of Mumbai. He has published 17 papers, and 2 book chapters, and holds 15 patents. Some of his recent publications are "Recovery from failures due to Mandelbugs in IT Systems," IEEE PRDC 2011 (along with K. Trivedi and others), "Performance Engineering of a Trading Exchange's Risk Management System," CMG 2010 (along with M. Nambiar), and "Performance Testing: Far from Steady State", IEEE COMPSAC 2010 (along with A. Khanapurkar and others). His current research interests are Performance Engineering Process in IT organizations, and Virtual Production Environments.

Dr. Mansharamani was a member of the ACM in 2011. He has served on Technical Committees of several conferences. Under his supervision TCS won awards for its Java profiler Jensor "Skoch-The World is Open Award!" in 2008, and for its WAN Emulator WANemp "FOSS India Award".