Proceedings of the 2005 Winter Simulation Conference
M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.

ON DEVELOPING SYSTEM DYNAMICS MODEL FOR BUSINESS PROCESS SIMULATION

Lianjun An
Jun-Jang Jeng

IBM T.J. Watson Research Center
1101 Kitchawan Road
Yorktown Heights, NY 10598, U.S.A

ABSTRACT

Business operations can be formally described in business process models that capture activities, information, and flow embedded in business operation. System dynamics modeling is a set of conceptual tools that enable business process designers to build computer simulations of complex business process behaviors. System dynamics models provide accurate description of system behavior along the time dimension. It gives a convenient tool to conduct what if analysis though dynamics points of view. However, to develop system dynamics models requires keen understanding of the "physics" of the target business operations, business organizations, and financial structures and so on. This paper is aimed to provide heuristics and guidelines of developing system dynamics models based on given business process models along with associated reference contexts. An example, from supply chain management domain, of using business process models to derive system dynamics models will be given in the paper.

1 INTRODUCTION

Business processes consist of partially ordered activities that correspond to the operations of their defined business in order to achieve their common goal. For example, a typical process in the manufacture industries tends to include ordering parts, assembling them into products, marketing their products, receiving orders from customers, accounting the order, shipping products to customers and so on. A business process covers one or more than one business functions and business organizations to realize the desired *enterprise* behavior. An enterprise may consist of multiple business processes.

The information structure for a business process can be defined as a network of activities performed by resources so as to transform inputs into outputs. Commonly, management policies are used to specify how a process should be operated over time—how and when it should be operated and which resources should be allocated to its activities. It is feasible for a process to strive achieving a global optimum in the enterprise instead of a local optimum for an organization if all organizations touched by the process can be taken into consideration as a whole. However, for a cross-enterprise business process, the global optimum is not achievable most of time.

Cohn and Stolze (2004) discussed the future direction of model driven enterprise. The model-driven enterprise approach provides the advantageous tool of understanding and modeling business structures. Wyssusek et al. (2001) argued that modeling can be considered as a process of knowledge acquisition about the target business operations. Business process model concretizes activities, information, and flow embedded in business operations into business tasks with explicit reference context such as business organizations, resource models, and financial structures. A formal business process model enables the simulation of target business operations in real world. An appropriate business process simulation would provide the insights of resource usage patterns and the performance of the organization where the business processes would be deployed and functioning. A business process model contains different levels of granularity on operational specification. Its structure represents the logical temporal sequence of functions consideration. A validated process model enables us to evaluate business process through simulation hoping to surface possible outcomes through what-if analysis and to improve process design through re-engineering.

There are different methodologies available to model business processes, for instance, flow diagramming based, and system dynamics based. The degree of using the methods varies. Models are an abstraction of the real world system. They represent different level of granularities, and their constituents can be mapped to real world entities. When modeling the same target system from the same perspective, its representations must be consistent and some connection exists between them. Topologically, two graphs are called homeomorphic if there is a continuous deformation taking one to the other. We borrow this terminology for models between which there exists continues structural transformation taking from one to the other. It implies that there is a "homeomorphic" map between constructs of each model from different technology. Under homeomorphism, one model might have an aggregated view of some part (contracting to one point, like handle payment in Figure 1); while the other might have a detail view of the same part. In this paper, we compare Business Process simulation model with System Dynamics model when being used in studying business process. By comparison, we identify the map between constructs in these two models. This would help us to build its System Dynamics model based on its corresponding Business Process simulation model.

Both modeling technologies, business process simulation (BPS) model and system dynamics (SD) model, have simulation capability to help modelers understand the business processes, to provide some insight to manage and to improve the business processes. Business Process model mainly supports discrete event simulation and System Dynamics model is good at continuous simulation. These two approaches compensate each other and help modelers to understand the problem from different perspectives. Simulation can reduces the duration of running scenario to manageable time, thus making what-if-analysis becomes possible. In this paper, a supply chain scenario will be used as the running example. The flowchart of the process given in Figure 1 describes the process that the manufacturer (1) orders parts from its suppliers; (2) produces well-configured products; and (3) tries to satisfy customer demand for its products. We will use Figure 1 as a running example in this paper. First, we develop a business process simulation (BPS) model by capturing activities and related data. Second, this model will be evolved into a system dynamics (SD) model. Third, the simulation outcome based on the SD model will be exploited to improve the original BPS model.

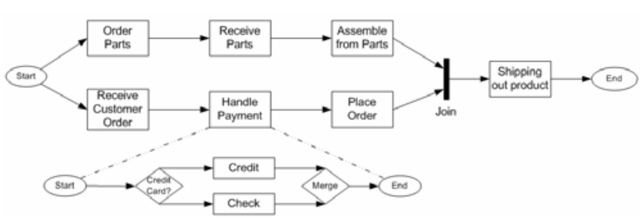

Figure 1: Flow Chart for a Business Process

The organization of this paper is as follows. Section 2 discusses different business process modeling techniques and makes comparison among them. Section 3 discusses business process simulation and compares discrete event simulation with continuous simulation. Section 4 demonstrates some result based on system dynamics model. Section 5 concludes this paper with discussion and future work.

2 BUSINESS PROCESS SIMULATION MODEL

This section describes two most prominent approaches to simulate the behavior of business processes: business process flow charting and discrete event simulation model.

2.1 Business Process Flowchart Modeling

A business process model is an abstraction of its target business operations in the real world. Intuitively, a business process can be represented by a set of activities with a set of links connecting them. The minimal semantics of a link represents the temporal or causal relationship between those two activities it connects. Figure 2 shows an example of business process model in IBM's WBI Modeler. Rectangles are used for representing business activities, diamonds for business decisions (a special kind of activity), and lines for linkages between activities. Based on the customer order, the manufacturer orders parts, produces product and fulfills the order. Before the tasks "Ship out Product", the activity to synchronizing previous activities is a *join* activity. It is a synchronization point where the products can be shipped out only if both finished products are available and orders are placed. An activity can delegate the operations to other process. For example, the activity "Handling Payment" can be mapped different processes which may use different payment methods. This notion of sub-process can be applied to decision points and then re-join the main flow in the process.

Business process flowcharting helps modelers grasp the behavior of business processes and have better communication with one another. In order to understand the process from certain perspective, modelers might need to decompose business process flowcharts for the sake of taming the complexity of the business process infrastructure. Similarly, some modeling elements in a business process flowchart may need to be either compressed/aggregated (zoom out) or expanded/re-factored (zoom in) during the modeling process in order to obtain the most appropriate level of abstraction for human modelers. IN many cases, the lower level control and data logics require to be hidden for encapsulating the unnecessary details. We are using the term homeomorphic structures to indicate those structure that have the same semantic behavior but different syntactic structure due to the aforementioned transformations. Note that an activity can be reduced to non-existence where, in this specific case, two homeomorphic structures are not isomorphic. Several business process modeling tools are already available in the market, for example, IBM's Websphere Business Integration (WBI) Modeler. In addition to the similar building blocks in business process flowcharting, WBI Modeler also has the capability of creating business resources (e.g. IT staff) and business organizations (e.g. human resource), which can be associated to tasks. Figure 2 shows the swim-lane view that tasks are laid out in different swim lanes based on the business units with which they are associated.

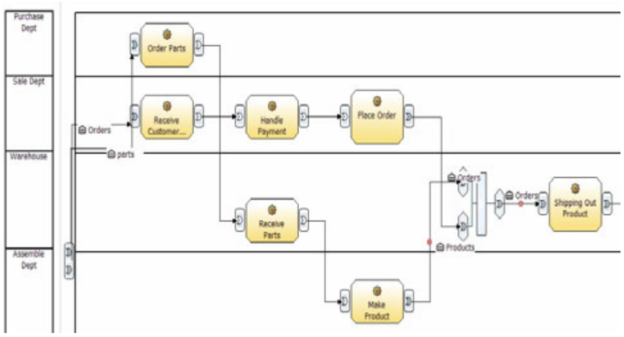

Figure 2: Swim-Lane View of Process Based on Business Organizations

This type of model would be very useful to align organization structure and location with the business process, to address resource allocation and utilization problem, to identify responsibility based on role. In summary, business process flowcharting and similar modeling notions help human modelers to create business processes create accurate structure for the business processes under consideration.

2.2 Business Process Simulation Model

However, to evaluate a business process, human modelers need the information beyond functional structures. The inherent behavior needs to be discovered from data that business processes are to manipulate. We name such related data used by business processes to be *business artifacts.* Nigam and Caswell (2003) explained in detail about business artifacts that are essentially just business records associated to corresponding business operations. An activity in a business process model manipulates business artifacts either whole artifact or part of it. We are using WBI Modeler to specific business process models. WBI Modeler has the concept of business items that are close to that of business artifacts. A business item can be associated with the input or output ports of an activity. The data, coming with business item and flowing into some activities through its input port, might not be sufficient for the business process to make decisions and perform embedded tasks. If that's the case, more data can come from data repositories. A repository can be only used by those activities that are defined by the same process. The data residing in a global repository are visible by all processes and tasks. The pre-and-post conditions of an activity are determined by result of evaluating the data values associated with both ports and repositories.

Figure 3 shows an example of business process simulation model that include activities and data repositories. A storage buffer may exist between two business activities. Some of them are included explicitly as data repositories since other activities need that information to make decision. On the left side of Figure 3, two parallel sequences of activities are defined to the right temporally. The top one starts with the activity "Ordering Parts". It takes inputs from the data repository "Part Demand" to determine how many parts to order and put order records in the repository "Part WIP." The next activity "Receives Parts" updates the repository "Part WIP" and puts a record in the repository "Part Inventory." The following activity "Make Product" updates the repository "Part Inventory" and uses the information from the repositories "Bill of Material" and "Product Demand." After products have been produced, a record is created in the repository "Finished Product." The product is then ready for ordering. The second sequence (at the bottom) starts with the activity "Receive Customer Order" followed by two activities: "Handle Payment" and "Place Order." The activity "Place Order" will create a record in the repository "Order Backlog."

In the second sequence, a branching logic is created to calculate the demand forecasting and to put record in a repository "Product Demand" and "Part Demand" that are used in the activities "Make Product" and "Order Parts" of the first sequence for decision making purpose. The activity "Receive Customer Order" puts record in the repository "Customer Order History" which is used by the activity "Forecast Demand." Then the following activity "Adjust Product Based on Finished Product" will adjust "Product Demand" based on how much products have been finished (the repository "Finished Product" is updated in the first sequence). The activity "Adjusting Based on Part Inventory Error" would consume "Product Demand" and produce the "Part Demand" using "Bill of Material" information. It will also adjust the demand figure based on the difference between the desired inventory level and current "Part Inventory" that was created in the first sequence. For the right side of the model, there is a synchronization point "Check Order and Product Status" of the above two sequences. It checks whether there are both "Finished Product" and "Order Backlog". Based on which number is bigger, there are different arranging order strategies taken. Finally the activity "Shipping out Product" leads to the end of the whole process.

Input criteria of business activities can be defined in WBI Modeler to expresses the when to trigger the task. Output criteria of a business activity can be used to trigger connected link. Human modelers can use built-in expressions to build the criteria for business tasks. It is possible to plug in some Java program to express the criteria and, at run time, the executing engine of WBI Modeler will call those procedures directly. In order to run business process simulation and to carry out what if analysis, we have to quantify all variables in the business items and formulate all relationships among variables. The most common study for such business process models is discrete event simulation. The business item related to the whole process is generated according to some type of distribution, e.g. Poisson distribution. Inside each activity, there could be also some distributions related to processing time, random variables in the business item to mimic the real-time activities. Therefore, the whole business process is treated as a stochastic process. Then some statistical analysis technologies are applied on data collected from multiple (Monte Carlo type in dynamic programming) simulations to derive characteristic of the underlying process, like cost implication, product throughput, and process bottleneck. Reader can consult the book written by Anupindi et al. (2003) for using variety of analytic methodologies to evaluate business processes and managing business process flows.

Another advantage of BPS model comes from its objective thinking based on the real business activities sequencing. As a result, the business process model can actually become active in the sense that each activity is implemented to hook up with real application. Then the model is transformed into certain kind of business process controller. In fact, WBI Modeler does have the capability to create BPEL scripts. BPEL stands for Business Process Execution Language which is well viewed as the de facto industry standard to describe business processes. In this case, the activities of a business process would be delegated to call Web services. The Web service is wrapped and adapted to the application through its API. The objective thinking paradigm also helps us to collect data consisting with the way in which the model identifies variables in the business item and to validate the model.

3 SYSTEM DYNAMICS MODEL

This section describes what a system dynamic model is and how it can leverage to simulate the behavior of business processes.

3.1 System Dynamics Methodology

System Dynamics modeling is a visual tool that is used to capture casual relationships among business tasks and also the feedback control loops for the target system. It consists of methodology constructs like *stocks* and *flows.* The *stock* corresponds to an accumulative variable like stack and level. The *flow* describes the rate in which the stock will be changed. The *flow* describes the rate in which the stock will be changed. A flow could have in-flow that increases stock and out-flow that decreases the stock. To determine the flow rate, it is required to have system think about the target with full dependency graph. Casual link from one variable to the other can be marked as positive or negative

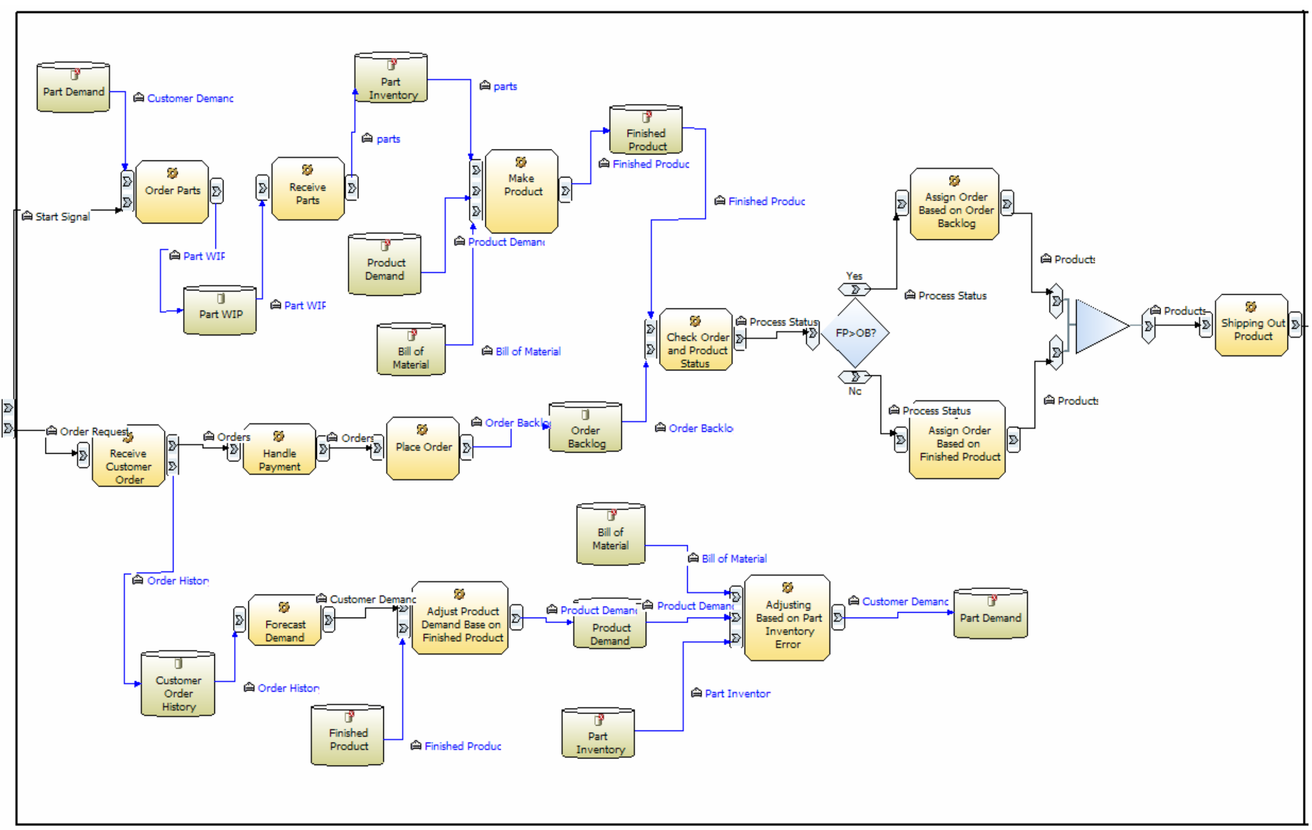

Figure 3: Business Process Simulation Model

based on whether their values change in the same direction. Feedback loops are identified by checking whether all arrowed links from a loop. If number of links with negative sign on the loop is even, then the loop is called a *positive feedback loop* and causes self-reinforcing. If the number of links with negative on the loop is odd, then the loop is called a negative feedback loop and causes self-correcting (re-balancing). Indeed, concept maps in the model make our thinking visible and organized. The visual tool helps us construct knowledge and recognize patterns and connections.

System Dynamics modeling can be used to study variety of systems. Forrester (1961) initiated systems dynamics to model various industry problems. Sterman (2000) argued that systems dynamics can be used to study business dynamics. We have also been using system dynamics to study the demand conditioning process in supply chain (An and Ramachandran 2005) and Web service management (An and Jeng 2005). Here we only discuss how to use it to model business process. Figure 4 shows the System Dynamics model of the same supply chain process. There are four stocks and each has their own in-and-out flows. Comparing it with its business process simulation model, we left out the handling payment part since it is not our concern here.

3.2 System Dynamics Formulation

There is a fundamental difference between SD and BPS models. SD modeling intends to capture physical laws governing the system. The modeling process is based subjective thinking (metal model) with the hypotheses of dynamic behavior manifested by all entities in the system. Stock and flow diagram is used to model the conservative quantities during the system evolving in time. When using SD to model business processes, the first step would be to capture the stock and flow structures in the process. Note that there is a storage buffer between activities in the business process simulation model. Its quantity can be modified by the surrounding activities. The stock "Order Backlog" in Figure 4 corresponds to the storage buffer between activities "Place Order" and "Check Order and Product Status". So the activity "Place Order" from customer will have the effect of increasing the stock "Order Backlog" and changing the in-flow "Demand Rate". Similarly, the activity "Shipping out Products" after the storage would have the effect of decreasing the stock and changing the out-flow "Fulfillment Rate". The stock "Finished Product" is the result of the activity "Make Product" and will be reduced by the activity "Shipping out Products." The stock "Part WIP" is after "Order Parts" and is reduced by "Receive Parts"; and the stock "Part Inventory" will be increased due to the effect of "Receive Parts" and reduced by the activity "Make Product". In fact, we purposely name all stocks with the same name of data repositories representing the corresponding storage buffers in Figure 3. It is less clear how the activities in the BPS model affect in-and-out flow rates for each stock. In the BPS model, dependencies can be expressed graphically only through the input ports. Any additional dependencies could be buried in the input criterion and executable expression. But dependencies in the SD model can be expressed graphically with the notion of *polarity*. The influence map can be created by connecting the direction links. For instance, the

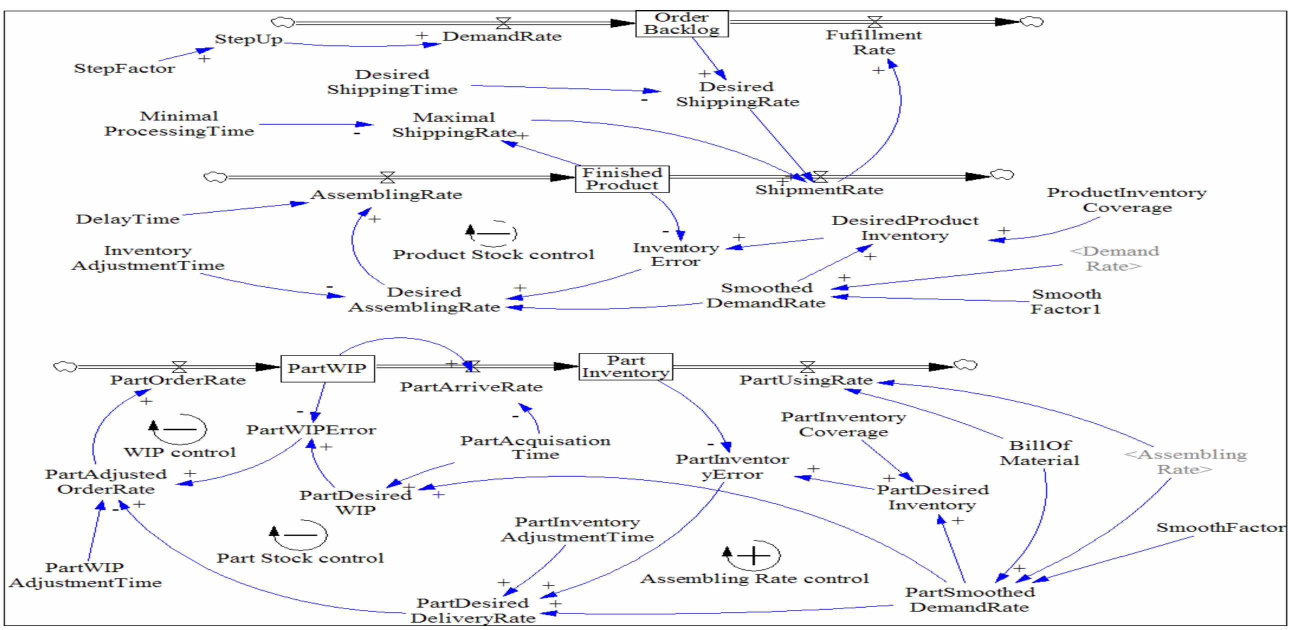

Figure 4: System Dynamics Model of Supply Chain Process

causality tree for "Shipment Rate" can be visually obtained as shown in Figure 5. The causality tree corresponds to the business activities between "Check Order and Product Status" and "Shipping out Product" shown in Figure 3. The "Desired Shipping Rate" is determined by "Order Backlog" and "Desired Shipping Time",

$$\mathrm{DesiredShippingRate} = \frac{\mathrm{OrderBacklog}}{\mathrm{DesiredShippingTime}}.$$

And the "Maximal Shipping Rate" is determined by "Finished Product" and "Minimal Processing Time",

$$\mathrm{MaximalShippingRate} = \frac{\mathrm{FinishedProduct}}{\mathrm{MinimalProcessingTime}}.$$

Finally, the "Shipment Rate" is determined by "Desired Shipping Rate" and "Maximal Shipping Rate" as the following

$$\begin{aligned}&\mathrm{ShipmentRate} = \\ &\min(\mathrm{DesiredShippingRate}, \mathrm{MaximalShippingRate}).\end{aligned}$$

Note that variables "Desired Shipping Time" and "Minimal Processing Time", which do not depend on others, can be assigned proper values based on its average value of real processing time. Should we use a normal distribution with given mean and variant? The answer is negative since an SD model tends to represent a deterministic model and is used to study overall behavior in certain given time scale. Comparing with the BPS model, the SD model is aimed to cover the detail "Shipping out Product" activities for the sake of mimicking the real world events in a much smaller scale. The assigned values in the SD model can be obtained from the simulation result of the BPS model. Other data sources are also possible, e.g., the average of the real data that are identified in the business artifacts of the BPS model.

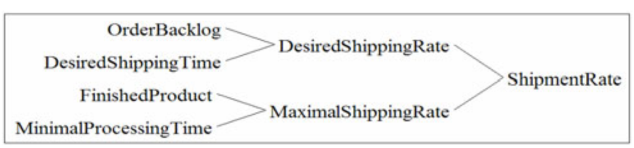

Figure 5: Causality Tree for Shipment Rate

The stock "Finished Product" is added through "Make Product" using "Product Demand" which comes from "Forecast Demand" plus "Adjust Product Based on Finished Product" in BPS model. The forecast model we have here is the exponential smooth of historical data,

$$\begin{aligned}&\mathrm{SmoothedDemandRate} = \\ &\quad \mathrm{Smooth}(\mathrm{DemandRate}, \mathrm{SmoothFactor}).\end{aligned}$$

Where the function smooth is one of built-in functions in the system dynamics tool. The adjustment from "Finished Product" introduces a negative feedback loop to rebalance the amount of products we should to assemble

$$\begin{aligned}&\mathrm{DesiredProductInventory} = \\ &\quad \mathrm{SmoothedDemandRate} * \mathrm{ProductInventoryCoverage} \\ &\mathrm{DesiredAssemblingRate} = \mathrm{SmoothedDemandRate} \\ &\qquad + \frac{\mathrm{DesiredProductInventory} - \mathrm{FinishedProduct}}{\mathrm{InventoryAdjustmentTime}}.\end{aligned}$$

The real assembling rate could be delayed

$$\begin{aligned}&\mathrm{AssemblingRate} = \\ &\quad \mathrm{DelayFixed}(\mathrm{DesiredAssemblingRate}, \mathrm{DelayTime}).\end{aligned}$$

The part usage rate is transformed from "Assembling Rate" using "Bill of Material",

$$\begin{aligned}&\mathrm{PartUsingRate}[i] = \\ &\quad \sum_j (\mathrm{BillOfMaterial}[i, j] * \mathrm{AssemblingRate}[j]) \\ &\mathrm{PartSmoothedDemandRate} = \\ &\quad \mathrm{Smooth}(\mathrm{PartUsingRate}, \mathrm{SmoothFactor}).\end{aligned}$$

The part demand, used to determine how much to order from suppliers, can be adjusted based on "Part Inventory" (another negative feedback loop),

$$\begin{aligned}&\mathrm{PartDesiredInventory} = \\ &\mathrm{PartSmoothedDemandRate} * \mathrm{PartInventoryCoverage} \\ &\mathrm{PartDesiredDeliveryRate} = \\ &\quad \mathrm{PartSmoothedDemandRate} \\ &\qquad + \frac{\mathrm{PartDesiredInventory} - \mathrm{PartInventory}}{\mathrm{PartInventoryAdjustmentTime}}.\end{aligned}$$

In process of developing BPS model, we used the data repositories to share data. They are also used to identify what data we need to make operational decision at several points (input ports of activities), and which activities are responsible to generate these data. That information is exploited in the modeling process to form causality relationships in a graphical manner. Based on the performance concerns in business operation, we only include necessary constructs from the corresponding BPS model.

3.3 System Dynamics Simulation

System dynamics is methodology to capture the behavior of the target system through stock flow diagrams with dependency graphs and mathematical formulation (system of ordinary differential equations). Simulation of this system would expose its dynamic behavior in time dimension. Because of complexity of system with nonlinearity and time delay, we may not be able to solve the system analytically. Based on available numerical method for ordinary differential equation, like Euler's first order finite difference, Runge-Kutta second and fourth order finite difference method, the system can be solved numerically.

By nature, a solution for the system of ordinary differential equation is continuous. But it would not prevent us from representing discrete states and time. The key is the scale about the measurement points along the time line. For example, if a system operates on the time scaled on days, it would be inappropriate to study its dynamics in the scale of hours. Multi-scale analysis may be necessary to study the behavior of a system covering different scales. For any system, there is a lower bound for the continuous time scale and the upper bound for the discrete time scale. Based on this premise, three cases can be considered as follows,

1) If the time-scale is greater than both scales in the system, then continuous time scale can be assumed. Thereby, we tend to use continuous simulation model such as system dynamics.
2) If the time-scale is less than both scales in the system, then the discrete event simulation technique would be more appropriate.
3) If the time-scale is between two bounds, then those events occurring in smaller time scale can be aggregated to the large scale, continuous based simulation models. In this case, we should use mixture of both techniques to simulate the behavior of the target system.

No mater what modeling methodologies we use, it should manifest similar behaviors as we investigate the same system with the same concern in mind and in the same scope. In discrete event simulation, the event comes into the input points with varying interarrival time and some variables may associate to certain distribution. In the continuous simulation, we can use amplitude that relates to the density of events. We can transfer the number of events in the unit of time into amplitude within equally distributed time steps. It is similar to relationship between frequency and amplitude modulations in other communi cation systems. An analog signal can be carried in the high frequency wave by either changing its frequency or changing its amplitude based on the signal. The analog signal can recover at the other end through demodulation.

The BPS model treats a business process as a stochastic process. Its deterministic behavior and concepts are derived from statistical analysis from multiple detailed simulations. But the SD model treats the business process as a deterministic system at very beginning by capturing governing law of system and establishing first principal (dynamic hypothesis). Usually, the BPS model is suitable for short time span; while the SD model can be used for large time span. As a result, SD model is suitable for stability study to investigate performance implication in a long time span. This point will be elaborated in the following section.

4 RETROFITTING BPS MODEL USING SD MODEL

Based on the SD model built up in the last section, we study behavior of its solution. Specifically, we demonstrate that some unexpected behavior could be fund through simulation. It leads to deep thinking to figure what missing in the BPS model and to correct the BPS model.

4.1 Model Verification and Validation

After having built the model and quantified the relationship in the model, we need to verify and validate the model. Verification is to check whether the model is right and its solution behaves properly. Validation is to check whether we have a right model and the behavior of solution is consistent with real system. The existence of a steady state solution would be a very good indicator for model verification. In our system, when the demand rate is constant and stocks have proper initial values, the system could have constant solutions. Indeed, that is the case for our system as shown in Figure 6.

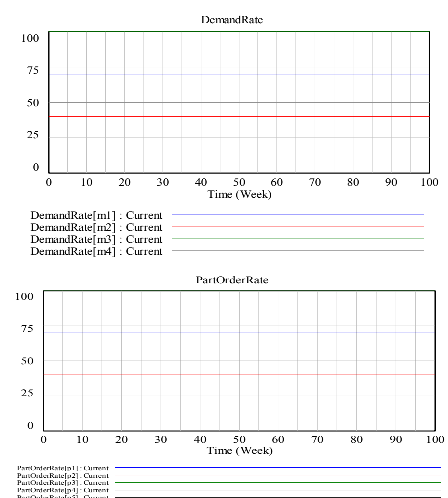

Figure 6: Part Order Rate for Constant Demand

Now what happens if the demand has jump at time=30 as shown in Figure 7. Figure 8 shows the fact that part order rate has oscillation occurring with current order policy. In fact this occurs because we only consider current demand plus adjustment based current inventory level, but we ignore the order in process. Recall that we forecast the current demand based on the current order history and require current inventory having two weeks safety stock. To adapt to sudden change of demand (stepping up at time=30 in our case), we need to step up ordering parts from suppliers based increasing demand plus safety stock adjustment. Note that there is a leading time to receive the order. When placing order only based current inventory level, we would over-order the part due to the accumulation of orders in processing. After all orders arriving, the inventory gets over-stocked and we have to reduce quantity in the subsequent orders significantly and lead to under-stock in the following weeks again. As a result, the rate of ordering parts from suppliers will oscillate as shown in Figure 8. The amplitude of oscillation increases as the leading time for order in process becomes longer. To correct the problem, a negative feedback loop from "Part WIP" should be included,

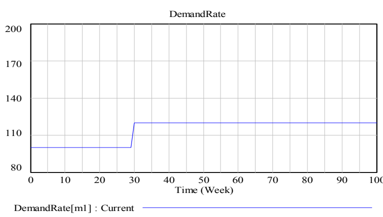

Figure 7: Stepped up Demand Rate

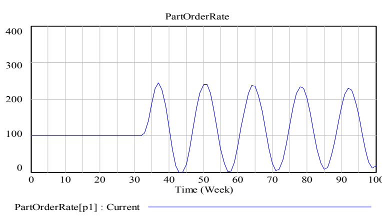

Figure 8: Part Order Rate for the Step up Demand

$$PartDesiredWIP = PartSmoothDemandRate * PartAquisationTime$$

$$PartAdjustedOrderRate = PartDesiredDeliveryRate + \frac{PartDesiredWIP - PartWIP}{PartWIPAdjustmentTime}.$$

Figure 9 shows the behavior of part order rate after correction. The order rate has ramp up and down change for one cycle and stabilize at a new constant value. The feedback loop from considering orders in processing stabilizes the system.

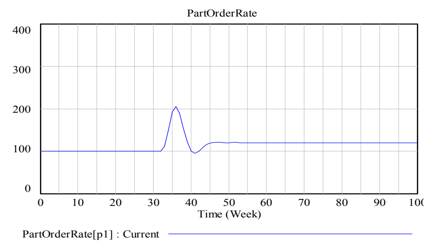

Figure 9: Part Order Rate after Correction

4.2 Modification of the BPS Model

Additional activity "Adjusting Based on Part WIP Error" should be added before the data repository "Part Demand". This activity has two inputs: one comes from the output of "Adjusting Based on Part Inventory Error" and the other from the data repository "Part WIP" that is up dated through the activity "Order Parts". The change is reflected in Figure 10. In the BPS model, such missing activity might not be deleted very easily since its simulation is built for small time span. But the SD simulation can be easily set for different time span and is used to study stability in a large time horizon and to investigate changing pattern along timeline and control mechanisms for the system. It is not straight forward to include decision rules in the business process simulation models. It is relative easier to include temporal dependencies comparing with including causal dependencies in BPS models. In fact, there is no visual support to set up causal dependencies excepting for information sharing through data repositories.

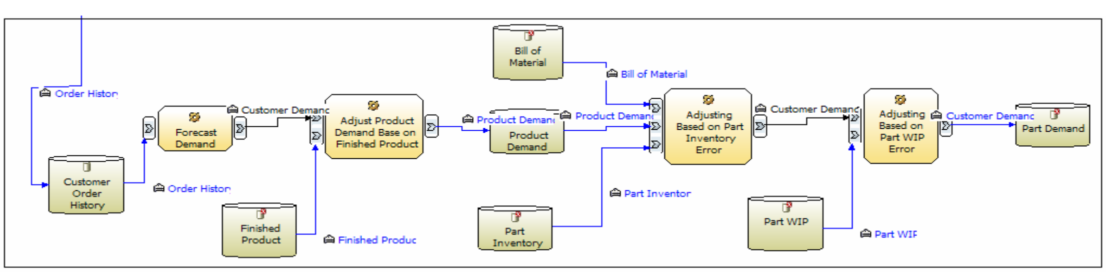

Figure 10: The BPS Model with Order in Process Adjustment

5 CONCLUSION AND DISCUSSION

We discuss the relationship among technologies that can be used in the business process modeling and simulation. Flowchart gives a stating point to capture activities as well as their sequencing for business operation. Business process simulation model includes business artifacts that are used to quantify business rule and control flow criterion. System dynamics modeling allows us to visually express the decision rule and dependencies, and visualize the feedback loops in the system model for the given business process. Those modeling techniques enable us to design business process and to re-engineer the process. The simulation capabilities of those modeling tool enable to evaluate the design process and to conduct what-if analysis of the process and tuning the decision to optimize the process.

Business process simulation model is based on objective thinking. The modeling process is intuitive and can easily map to reality. It is natural to use business process modeling tool to capture association organization and resource with activities. Its discrete event simulation can be used to mimic detail activities in the process. And then statistical analysis can be used on the simulation data to derive deterministic behavior in a relative short time span. The system dynamics model is based on subjective thinking. It captures driving force and governing law for the system evolution in the time dimension. Its simulation can be used for study system characteristics such as stability issues and feedback control structure in a relative large time span. It is desirable to combine both techniques in the study of business processes behavior. Discrete event simulation represents real activities in a small scale with stochastic behavior; while system dynamics simulation represents scaled up (averaging, aggregation) intrinsic structure in a large scale with deterministic behavior.

To design an effective business process, simple physical activities sequencing and synchronization is not enough. It is important to capture metrics that measure business performance and to make intelligent decision whenever those metrics change. The SD model enables us to evaluate metric change in time dimension. The analysis of simulation results exposes the hidden factors that affect business performance and helps us to build better decision making processes. Thereby, business process simulation model and system dynamics model are complementary in the sense that the former provides structural proximity of human behavior in the context of business processes; and the latter help modelers to discover and describe formally the inherent causalities within business processes. In real practice, the methods presented in this paper can be used as reference guidelines for modelers to perform their analysis and explanation of business process behavior in a more comprehensive manner.

REFERENCES

An, L., and J.J. Jeng. 2005. Web service management using system dynamics, *2005 IEEE International Conference on Web Services*, pp. 347-354.

An, L., and B. Ramachandran. 2005. System dynamics model to understand demand conditioning dynamics in supply chain, *The 23rd International Conference of the System Dynamics Society*.

Anupindi, R., S. Chopra, S.D. Deshmukh, J.A. Van Mieghem, E. Zemel. 2003. *Managing business process flows*, Pearson Education Asia.

Cohn, D., and M. Stolze. 2004. The Rise of the model-driven enterprise, *IEEE International Conference on E-Commerce Technology for Dynamic E-Business*, pp. 324-327.

Forrest, J. W.. 1961. *Industry dynamics*, MIT Press, Cambridge, MA, 1961.

Nigam, A., and N.S. Caswell. 2003. Business artifacts: an approach to operational specification, *IBM System Journal*, Vol. 42, No. 3.

Sterman, J.D.. 2000. *Business dynamics: system thinking and modeling for a complex world*, Irwin McGraw0Hill, Boston.

Scholl, H.J.. 2001. Agent-based and system dynamics modeling: a call for cross study and joint research, *Proceedings of the 34th Hawaii International Conference on System Sciences*.

Wyssusek, B., M. Schwartz, B. Kremberg, F. Baier, H. Krallmann. 2001. Business process modelling as an element of knowledge management – a model theory approach, *Conference of Managing Knowledge: Conversations and Critiques*.

AUTHOR BIOGRAPHIES

LIANJUN AN is a Researcher at IBM's T.J. Watson Research Center and currently working on studying stability of supply chain through system dynamics modeling and simulation, and designing business performance monitoring and management system. He received Ph.D. degree in Applied Mathematics from Duke University in 1991. He worked on Analysis of Granular Flow and Plastic Deformation, Scientific Simulation of Oil Reservoir on Parallel Computer in McMaster University and the State University of New York at Stony Brook (1992-1997). He subsequently joined IBM and worked on Network Configuration Management System, Websphere Commerce Suite and Grid Computing Projects since 1998. His email address is alianjun@us.ibm.com.

JUN-JANG JENG is a Researcher at IBM's T.J. Watson Research Center. He is leading the research and development efforts in business process, innovation and optimization. He received Ph.D. degree in Computer Science from

Michigan State University in 1994. Hi interested research areas include software engineering, formal modeling and verification, AI and agent technologies, and decision support systems. Prior to joining IBM, Dr. Jeng had affiliated with for Bell Labs as a researcher and George Washington University as an assistant professor. He can be reached at `jjjeng@us.ibm.com.`