

The development of a dynamic supply chain analysis tool— Integration of SCOR and discrete event simulation

Fredrik Persson^{a,*}, Mirko Araldi^b

^a*Department of Production Economics, Linköping Institute of Technology, S-581 83 Linköping, Sweden*

^b*Department of Industrial Engineering, University of Parma, Parco Area delle Scienze 181/a, 43100 Parma, Italy*

Received 9 May 2006; accepted 12 December 2006

Available online 12 March 2007

Abstract

The supply chain operations reference (SCOR) model is developed and maintained by the supply chain council (SCC). The SCOR model is a reference model that can be used to map, benchmark, and improve supply chain operations. SCOR provides companies with a basic process modelling tool, an extensive benchmark database, and defines a set of supply chain metrics.

The software ARENA is a commercial simulation tool that can be used for simulation modelling in various applications. An integration of SCOR and ARENA provides the supply chain analyst with a comprehensive and dynamic tool.

This paper reports on the ongoing work to integrate the SCOR methodology and a discrete event simulation tool. SCOR is basically used to study the static operations of a supply chain. However, there is also a need to study the dynamic effects e.g. of changes in production rate, poor quality in raw materials, and other effects related to the “bullwhip” behaviour of a supply chain.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Simulation; SCOR; ARENA; Supply chain

1. Introduction

Supply chain management incorporates the use of analysis tools such as system dynamics, optimisation, and simulation. Theoretical models of the supply chain behaviour can be created by observing the supply chain's historical data or by collecting new data. Experiments that study the supply chain behaviour are useful in order to find causal effects and to test different or even extreme scenarios. Causal effects are, however, difficult to find if they are separated in time and space and extreme

scenarios are hard to control in a supply chain. An alternative to conducting experiments in the actual system is of course to use a model of the system for experimentation.

Many of the supply chain models found in the literature are models used for optimisation. These models answer questions about plant location, product mix, choice of technology, means of distribution, inventory planning and control, choice of vendors, configuration, and reverse logistics; see Goetschalckx et al. (2002), Shapiro (2001) and Fandel and Stammen (2004) for extensive work on supply chain optimisation models. Optimisation models consider the supply chain at specific instances in time and do not take on a dynamic

*Corresponding author.

E-mail address: fredrik.persson@liu.se (F. Persson).

view, as is the case with simulation models. Optimisation models often lack an estimation of the variability or robustness of a solution in a stochastic environment. Metrics such as lead-time variability, percentage of on-time delivery and so on, are hard to obtain when using an optimisation model. In a literature review, [Goetschalckx et al. \(2002\)](#) examine seven different modelling approaches for global logistics systems using mathematical programming. Only one model in the review utilises stochastic lead times and only a few included other stochastic characteristics. Stochastic characteristics are an important factor of supply chains. Especially, a stochastic demand is regarded as having great impact on financial performance ([Chwif et al., 2002](#)).

There is a methodological and practical difference in the way optimisation and simulation finds optimal solutions. In optimisation, the solution is dependant on the scenario that defines the experimental domain (cf. [Zeigler et al. 2000](#) for the definition of experimental domain). The optimal solution is only valid for that scenario and the solution will become invalid if the scenario changes. In simulation, it is possible to experiment with a set of different scenarios in order to find an optimal solution. The solution from simulation can therefore be made more insensitive for environmental changes as the optimal solution obtained through optimisation. Simulation in supply chain management can offer a complement to the more prevailing modelling using optimisation models, since simulation is more suited for representing random effects used for simulation-based optimisation and predicting the dynamic behaviour of supply chains.

The supply chain operations reference model (SCOR) is being developed and maintained by the supply chain council (SCC). SCOR is a reference model used as a tool to map, benchmark, and develop the operations of supply chains ([SCOR, 2006](#) and [SCC, 2006](#)). SCOR provides companies with a basic process modelling tool, an extensive benchmark database, and guidelines on how to measure the supply chain operations. Since the process-mapping facilities in the SCOR model is much like the process modelling done in simulation, it is close at hand to merge the two disciplines. The e-SCOR software suite from Gensym ([Gensym, 2005](#)) is a simulation tool that is based on the SCOR model. The e-SCOR software works as a building tool for modelling using the SCOR processes and at

the same time it allows for a simulation analysis of the modelled system.

In this paper, an Arena-based SCOR template for simulation of supply chains is presented. The paper describes the SCOR model and the SCOR template and reports on two case studies where the template has been used in order to simulate the supply chains. The paper is organised as follows; first supply chain simulation is discussed followed by a description of the SCOR model. The third section describes the SCOR template. The focus here is on the functionality of the template. The next section describes the case studies, first the company and its supply chain, then the simulation model in the SCOR template. Last in the paper, the research is summarised and conclusions from the work are drawn. The SCOR template in the next version is also outlined.

2. Supply chain simulation

The main reasons to use discrete event simulation for system analysis in supply chain management are (i) the possibility to include dynamics and (ii) the simplicity of modelling. Discrete event simulation is suited for these kinds of studies where time-dependant relations are analysed. Simulation also has a capability of capturing uncertainty and complexity that is well suited for supply chain analysis ([Jain et al. 2001](#)). [Manivannan \(1998\)](#) provides different examples of supply chain simulation including warehousing and distribution systems and trucking operations, among others.

[Jain et al. \(2001\)](#) point out the model's level of detail as being one of the major difficulties in supply chain simulation. It is not uncommon to simulate at a level of detail that does not match the objective of the analysis. The choice of the level of detail is therefore an important issue in supply chain models. The model must also be credible in order for the results to be useful. Validation of a supply chain model can be a difficult task because of lack of data and lack of system experts. [Manivannan \(1998\)](#) points out the complex nature of supply chains as one of the main obstacles in supply chain simulation. Other modelling challenges that [Manivannan](#) highlights are the missing support for logistic processes in simulation software and unfamiliarity to simulation in the logistics industry. The wide use of optimisation methods in logistics and the fact that many problems have a closed-form solution are other challenges for supply chain simulation.

Banks et al. (2002) discuss what makes supply chain simulation different from other simulation applications. One major difference from e.g. simulation of manufacturing systems is that supply chain models contain information flows together with the flow of materials. The ability to handle different levels of detail is very important in supply chain simulation. Different actors in the supply chain store data in different ways, which makes data collection harder. It is, therefore, difficult to model the whole supply chain at the same (desired) level of detail. The difficulties with different levels of detail together with the size of supply chain models tend to make the model building process longer and more time consuming in supply chain simulation. To experiment with supply chain simulation models often includes a large number of alternative scenarios demanding efficient experimental planning. Validation is another field where supply chain simulation meets difficulties. Subjective methods such as walkthroughs are hard to accomplish on the supply chain level. Sub-model validation is a way around the problem with huge systems since the problem is to get system experts with detailed knowledge about the whole supply chain.

3. The SCOR model

The SCOR model is a reference model with standardised terminology and processes (Meyr et al., 2002). The SCOR model consists of three major parts. Firstly, it is a modelling tool that utilises standardised processes as building blocks. Secondly, it is a set of key performance indicators (KPIs) named metrics, and third it is a benchmarking tool where companies can see how they compare with KPIs to other companies. The SCOR model is also a methodology where the SCOR model documentation can be used as guidelines to improve a company's supply chain. The SCOR model is being developed by the Supply Chain Council (SCC, 2006), a non-profit organisation that also develops other reference models like the design chain operations reference (DCOR) model. The SCOR model version 7 is used throughout this paper.

The SCOR model documentation includes a glossary of the common supply chain terminology. This list of definitions ensures that all companies that work with SCOR will use the same definitions and thus, talk the same language.

3.1. Modelling with the SCOR model

The SCOR model defines processes in three levels. Level 1—process definitions—consists of the five different process types: *Source*, *Make*, *Deliver*, and *Return* for the information flow and physical flow, and *Plan* to coordinate the other four (SCOR, 2006). The process *Source* contains the procurement of goods and services to meet the demand of raw material, components, and other services needed in production. The *Make* process contains the activities that transform products to a finished state. The *Deliver* process contains the transportation and distribution of finished products to the company's customers. The process *Return* is associated with the returning product to suppliers and receiving returned products from customers. The *Plan* process balances the demand and supply of products and resources in the other processes.

At level 2—the process type level—introduces more details to the process building blocks in the SCOR model. At level 2, the model distinguishes between make-to-stock (MTS) products, make-to-order (MTO) products, and engineer-to-order (ETO) products. Each level 1 process is divided into sub categories depending on the product. The *Make* process (M) e.g. is divided into *Make-to-stock* (M1), *Make-to-order* (M2), and *Engineer-to-order* (M3). *Source* follows the same terminology. The *Deliver* process also contains the *Deliver retail products* (D4) apart from the other three. The plan process contains the overall process *Plan supply chain* (P1) and one planning process for each of the other level 1 processes *Source*, *Make*, *Deliver*, and *Return*. The *Return* process is actually two processes; *Source return* and *Deliver return*. These two processes are divided into three sub processes; return of defective products, return of MRO (maintenance, repair, and overhaul) products, and return of excess products. Together with all of these level 2 processes, the complete SCOR model also includes the enabling processes. Enabling processes are supporting the other processes and defines most of the methodologies and define planning and control policies.

The level 3—process category—defines the sub processes of level 2. Level 3 is a generic process map that defines the underlying processes of level 2. It is implied by the SCOR model that the level 3 processes are the same for all companies. As an example, the level 2 process *Source stocked product* (S1) consists of the sub processes; *Schedule product*

deliveries (S1.1), *Receive product* (S1.2), *Verify product* (S1.3), *Transfer product* (S1.4), and *Authorise supplier payment* (S1.5).

3.2. SCOR metrics

The SCOR model defines two types of performance attributes (SCOR, 2006). The customer-facing performance attributes is divided into reliability, responsiveness, and flexibility and the internal-facing attributes are cost and assets. To each of the performance attributes is a set of metrics attached. As an example, the internal-facing attribute named *Cost* is measured by the metrics *Cost of goods sold* and *Supply chain management cost*. The metrics can be broken down to the lower levels (levels 2 and 3) and aggregated from the lower levels up to the top level (level 1). Each process in level 1, 2, and 3 contains a set of metrics that can be used for process evaluation. Each process also contains a list of best practices that can be used in order to improve the metrics. Best practices are the methods and techniques used by the best-performing companies in that segment.

The SCOR model also provides level 1 metric data to be used for benchmarking purposes (SCC, 2006). With this benchmarking database, it is possible to map a company's performance against other companies.

3.3. SCOR methodology

The suggested methodology to implement a SCOR-based working environment consists of four steps (Meyr et al., 2002). (i) Analyse the basis of competition, (ii) configure the supply chain, (iii) align performance levels, practices and systems, and (iv) implement supply chain processes and systems. The second step (ii) configure the supply chain, refers to the modelling using the standardised processes defined by the modelling tool in the SCOR model. This modelling is done from the top and down, increasing the level of detail for each level of the SCOR model. Modelling is only done for level 1 and 2, whereas level 3 is generic and left untouched.

The SCOR model has been used as a tool for supply chain improvements since it first was defined in 1996. Since then, several companies have adopted the SCOR model and methodology. The SCOR model has been highlighted in Sweden during the last couple of years and Ericsson (one of two case

companies in this paper) is one of the leading Swedish companies in SCC.

4. The SCOR template

The SCOR template has been built with Arena, which is a general-purpose simulation package that allows building, verifying, and analysing simulation models. The discrete event simulation software Arena is often used in academics due to the extensive documentation that follows with the software.

The purpose of the SCOR template is to create an easy-to-use, graphical interface for simulation modelling of supply chains. The template utilises the symbols and terminology of the SCOR model as an add on to the simulation software Arena. In this way, the user of the SCOR template only needs to master the SCOR model and terminology and only have limited experience in the use of the Arena software.

4.1. Template structure

A template in Arena is composed of simulation building blocks called modules. The SCOR template has in its current state ten modules. These are; *Plan source* (P2), *Plan make* (P3), *Plan deliver* (P4), *Source stocked product* (S1), *Source make-to-order product* (S2), *Make-to-stock* (M1), *Make-to-order* (M2), *Deliver stocked product* (D1), *Deliver make-to-order product* (D2), and the module *Supply chain resources*, which is a data module that shows the list of the resources of the supply chain and their capacity needed to administer all the simulation resources in the model. The reference among the other modules to the second level of the SCOR model is clear. The SCOR-enabling processes are incorporated into the respective modules. However, the template does not include modules to simulate the return processes (level 2: P5, DR1, DR2, DR3, SR1, SR2, SR3) and the production of engineer-to-order products (level 2: S3, M3, D3). The return processes are not considered core processes in the construction of this template and therefore not included in this version of the SCOR template. Return processes, together with engineer-to-order processes will be incorporated in later versions.

The third level of the SCOR model has been used as a guide to develop the logic of each module. For instance, the level 2 process M1 includes sub processes like *Produce and Test* (M1.3) or *Package*

(M1.4). Therefore, the M1 module of the SCOR template has an Arena process for each of these. The user needs to decide which resources should be modelled in the supply chain simulation model. The resource name and quantity are entered into the module together with the time a product or component occupies the resource. This time can be entered as a fixed number or as a stochastic variable. If this simplified model of the M1.3 *Produce and test* is not in line with the goal of the simulation study, there is always the possibility to model an extensive manufacturing system outside the M1 module. The product flow is simply rerouted to pass through the user-specified manufacturing system instead of the standardised M1.3.

4.2. Supply chain modelling

The modelling is done by dragging and dropping modules of the SCOR template into the model window, connecting them to indicate the flow of entities (products) through the modules (for example: from M1 to D1), and then detailing the modules using dialog boxes. Following the SCOR model, each node (i.e. factory) in the supply chain is represented by the three modules *Source*, *Make*, and *Deliver*. Depending on the placement

of the customer order decoupling point (CODP), these processes are either MTS or MTO. The planning of activities in the node (or factory) is done in the *Plan* processes P2, P3, P4, which plan the *Source*, *Make* and *Deliver* processes, respectively.

The three modules P4, P3, and P2 of a node are directly connected by the flow of entities, the same is for the *Source*, *Make*, or *Deliver* modules. Each *Source* module has a related P2 module that plans its activities, as well as each *Make*, or *Deliver* module is related to a specific P3 or P4 module, respectively. The connection between these two different types of modules (planning and execution) is made by signals. The user specifies in the dialogue box of a particular execution module the related *Plan* module (a list of the existing *Plan* modules is provided) and a unique signal to allow the communication between the two modules is automatically generated. In Fig. 1, an example supply chain (in this case only one node) is used in order to show the connections between planning modules and execution modules, in this case between P4 and D1. Every D1 (or D2) module in a supply chain model must be connected to its unique P4 module. The P4 module drives the planning of the D1 and creates the delivery plan.

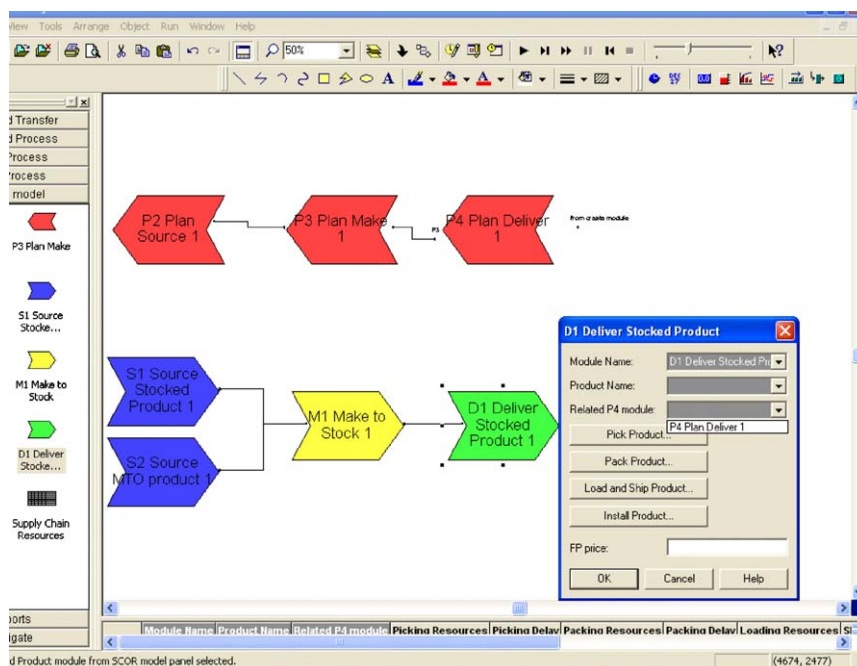


Fig. 1. Connection between P4 and D1.

4.3. Supply chain planning

The SCOR model does not distinguish between a *Plan* process for a make-to-stock product or a make-to-order product, in each P module it is necessary to specify the environment in which the factory is situated.

In order to simulate a MTS environment, it has been chosen to adopt a (R, s, S) policy. The inventory level is inspected every R time units, so that an order is placed to replenish the inventory level up to S (order-up-to point) if the inventory level is below or equal to s (reorder point). Since it is the users who set these three parameters (R, s, S) , this model grants them flexibility in inventory management. This flexibility can be achieved in different ways. If s is equal to S , a fixed order-period system is obtained; whereas, if s is lower than S and R is close to zero (for example only seconds), the inventory position is continuously reviewed and, being the size of each order approximately equal to $S-s$, a fixed order-quantity system is implemented. In a MTS factory, the (R, s, S) policy is applied either to the finished products or to each component.

In a MTO environment, the planning of the *Source*, *Make*, and *Deliver* processes is established on the basis of customer orders and the different lead times (production lead time, supplier lead time, etc.). To prevent the factory from shortages due to received orders of components with the wrong quantity and/or the wrong quality, it is possible to have an initial availability of the finished products and a safety coefficient for the orders of the components, which logically means to order more than what is needed. The components in excess are then considered to establish the next orders.

Fundamental inputs for planning are the customer requirements. A factory (its P4 module) can receive three different types of customer requirements, depending on its position into the supply chain. The orders for the factory closest to the final customer of the chain can be read from an MS Excel file (useful if historical data are available) or be generated with stochastic values (for example: sampled from a normal distribution). The third way of receiving orders is from the upstream factory in the supply chain, or rather from the *Source* module that receives the product.

In both the case of MTS and MTO, the planning is carried out in the following way. An entity is provided with the information of a specific order

(a customer order or an order to fill up the warehouse of finished goods). The information is stored in the entities' attributes. The entity arrives at the P4 module. The presence of the entity generates a new position in the delivery plan (since the P4 is connected to the *Deliver* module). If the order cannot be delivered directly, the entity continues to P3 and P2 in order to start production of the product and to source its components. This flow of entities among the *Plan* modules represents an information flow. The flow of entities between the *Source*, *Make*, and *Deliver* modules represents the physical flow of products. The connection between the two flows is made by variables, signals, and writing/reading in files. The material flow updates variables such as the finished products availability, the *Plan* modules write the plans (into files) considering the value of these variables and send signals to the *Source*, *Make*, and *Deliver* modules. When a signal is received by a *Source*, *Make*, or *Deliver* module, the information in the plan is read and the different processes are consistently executed.

The entities that generate the material flow write the information of their attributes or the value that the variables have in specific instants in an output file. As a result, it is possible to calculate all SCOR metrics (KPIs).

4.4. Supply chain configuration

In order to connect two (or more) nodes in a supply chain the P2 of the downstream company needs to be connected to the P4 of the upstream company. This is done through the connection of delivery plans from the S1 module (see Fig. 2).

In Fig. 2, the S2 module is also connected but that node is hidden in the picture. With this

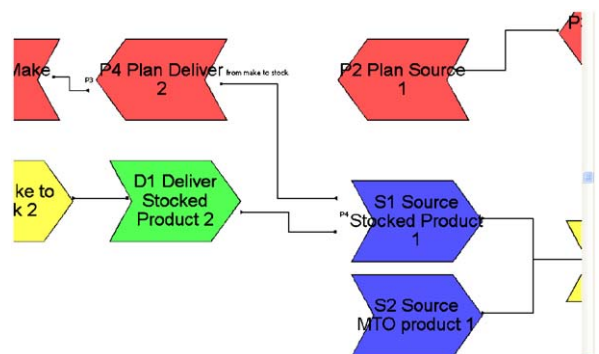


Fig. 2. Two nodes of a supply chain are connected.

way of modelling, it is possible to model a supply chain that contains many companies in a network configuration.

5. Case studies

This paper presents two case studies from Ericsson, Business unit Transmissions and Transport Networks in Borås, and Autoliv, Autoliv Sverige AB in Vårgårda. The cases were used to test the SCOR template in a real environment, using real supply chains and real data. The cases have been described in Selldin (2005), but with the focus on supply chain design within the framework of physically efficient and market-responsive supply chains by Fisher (1997). The study reported in Selldin (2005) makes the latest publication in the project that is now continuing with the addition of the SCOR template.

5.1. Case description: Ericsson

Business unit Transmission and Transport Networks is one of the business units within Ericsson AB. Ericsson produces point-to-point and point-to-multipoint microwave radio solutions for applications in e.g. fixed and mobile telecom networks.

Products are segmented based on demand volume and demand frequency (Selldin, 2005).

5.1.1. Supply chain structure

Suppliers deliver printed circuit boards (CB), components, and modules to Ericsson. The components are surface mounted on printed CBs and supplied to an intermediate inventory. From the intermediate inventory, the CBs are assembled to form a microwave unit. The microwave units are tested and filters are mounted. Filters give each unit the specified radio frequency and are supplied from outside suppliers together with mechanical parts such as covers. Some auxiliary materials are supplied from external suppliers. The supply chain structure as modelled in SCOR is depicted in Fig. 3.

5.1.2. Planning model

Information from customers comes in form of forecasts and orders from Ericsson's own market units as well as directly from certain customers. Actual sales information in the form of customer orders is passed on to suppliers. Even if the suppliers are unable to react within the delivery lead time, they can more easily adapt to the situation and modify their production accordingly.

Ericsson uses Kanban to control the flow of material from the raw material inventories up until

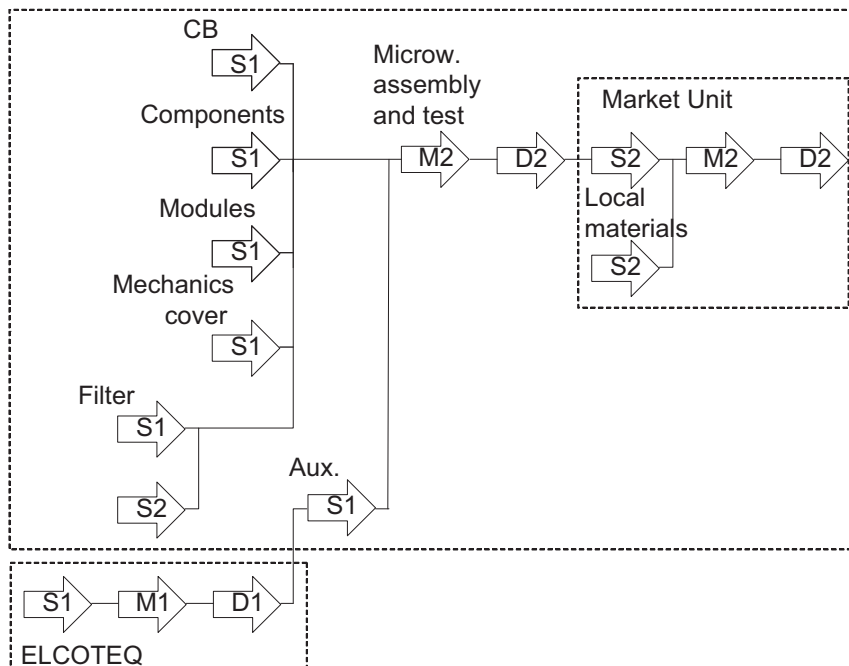


Fig. 3. SCOR model of Ericsson supply chain.

the customer order decoupling point. The production planner inserts firm customer orders in the Kanban circulation and the operator does not have to consider it as two separate flows.

5.1.3. Supply chain simulation model

The Ericsson supply chain SCOR model is easily transformed to a simulation model using the SCOR template; see Fig. 4 for a screenshot of the SCOR template simulation model. Some simplifications are always necessary when adopting a static SCOR model to a more dynamic SCOR simulation model. Here, the Kanban control of the manufacturing process is imitated by the (R, s, S) policy acting as a reorder point inventory control system.

5.2. Case description: Autoliv

Autoliv Sverige AB is a company in the Autoliv group. Autoliv produces automotive safety equipment for car and truck manufacturers. The equipment is mounted in cars and trucks at the assembly line of the manufacturers. Two important products are airbags and safety belts of which there were produced almost one million in 2004 (Selldin, 2005).

5.2.1. Supply chain structure

Autoliv is using several different plants in the production. The first assembly is performed in their own facility in Sweden, Livbag. The assemblies are

then shipped to TECL for a cross dock of materials. From the Autoliv plant in Vårgårda, the finished products are sent to the customer. The supply chain structure as modelled in SCOR is depicted in Fig. 5.

5.2.2. Planning model

As Autoliv is a first-tier automotive supplier for most of their products, the information from the customer is well structured and always electronically transmitted. The actual customer orders are made in sequential call-offs communicated once every minute for each new car planned for production. Each day Autoliv receives a 1-day plan, an 8-day forecast, and a 6-month forecast from the customer. The 1-day plan indicates the number of each article number to be called off during the next day. The 8-day forecast indicates the numbers of cars to be produced each day, but not indicating the product mix, for the following 8 days. The 6-month forecast indicates the expected number of cars to be produced over the next 6 months. The delivery lead time from the sequencing plant is 2–4 h. Therefore, the sequencing plant has to be located close to the customers manufacturing plants.

The production is planned by the MRP system in a 14-day plan for all plants, the plan is then updated daily, and the updated plan is distributed to the plants and suppliers. The plans and the forecasts from the car manufacturer are communicated to the

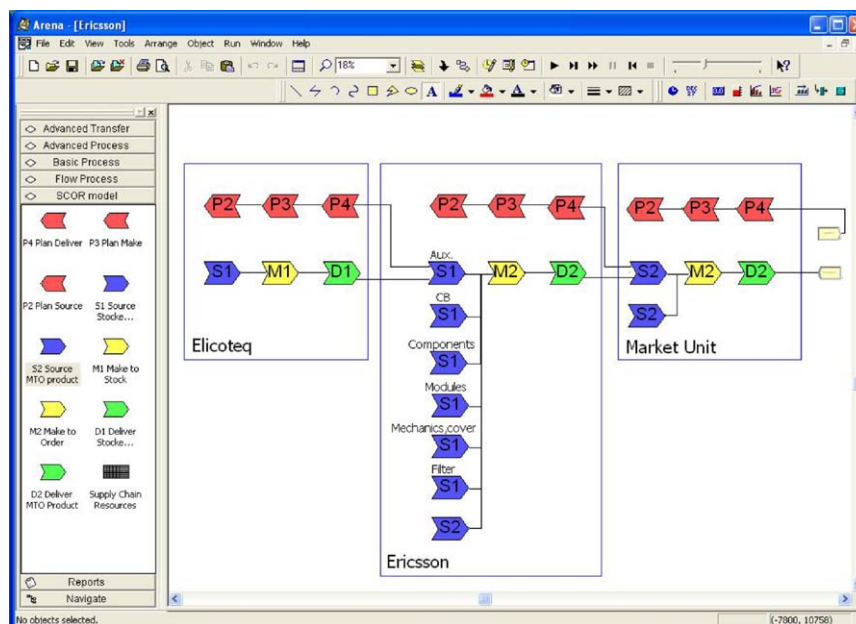


Fig. 4. SCOR template model of Ericsson.

suppliers adjusted for component quantities and lead-time adjustment.

5.2.3. Supply chain simulation model

The Autoliv supply chain SCOR model in the SCOR template is depicted in Fig. 6. Here, the material flow is very straight forward and each node in the supply chain delivers to the next. The flow of materials is shown as the lower connectors from left to right and the information flow is the above flow, through the *Plan* modules.

5.3. Results of the case studies

The two cases presented above were used as means for evaluating the SCOR template in a real

industrial case. Numerical results are omitted due to reasons of confidentiality. The real testing environment highlighted the need for some of the modules to be made more general in order to handle the complexity of the supply chains. The test also showed that the SCOR template is a very useful tool for visualising a supply chain configuration and that it also provides the ability to test “what if” scenarios in a dynamic simulation setting.

6. Conclusions

During the construction of the SCOR template and the implementation of the two cases, it became clear that the SCOR model and the new SCOR template offers solutions to some of the problems

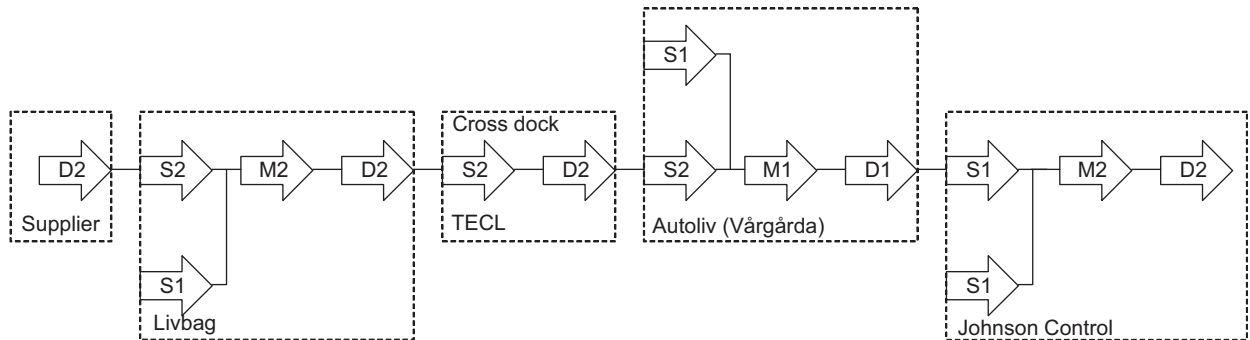


Fig. 5. SCOR model of Autoliv supply chain.

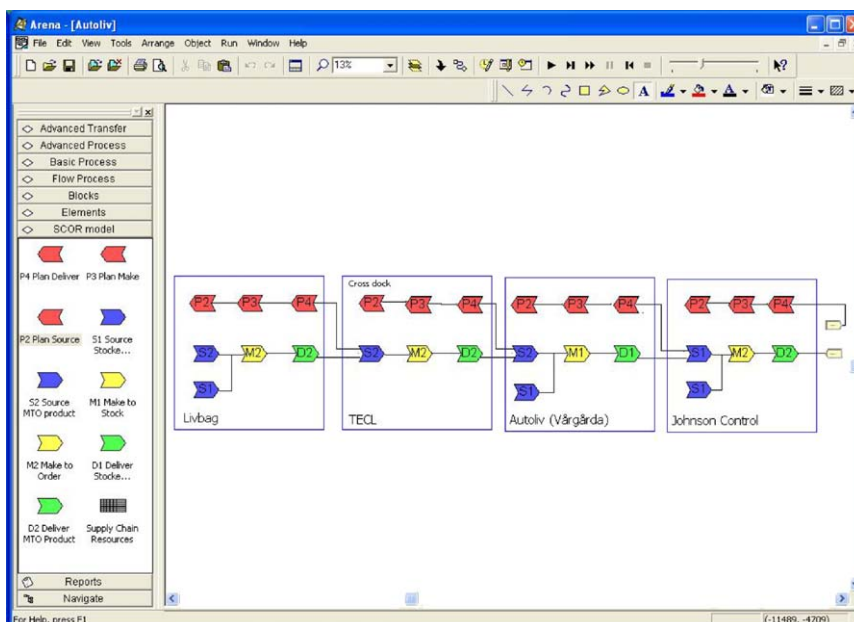


Fig. 6. SCOR template model of Autoliv.

with supply chain simulation, outlined in the two introductory sections. At the same time, new and unforeseen problems surfaced. This concluding section discusses these issues and provides an outlook for future development of the SCOR template.

SCOR differentiates between the processes M1 (MTS) and M2 (MTO) processes. In short, a M1 process indicates that the customer order decoupling point (CODP) is positioned outside the process. Usually, in MTS, the CODP is situated at the finished goods warehouse. The M2 process, on the other hand, places the CODP somewhere inside the process itself. In the beginning of the project, the created SCOR models had a M1 connected with a M2 process to mimic the CODP between them. Soon, it became apparent that this was not the correct way of modelling. Any production process with a CODP is to be modelled as a M2 process, regardless of the fact that the production process before the CODP can be seen as MTS. This was solved using a high-level modelling approach. Only aggregated data were utilised. The motivation for doing this was the fact that SCOR focuses on the supply chain operations, not the operations of the production processes.

The SCOR model provides the modeller with a solution to the problem with different levels of details in simulation models. Since SCOR have predefined levels of aggregation, it is easy to use these levels. Metrics are also already defined for each level, so a company or a supply chain that is already working with SCOR will have no problem to model and simulate its operations.

In simulation model validation, the SCOR model offers a lot of help. Level 3 in the SCOR model provides a validated “generic” model, also adopted in the SCOR template. Since all companies share the same set of processes in level 3, following this approach will create a valid simulation model. Again, if a company is already working with SCOR, all metrics are available and can be used for validation purposes.

This paper has described a new simulation toolbox based on the SCOR model. The proposed

SCOR template is now presented in version 1.0 and the development will continue. The next step is to incorporate the *Return* processes and *Engineer-to-order* functionality.

References

- Banks, J., Buckley, S., Jain, S., Lendermann, P., Manivannan, M., 2002. Panel session: Opportunities for simulation in supply chain management. Proceedings from the 2002 Winter Simulation Conference, 1652–1658.
- Chwif, L., Barretto, M.R.P., Saliby, E., 2002. Supply chain analysis: Spreadsheet or simulation. Proceedings from the 2002 Winter Simulation Conference, 59–66.
- Fandel, G., Stammen, M., 2004. A general model for extended strategic supply chain management with emphasis on product life cycles including development and recycling. *International Journal of Production Economics* 89, 293–308.
- Fisher, M., 1997. What is the right supply chain for your product? *Harvard Business Review* 75 (2), 105–116.
- Gensym, 2005. URL: <http://www.gensym.com>, accessed 2005-01-03.
- Goetschalckx, M., Vidal, C.J., Dogan, K., 2002. Modeling and design of global logistics systems: A review of integrated strategic and tactical models and design algorithms. *European Journal of Operational Research* 143 (1), 1–18.
- Jain, S., Workman, R.W., Collins, L.M., Ervin, E.C., 2001. Development of a high-level supply chain simulation model. Proceedings of the 2001 Winter Simulation Conference, 1129–1137.
- Manivannan, M.S., 1998. Simulation of logistics and transportation systems. In: Banks, J. (Ed.), *Handbook of Simulation*. Wiley, New York, pp. 571–604.
- Meyr, H., Rohde, J., Stadtler, H., 2002. Basics for modelling. In: Stadtler, H., Kilger, C. (Eds.), *Supply Chain Management and Advanced Planning*, second ed. Springer, Berlin, pp. 45–70.
- SCC, 2006. URL: <http://www.supply-chain.org>, accessed 2006-01-10.
- SCOR, 2006. SCOR Version 7.0 Overview. URL: <http://www.supply-chain.org>, accessed 2006-01-10.
- Selldin, 2005. Supply Chain Design—Conceptual Models and Empirical Analyses, Doctoral Thesis, Department of Production Economics, Linköping Institute of Technology, Linköping.
- Shapiro, J.F., 2001. *Modelling the Supply Chain*. Duxbury, Pacific Grove.
- Zeigler, B.P., Praehofer, H., Kim, T.G., 2000. *Theory of Modelling and Simulation*, second ed. Academic Press, San Diego.