

Combining Discrete Event Simulation and Material Flow Analysis based on a Component-Oriented Approach to Industrial Environmental Protection

Volker Wohlgemuth^{1*}, Bernd Page² and Wolfgang Kreutzer³

Affiliations:

¹ FHTW Berlin, Department of Technics II, Industrial Environmental Informatics, Ostendstraße 25, 12459 Berlin, Germany, email: Volker.Wohlgemuth@fhtw-berlin.de Phone: 0049-30 5304-1295, Fax: 0049-30 5304 1299

² University of Hamburg, Department of Informatics, Vogt-Kölln-Str. 30, 22527 Hamburg, Germany, email: page@informatik.uni-hamburg.de

³ University of Canterbury, Department of Computer Science and Software Engineering, Private Bag, Christchurch, New Zealand, email: wolfgang@cosc.canterbury.ac.nz

Abstract:

While eco-balances and material flow networks have become common parts of Environmental Management Information Systems (EMIS) to analyse the material flows within a company, discrete event simulation has not yet been used widely in this domain. This paper argues for combining discrete event simulation with material flow analysis in order to analyse both the environmental and economic impact of business decisions.

After a brief overview of core characteristics and benefits of discrete event simulation and material flow analysis based on so-called Material Flow Networks, a plug-in-based architecture for generating a customised material flow simulator is presented. The benefits of this framework's component-based structure are then discussed in the context of its practical application to the lithography stage in the manufacture of semiconductors. A summary of the framework's most significant aspects and areas of application conclude the paper.

Keywords: material flow analysis, discrete event simulation, software components, plug-in architectures, eco-balances

Software Availability

Software Umberto:

Developed by ifu Hamburg GmbH and ifeu (Institut für Energie- und Umweltforschung Heidelberg GmbH). Contact address: ifu Hamburg GmbH, Große Bergstraße 219, 4.OG D-22767 Hamburg, Germany, Phone: ++49404800090, Fax: ++49404800922, info@ifu.com, 1993. Prices: from 300 € to 9.900 €.

* corresponding author

Software Milan:

Developed by Department of Informatics, University of Hamburg, and ifu Hamburg GmbH. Contact address: University of Hamburg, Dr. Volker Wohlgemuth and Prof. Dr. Ing. Bernd Page, Vogt-Kölln-Strasse 30, D-22527 Hamburg, Germany, Phone ++4940428832426, Fax ++49040428832311. In prototype status and currently free of charge.

1 Introduction: EMIS and Modelling

All business activity creates flows of energy and materials, whose management forms an integral part of how a company interacts with its environment. *Industrial Environmental Information Systems*, also referred to as *Environmental Management Information Systems (EMIS)*, are designed to detect, evaluate and prevent a wide range of environmental dangers and stresses. In more concrete terms *EMIS* consist of computer programs that support management by collecting, documenting and evaluating all relevant data about an enterprise's interaction with its environment and plan, initiate and control all activities related to environmental protection; see Wohlgemuth, (2005), page 144. To prevent or reduce any negative environmental impacts, simple recording of relevant flows, e.g. in the form of an environmental balance sheet, is rarely sufficient. Identifying critical ecological factors and judging the likely effectiveness of corrective measures often requires a *model*, which must reveal all relevant structures and flows.

Modelling is a powerful tool to investigate complex systems, such as those formed by an enterprise's network of material and energy streams. In the context of *EMIS* all aspects of relevance to the environment can be mapped into models, whose behaviour can then be analysed in place of the real systems of which they are representations. Applied Computer Science has developed many software tools for easing this process and in the domain of environmental protection we can now choose from a wide range of methods and software, which, when integrated in an *EMIS*, help users to model flows of energy and materials in a simple and effective fashion.

Since *EMIS* normally focus on ecological aspects of a production system, their capabilities of analyzing economic aspects as well are usually rather limited. By assigning costs or revenues to the flows of energy or materials in a production chain in an *EMIS* based on Material Flow Networks (see section 3) a monetary cost accounting in addition to an environmental assessment could be introduced. Beyond that models covering both economic and ecological viewpoints would be most useful allowing the study of more complex economic-relevant problems, e.g. such as production programme planning or bottleneck detection in production facilities. For such questions, the application of simulation models would be very suitable.

Simulation studies that combine economic with ecological models in order to analyse industrial systems under economic as well as ecological criteria are quite rare. A few more or less related approaches have been recently published, e.g. a model-based integrated toolbox that allows linking hydroclimatic and surface and groundwater hydrological analyses to economic and environmental impact models (see Quinn, et al. (2004)), the STEAP model (STeel Envi-

ronmental Assessment Program) for analyzing CO₂ taxes (see Gielen, et al. (2002)), an online system for simulating industrial accidents (see Bellasio, et al. (2005)), various approaches for simulating activated sludge wastewater treatment plants (see Gernaey, et al (2004)), eco-balances for supporting environmentally and economically sound regional development (see Lenz, et al. (2003)), and a decision support system for integrated agricultural production (see Perini, et al. (2004)). In contrast to these approaches the work reported in this paper focuses on integrating discrete event simulations of industrial production with flow-based descriptions of material and energy distribution and dissipation in Material Flow Networks. It also discusses relevant issues from an environmental perspective.

In this paper we focus on a novel approach of integrating the well established discrete event simulation methodology for modelling industrial production processes with the rather novel ecological material flow analysis approach for describing material and energy flows, thus combining both economic as well as environmental viewpoints in the same modelling framework. Our paper is organized as follows: In section 2 we briefly cover discrete event simulation as a modelling and planning tool for industrial production and logistics processes under economic aspects (e.g. bottlenecks, machine utilization, maintenance, inventory etc.). Material flow analysis is a rather recent approach to model material and energy flows in production chains, mainly under ecological aspects (section 3). In section 4 we outline how we can integrate simulation capabilities into Material Flow Networks for industrial environmental protection purposes. However, one drawback of this approach is that the simulation model can only run locally; i.e. as part of a single transition of a Material Flow Network. In the central section 5 of this paper we go one step beyond and demonstrate how we can combine material flow analysis and discrete event simulation within a single software environment by introducing a component-based framework called Milan. This framework assists users in performing material flow analyses by means of a discrete event simulation model without any need to create an extra material flow model or for changing the software environment. A material input/output account, which is based on the same model used for compiling simulation results, can be created for the entire model. To test the design and applicability of the component-based combination of discrete event simulation and analysing material flows, a case study has been performed at a semiconductor manufacturer (section 6). In the final section we give a summary and conclude that the software described in this paper creates customized Environmental Management Information Systems (*EMIS*) in which production processes and all kinds of energy and material flows can be analysed in terms of both economic measures and their environmental impact.

2 Discrete Event Simulation of Industrial Production

Discrete Event Simulation has long been a popular technique for studying industrial processes, but it is also widely used for planning purposes – especially for evaluating different design alternatives in a production process; see Law and Kelton, (2000) and Oakshott, (1997). Discrete event simulation's methodology is based on the notion that, once properly validated,

we can use *models* to help us in answering difficult questions about complex systems. A good simulation model can significantly improve our understanding of a system's behaviour.

In many dynamic processes, particularly in industrial contexts like manufacturing, transportation and inventory management, system states change at discrete points in time (i.e. events), rather than through continuous state fluctuations. Queues in front of a group of machines or other manufacturing facilities are an example, since their lengths (number of items queuing for service) only changes when items arrive or depart. Such *queuing networks* typically consist of discrete components, such as machines and workpieces, whose behaviours cause state changes through discrete events, which will typically be dispersed randomly along a model's timeline. In such discrete event simulations it is often desirable or even necessary to treat many model components as individuals, each with their own properties and processing history. During a simulation models are "moved" along a simulated timeline, for which a model monitor must administer a simulation clock and an event list. Event lists are used to record future events, and to allow the model monitor to repetitively select and remove the *next imminent* one for execution. The monitor then updates its clock to the relevant model time, which means that clock values jump in irregular intervals - from current model time to the time of the next event. In this way all intermediate time values are skipped and all activities' durations are simply modelled by scheduling corresponding start and stop events. The model monitor also transfers control to a program that performs relevant changes to model states; e.g. the start of a processing activity. Such programs are called *event routines*, and their execution may schedule further events (e.g. the end of a processing activity) for some time in the simulation's future. These are placed on the model's agenda. Chapter 2 of Page and Kreutzer, (2005) offers a concise introduction and summary of discrete event simulation's foundations.

In contrast to its popularity for modelling manufacturing processes and information flows and logistics, discrete event simulation has not yet been widely applied to the analysis of environmental issues; see Neumann, (1985) or Wohlgemuth, (2000). In our opinion this can be traced to a lack of familiarity with discrete event simulation by most experts in environmental analysis, rather than to any conceptual inadequacies of the method itself. The increase in model complexity caused by using discrete event simulations in *combination* with other established techniques (e.g. Material Flow Networks for material flow analysis) may have further contributed to this state of affairs.

3 Material Flow Networks and the Software Tool *Umberto*

Material Flow Networks are a special form of Petri-Nets that model material and energy flows in multi-stage production systems. Petri-Nets are well established in Computer Science and offer an interesting perspective on modelling and analysing complex systems. When adapted and modified for modelling the *environmental* aspects of an enterprise's operations, Petri-Nets have become known as *Material Flow Networks*.

Material Flow Networks were originally developed at the University of Hamburg (see Möller, (2000)) to describe the flow of materials and energy caused by economic activities; see e.g.

Schmidt, et al. (1997) and Wohlgemuth, et al. (1997). Recording and analysing the environmental impact of such flows is one of the most crucial tasks of effective environmental management; see Rautenstrauch, (1999).

While the development of Material Flow Networks has been inspired by concepts from several disciplines, its two most important roots are Computer Science's Petri-net theory and Business Administration's double-entry bookkeeping and cost accounting. Material Flow Networks are Petri-Nets, which consist of transitions, places and arrows. Petri-Nets can therefore be formalized as 3-tuples $N = (T, S, F)$, $F \subseteq (S \times T \cup T \times S)$, where T is a set of transitions, S is a set of places, and F is a set of arrows between them; see Baumgarten, (1996). In diagrams *transitions* are shown as squares and indicate the location of material or energy transformations. *Places* separate transitions and serve as inventories, which are depleted and replenished by flows. In diagrams we draw them as circles. *Arrows* show flows; i.e. how items such as information, energy and material move between transitions and places. *Figure 1* shows an example. Note, that in this sense the Material Flow Network approach uses the structure of Petri-Nets but does not refer to more dynamic characteristics of Petri-Nets like fairness, liveness etc.

Figure 1: A simple Material Flow Network

The network in figure 1 describes only a Petri-Net's structure, and further detail is needed to specify its behaviour. Such behavioural specifications are of particular interest in models of computer systems, computer networks, or communication systems, where Petri-Nets are most often used. Since *Material Flow Networks* serve as accounting systems for material and energy flows, double-entry-bookkeeping can be used to describe their dynamic behaviour; see Moeller (2000). From this perspective arrows can be viewed as accounts (of material and energy flows) and places keep track of all relevant stocks. The first step in data collection therefore records all stocks at the beginning of some time interval (i.e. the starting inventory) and the second step determines all flows during this period. Stock levels at the end of a period (i.e. the final inventory) can be calculated from these numbers. The result is a material and energy model that can serve to inform our analysis in different ways.

Material Flow Networks offer a wide range of representations for complex material flows and can be used to support a whole range of environmental management tasks. For example, they can be employed to gain deeper insights into production, consumption, transportation, waste treatment and other such processes. Transitions describe an activity or task, which is entered by all its required materials and generates new or modified materials as its output. In this way transitions link material consumption and production. So-called *transition specifications* can summarize underlying processes in terms of mathematical expressions or algorithms, and programs are used for their computation. In addition to decomposing large systems and calculating a missing value, transition specifications also allow seamless change from a descriptive to a predictive model.

We can distinguish between different types of transition specifications. The simplest are so called *production coefficients*, which define linear relationships between a process' inputs and outputs. In this context production coefficients become scale factors and knowing a single flow in the transition's immediate vicinity is sufficient to calculate the strengths of all other flows. *Figure 2* illustrates the procedure.

Figure 2: A simple transition specification using production coefficients and its application to building; based on Schmidt, et al. (1997), page 775.

A slightly more complex but somewhat more flexible method uses a set of expressions to describe relationships between input and output flows of a transition. This makes it possible to model non-linear dependencies. By using an if-expression this method can also range across several cases or profiles and embedding an algorithm within a transition gives us ultimate flexibility. For example, we could use fuzzy logic within a transition that is part of a dye works model, or we could code a discrete event simulation of a bottle filling process, that would be run whenever the calculation algorithm within the Material Flow Network touches the relevant transition. Without violating any of their theoretical foundations, using scripts to attach complex rules or models to a transition can extend Material Flow Networks further (see Page and Wohlgemuth, (2002)).

Material Flow Networks' most attractive feature is that they can seamlessly combine the compilation of input/output eco-balances for a company, industrial plant or production process, with an analysis of material flows associated with given products or services. In addition, they offer a graphical representation for visual analysis of distributed material and energy flows.

Software support for Material Flow Networks must meet a number of requirements. For instance, the software must be capable of modelling complex production processes from different fields of application, such as chemistry or engineering, and must provide flexible data management for updating and extending the model. Different ways of data interpretation and display must also be supported. The Umberto^{®1} system, which entered the market in the mid-1990s, was the first suite of programs that tried to meet these goals. Umberto[®] is a powerful and user-friendly material flow analysis tool. It runs under the Microsoft's Windows[®] family of operating systems and meets software ergonomic standards; see Schmidt, et al. (1997). It also offers several interfaces to other programs, one of which allows the specification of transitions. In combination with complex algorithms it can embed sub-models into Material Flow Networks that have been developed outside of Umberto[®]. This can be done through a script written in any of the languages supporting *Microsoft's Active Scripting Architecture*. This scripting technique extends Material Flow Networks in a powerful way, and its users can specify a transition without being restricted to Umberto's[®] defaults.

¹ For more information about Umberto[®] visit <http://www.umberto.de/en>

4 Augmenting Material Flow Networks with Simulation

We have used Umberto's[®] capability for associating transitions with models to integrate discrete event simulation into material flow networks and return a model's results to Umberto[®]. Implementation of this technique is based on Microsoft's *Active Scripting Architecture*. This strategy has enabled us to embed a discrete event simulation of a simple inventory model within a material flow network's transitions (see Page and Wohlgemuth, (2002)). To significantly reduce implementation effort the *Delphi* version of the DESMOJ discrete event simulation framework, developed at the University of Hamburg (see Page and Kreuzer, (2005)) was used for coding the simulation. This framework pre-packages all required discrete event simulation functionality, such as random number generation, model time management, event handling and statistical reporting. Only a few extensions for automatic recording of typical performance measures of material handling and transformations at each machine, such as production quantity, machine hours, setup times, etc., needed to be made to this simulation framework. Compared to simple Material Flow Networks, this simulation-augmented approach takes a more job-related point of view and allowed us to observe all material transformations at each machine and for each job.

Inventory systems are of great economic importance to many industrial sectors. They also have some interesting environmental implications. The main economic goals of inventory systems are to minimise inventory costs and maintain a high level of service; e.g. in terms of the speed with which an order can be filled. Mathematical methods of Operations Research can help balance these goals. Because of their complexity and their largely stochastic nature, computer simulations are frequently used to find optimal trade-offs between costs and service. Since filling orders and replenishing stock leads to product deliveries (mainly by trucks) that carry an environmental burden (such as pollution and noise), different inventory strategies may have different environmental impact. Reducing pollution can be achieved by decreasing the number of deliveries, reducing the distances for transportation (e.g. by using local suppliers), or by selecting environmentally friendlier vehicles. Choosing an ordering policy therefore becomes a strategic business decision with economic as well as ecological impact, and it makes sense to link simulations of economic factors to Material Flow Networks, so that both environmental and economic indicators can inform inventory strategies. By linking these two approaches we can answer questions like:

- What emissions are caused by the current inventory strategy?
- Can emissions be reduced by using alternative strategies?
- How large is the cost increase or decrease for each strategy?
- Can the required level of customer service be maintained when inventory strategies change?

In the following example we attach a simple inventory simulation to a transition specification in an Umberto[®] Material Flow Network. This inventory simulation only runs when a relevant transition occurs. All other energy transformations are described with "normal" Umberto[®] methods. The simulation model mirrors some aspects of a fictitious clothing company, which

sells trousers, shirts and shoes. Orders for clothing arrive by phone and their delivery is delegated to a conveyance company. The clothing company guarantees delivery within one working day to their clients. Shirts and trousers are normally sent by ship, except for stock shortages when emergency delivery by plane is required to ensure that the desired service level is met. Shoe deliveries are sent by airfreight. A reorder activity for up to the maximal inventory level is triggered whenever a minimum stock threshold is reached. Client demand is stochastic and follows a normal distribution. All transportation times are also normally distributed. Cost data consist of inventory costs, indirect and direct order costs, costs of shortage (i.e. caused by failure to meet service levels), and transportation costs. The model investigates two scenarios – one using a solely economically motivated inventory policy and one which takes environmental considerations into account. In the second scenario transportation distances are reduced and more environmentally friendly modes of transportation are chosen. This makes order deliveries more eco-efficient and should be achievable without necessarily increasing inventory costs or decreasing the company's service level.

The above problem statement is a typical challenge for any trading company looking for environmentally friendly optimisation of order logistics. Instead of DESMOJ, the case study's implementation of the two scenarios made use of a different Delphi-based discrete event simulation framework. The interface which permits calling Umberto's[®] functionality from within the simulation framework used Microsoft's *Component Object Model* (COM) and is described in section 5 of this paper. In the example, the simulation is activated whenever Umberto's[®] computational algorithm reaches Transition T4 (see *Figure 3*).

Figure 3: Linking a Discrete Event Inventory Simulation to a Material Flow Network;
see Wohlgemuth, et al. (2001).

Relevant information, e.g. simulation run length, which corresponds to the accounting period of the Material Flow Network, is exchanged through the COM interface and control is transferred to the simulation model. After a simulation run finishes, the mean number of deliveries per transport, the number of products requested and supplied, and the predicted costs are returned to Umberto[®]. Based on this information Umberto[®] will then determine all relevant transport emissions. As a result we can generate both *economic* and *ecological* planning data for different patterns of order logistics and inventory management. More details can be found in Wohlgemuth, et al. (2001).

One drawback of this way of combining simulation with material flow networks is that the simulation model can only be run locally; i.e. as part of a single transition. No information from other parts of the network can be included. A further constraint is that the accounting period of the material flow network must correspond exactly to the length of the simulation run. Programming the model also requires both methodological knowledge and coding skills, and we must make sure to record all relevant changes in energy and material flows during a simulation. Apart from the need for recording this data it is quite rare that the model's behaviour will be influenced by the strength of the flows.

5 Component-based Architecture of the Material Flow Simulator Milan

Modern environmental information systems are typically stand-alone programs that allow only few extensions and modifications with new methods and user-defined functions. Any substantive extension is constrained by what a particular vendor foresees and permits. Existing EMIS architectures are rarely open enough to allow replacement and changes to much of their built-in functionality. Umberto[®], for example, permits script-based augmentation for computing additional information (e.g. missing flows). Replacing, for example, the software's flow management strategy or its material flow editor with new components carrying user-defined functionality, however, is not easily possible - and neither can new network-wide computational patterns be integrated into the system. Although they sometimes put vexing limits on what users can do, such constraints are quite typical for the current generation of EMIS.

In the remainder of this paper we propose and describe a conceptually different approach, which permits the creation of customized application frameworks from EMIS-specific components, similar to the way in which plug-in extensions can augment Java programming environments; see e.g. *Eclipse* – Clayberg and Rubel, (2004). In such a framework components can be freely replaced and applications remain independent of any existing EMIS. Domain-specific components derive environmental information from discrete event models, which simulate relevant flows of materials. By combining many different domain-specific components into a common framework, such a component-based architecture for material flow simulations enables a user to choose functionality targeted at a specific area of application. Composition and communication between components is based on well-defined interfaces rather than scripts, and these interfaces' abstractions form "domain languages" which must be used by component designers who wish to create plug-ins augmenting the framework's functionality.

5.1 Basic Ideas

To combine material flow analysis and discrete event simulation within a single software environment, and without limiting communications between the two methods to a single transition, we have over the last three years developed a component-based framework using Microsoft's COM-Interface. The Milan framework provides basic functionality for discrete event simulations that study dynamic systems' characteristics and behaviours. For example, industrial production can be examined to pinpoint critical areas and bottlenecks. The framework also assists users in conducting material flow analyses without any need to create a new model or for changing the software environment. This means that a material input/output account that is based on the same model as the one used for collecting simulation results can be created for an entire model or any single model component. In this way material flow analysis and discrete event simulation form an integrated environment, which, for example, in the case of a semiconductor manufacturer (see section 6) allows us to ask questions like:

- Can we save energy if we buy a new machine without changing throughput?

- Can we increase the utilization of a machine by changing the production process without increasing waste accumulation or energy demand?
- Can we produce more wafers without buying more lacquer, and in compliance with legal limits on solvent emissions?

To reach this goal we have tried to avoid “reinventing the wheel” yet again; i.e. our component framework uses as much existing functionality as possible. For example, a typical feature of current software for managing material flows is a material administration component, through which users specify all materials that are needed for a production process and what their relevant properties are (see the Industrial Environmental Management Information System Umberto[®] for an example). A suitable material administration component should therefore be plugged into our framework. To provide functionality for discrete event simulation we connect a simulation engine based on the DESMOJ discrete event simulation framework; see chapter 10 of Page and Kreutzer, (2005). In this way all functionality of a particular material flow simulator is defined through commands and components provided by plug-ins registered for a specific application. The framework itself only needs to enforce relevant communication rules and serve as a mediator for the registered plug-ins. If a user is not satisfied with the commands and model components a particular plug-in can offer, she can replace it with a different one, which she may have purchased or written herself.

Figure 4: Building an Application Framework for Material Flow Simulations from Component-Based Plug-ins

5.2 Core Concepts

The described Milan framework derives all of its functionality from registering and deregistering suitable plug-ins. Each plug-in’s commands and model components are only accessible to an application it is registered for. A user can access available commands through an application’s menu structure, or - if allowed by the plug-in developer - via a button on a toolbar (which is automatically created for each plug-in). In a similar fashion a framework application can present modelling functionality through its graphical interface. If a plug-in requests it, windows can be dispatched to act as containers for ActiveX-controls (see *figure 5*) and the application will serve as a mediator for all user actions.

Figure 5: Commands and Components for Plug-ins - as seen by a Framework Application

Since work on the component-based material flow simulator started some years ago², the idea for the plug-in architecture was inspired by Microsoft’s *Component Object Model* (COM). In this architecture any plug-in that wants to be registered for a framework application must im-

² See Box (1997) for more details about COM.

plement a specified interface, details of which are published in a so-called *type library*, which is itself registered at the operating system level. To register a plug-in the framework application walks through all available COM-Objects and looks for the specified interface. It then creates a COM-Object for each command and each model component which the relevant plug-in defines. Since all functionality is accessible through buttons or menu entries, all commands and components for creating a suitable model become available to the application (see *figure 5*).

To provide an integrated view, based on the same model, of both simulation performance (e.g. through measures like throughput, utilization, adherence to delivery dates etc.) and material flow analyses, we have developed a strategy in which so-called *material indicators* play an important role. These indicators combine the event-oriented worldview of the simulation engine (see chapter 10 of Page and Kreutzer, (2005)) with methods for keeping track of materials. More specifically, a material indicator defines which materials are needed as inputs to or emitted as outputs from model components. Returning to the example of a semiconductor manufacturer, a material indicator for a coater can be defined who reserves a certain amount of lacquer as its required input at the start of a task. At the end of the task any remaining lacquer is then recycled and some solvent is registered as an output flow (see *figure 6*).

Figure 6: Relationships between Material Indicators and Events for a Coater

Indicators are available through a special plug-in. Their implementation is based on software engineering's well known *observer* design pattern; as described in Gamma et al. (1998). Whenever an event involving a model component occurs inside the simulation engine, all indicators observing the model component must decide what materials to reserve for this type of event. Users can develop their own indicators if no existing ones satisfy their specific needs. In this manner material input and output balances can be created for all model components and the entire model.

5.3 Available Plug-ins

We have developed a number of plug-ins for use in the Milan framework. These can be assigned to one of two groups. The first group's plug-ins offer functionality for discrete event simulations and include:

- a plug-in for creating random variates from a specified probability distribution; e.g. exponential, normal, uniform etc.
- a plug-in for basic time and event list management; i.e. a so-called simulation engine, which provides a scheduler, a clock, an event list handler etc; see chapter 6 of Page and Kreutzer, (2005).
- a plug-in that allows model creation in a graphical style; i.e. a model editor.

- plug-ins for specific application domains, which contain specialized model components for this area of application; e.g. model components for production systems like buffers, work stations, fork lifts, resources etc..
- a plug-in for analysing and visualising simulation results; e.g. pie charts and x/y-plots for single model components or entire models.

The second group of plug-ins deals with aspects of material flow analysis. This includes:

- a plug-in for the administration of all materials used in the model which were obtained from the Umberto[®] material flow analysis software. This plug-in contains commands for exporting simulation results and model structures to Umberto[®], for further processing.
- a plug-in with material indicators for the integration of discrete event simulations and material flow analyses, as detailed above.

All these plug-ins can be integrated into the application framework and can theoretically be replaced with alternative plug-ins; as long as their developers adhere to a few interface rules. However, we must take care to preserve logical dependencies. For example, because they are intentionally designed to work with the *discrete event* simulation engine, it would not make much sense to use material indicators when a user replaces the model's simulation engine with one supporting *continuous time* simulations.

6 A Case Study from the Semiconductor Industry

An empirical study was performed at a semiconductor manufacturer in order to test the design and applicability of the proposed combination of discrete event simulation and material flow analysis. Since the manufacture of semiconductor components is a complex process, during which many planning and control problems need to be solved, this domain is well suited to an empirical test of the material flow simulator's implementation. Using simulation as a tool for decision support has a long history in the semiconductor industry (see e.g. Yücesan, et al. (2002), volume 2, pages 1357-1437) and many expensive materials (e.g. developer chemicals, paints, solvents, acids), whose emission into the ecosystem could cause substantial damage and whose safe decommissioning is an expensive process (see Vishay (2002)), are used in semiconductor production. Testing our framework with a model of semiconductor production including both economic (e.g. bottlenecks detection, maintenance planning, machine acquisition etc.) and environmental aspects (e.g. emissions, raw material and energy consumption etc.) was therefore much help in the framework's further development. Because of tight time constraints and a desire to reduce model complexity, only the production stage of *lithography* was chosen for this experiment.

Figure 7: Results of a Lithography Simulation and its Material Flow Analysis

Since they determine a semiconductor's geometry, the production processes in the lithography stage play an important role in structuring wafer surfaces. Lithography uses light-sensitive photo paints, which are deposited on a wafer and then exposed and developed through lithography masks. Since this happens at multiple layers, wafers must be routed through the lithography stage multiple times. Exactly how many passes are needed will depend on the type of semiconductor. Although neither simulation nor material flow analyses had ever been used in this part of the company, using them in combination proved advantageous for pinpointing bottlenecks and for observing all legal requirements on solvent emissions.

In the experiment we used our material flow simulator to model the lithography process and constructed a plug-in for this domain. This focussed attention on semiconductor-specific model components, which helped us to map typical lithographic and other processes to a simulation model (see *figure 7*). During this process we created a number of different model components. *Coater*, *Stepper*, *Developer* and *Repacker* were represented by server components, two measuring devices were mapped to a *MeasuringDevice* component, and entry and exit point components were used to define system boundaries. A *RoutingControl* component specified wafers' paths through the model and a *Connection* component was used to bind all the other components together. Developing these components and integrating them into our application framework was achieved in a relatively timely and rapid fashion. Other existing components, such as the model editor, the discrete event simulation engine, and various material management functions could be directly reused. To sow residence times and other semiconductor-specific characteristics, some extensions to the existing analysis plug-in were made.

Using the material flow simulator framework significantly reduced the effort of combining simulation and material flow analyses of the system under investigation. In the past such investigations could not have been made with a single piece of software, and would instead have required the use of separate programs, which would have caused a significant increase in development, maintenance and analysis effort.

After analysing the production of 6 inch wafers, a second model for investigating 8 inch wafers was built on request of the host company, since it planned to change wafer production to this larger size. The model allowed the analysis of various aspects of such a change; e.g. changes in throughput and bottleneck shifts, and changes to the consumption of energy and raw materials. All these had ecological consequences as well as effects on the costs of wafer production. The investigation also highlighted a further advantage of combining simulation with material and energy flow analyses. Since both techniques use a common model, we can explore both environmental and economic impacts of changes to production techniques and material flows at *both* tactical and strategic management levels.

7 Summary and Outlook

Discrete Event Simulation and Material Flow Analysis are complex techniques, whose mastery requires sound technical knowledge. Both methodologies are well served by specialized

software. Although they have different roots, they share a common point of view and make use of computer models of production and distribution processes. Traditionally two separate models have been required to feed information to each of these methods, and different software tools were employed for their capture. As a result, two separate models needed to be maintained, which often showed high levels of redundancy. Assuring model consistency was a crucial but far from trivial task.

This paper has suggested a novel approach for integrating Discrete Event Simulation and Material Flow Analysis into a component-based framework that is based on a *single* model. This has the potential to significantly ease both a model's development and its maintenance. The software described in this paper creates a customized Environmental Management Information System (EMIS) in which production processes and all kinds of energy and material flows can be analysed in terms of both economic measures (e.g. throughputs, utilizations, bottlenecks etc.) and their environmental impact. The component-based architecture of the software permits effective reuse of existing components, whose functionality can then be accessed through a small number of well documented interfaces. From this perspective the described simulation-based EMIS even offers a generalized framework for cooperation between components which were developed by different authors.

An example for cooperation between the Umberto[®] EMIS and some simulation components patterned after Microsoft's COM architecture was used to show the effectiveness of this concept. To develop good plug-in components we must identify suitable linguistic abstractions for relevant application domains, since such "application languages" suggest suitable interfaces and signatures. A more detailed example for this process can be found in Wohlgemuth, (2005).

References

- Baumgarten, B., 1996. Petri-Netze. Grundlagen und Anwendungen. Spektrum Publ.: Berlin, Heidelber, New York.
- Box, D., 1998. Essential COM. Addison-Wesley: Reading (MA) (United States).
- Clayberg, E. and Rubel, D., 2004. Eclipse: Building Commercial-Quality Plug-ins. Addison-Wesley Professional: Reading (MA).
- Gernaey, K.V., et al. 2004. Activated sludge wastewater treatment plant modelling and simulation: state of the art. Environmental Modelling & Software 19(9), Pages 763-783.
- Gielen, D. and Moriguchi, Y., 2002. Modelling CO₂ policies for the Japanese iron and steel industry. Environmental Modelling & Software 17(5), Pages 481-495.
- Heffter, J., 1999. Konzepte und Realisierung einer Einbettung von Tourenplanungsmodellen in Stoffstromnetze. Diploma thesis, Department of Computer Science, University of Hamburg.
- Law. A. M. and Kelton, W.D., 2000. Simulation Modeling and Analysis. McGraw-Hill: New York.

- Lenz, R. J. M. and Beuttler, A., 2003. Experiences with GIS-based planning tool for spatial eco-balances. *Environmental Modelling & Software* 18(6), Pages 581-585.
- Möller, A., Häuslein, A., Rolf, A., 1997. *Öko-Controlling in Handelsunternehmen: ein Leitfaden für das Stoffstrommanagement*. Springer Publ.: Berlin, Heidelberg, New York.
- Möller, A., 2000. *Grundlagen stoffstrombasierter Betrieblicher Umweltinformationssysteme*. Projekt Publ.: Bochum.
- Möller, A., Page, B., Rolf, A., Wohlgemuth, V., 2001. Foundations and Applications of computer based Material Flow Networks for Environmental Management. In: Rautenstrauch, C., (Ed.), *Environmental Information Systems in Industry and Public*, Pages 379-396. Idea Group Publishing: Hershey.
- Neumann, K., 1995. *Betriebliche Umweltschutzplanung mit Hilfe der Simulation. Ein integrierter Planungsansatz mit Anwendung auf einen landwirtschaftlichen Betrieb*. Peter Lang Publ.: Frankfurt.
- Oakshott, L., 1997. *Business Modelling and Simulation*. Pitman: London.
- Page, B. and Kreutzer, W., 2005. *The Java Simulation Handbook – Simulating Discrete Event Systems in UML and Java*. Shaker Publ.: Aachen.
- Page, B. and Wohlgemuth, V. 2002. Linking Economic Optimisation and Simulation Models to Environmental Material Flow Networks for Ecoefficiency. In: Hilty, L.M., Seifert, E.K., Treibert, R. (Eds.), *Information Systems for Sustainable Development*, Pages 94-108. Idea Group Publishing: Hershey.
- Perini, A. and Susi, A., 2004. Developing a decision support system for integrated production in agriculture. *Environmental Modelling & Software* 19(9), Pages 821-829.
- Quinn, N. W. T. et.al., 2004. Model integration for assessing future hydroclimatic impacts on water resources, agricultural production and environmental quality in the San Joaquin Basin, California. *Environmental Modelling & Software* 19(3), Pages 305-316.
- Rautenstrauch, C., 1999. *Betriebliche Umweltinformationssysteme*. Springer Publ.: Berlin, Heidelberg, New York.
- Schmidt, M., Möller, A., Hedemann, J., Müller-Beilschmidt P., 1997. Environmental Material Flow Analysis by Network Approach. In: Geiger, W., Jaeschke, O., Rentz, O., Simon, E., Spengler, Th., Zilliox, L., Zundel, T., (Eds.), *Umweltinformatik '97, 11th International Symposium of the German Society for Computer Science (GI)*, Strasbourg, France, Pages 768–779. Metropolis: Marburg.
- Vishay (2002). *Umwelterklärung 2002*. Vishay Semiconductor Itzehoe GmbH. Itzehoe.
- Wohlgemuth, V., Page B., Meyer, U., Möller, A., Rolf, A., Schmidt, M., 1997. Computer based support for LCA's and company ecobalances using Material Flow Networks. In: Alef, K., Brandt, J., Fiedler, W., Hauthal, O., Hutzinger, D., Mackay, D., Matthies, M., Morgan, K., Newland, L., Robitaille, H., Schlummer, M., Schüürmann, G., Voigt, K., (Eds.).

- Eco-Informa '97, Information and Communication in Environmental and Health Issues, Munich, Germany, Pages 179-191. Eco-Informa Press: Bayreuth.
- Wohlgemuth V., Page, B., Heffter, J., 2000. Coupling a Material Flow Network with an Optimisation Model. Proc. *Simulationstechnik*, Pages 389-394. SCS European Publishing House: Ghent.
- Wohlgemuth V. and Page, B., 2000. Einbettung von Transportmodellen und diskreten Simulationsmodellen in Stoffstromnetze. In: Cremers, A.B., Greve, K., (Eds.), Environmental Information for Planning, Politics and the Public. 14th International Symposium for Environmental Protection, Pages 345-360. Bonn.
- Wohlgemuth, V., Bruns, L., Page, B., 2001. Simulation als Ansatz zur ökologischen und ökonomischen Planungsunterstützung im Kontext betrieblicher Umweltinformationssysteme (BUIS). In: Hilty, L. M., Gilgen, P. W., (Eds.). Sustainability in the Information Society. 15th International Symposium for Environmental Protection, Pages 999-1008. Zuerich.
- Wohlgemuth, V., 2005. Komponentenbasierte Unterstützung von Methoden der Modellbildung und Simulation im Einsatzkontext des betrieblichen Umweltschutzes. Shaker Publ.: Aachen.
- Yücesan, E., Chen, C.-H., Snowdan, J.L., Charnes, J.M. (Eds.) 2002. Proceedings of the 2002 Winter Simulation Conference. Association for Computing Machinery: New York.

Vitae:

From 1991 to 1997 *Volker Wohlgemuth* studied Computer Science at the universities of Hamburg (Germany) and Christchurch (New Zealand). In 1997 he gained Dipl. Informatiker degree from the University of Hamburg, and he has since worked as a research assistant for the University of Hamburg's Computer Science department and as a senior software developer and key account manager at the ifu Hamburg GmbH (until 2005). In 2005 the University of Hamburg awarded him a PhD in Applied Computer Science and he was appointed to a professorship for industrial environmental informatics at the FHTW Berlin in October 2005. Professor Wohlgemuth's research focuses on modelling and simulating environmental effects of industrial and business systems.

Bernd Page holds a PhD in Applied Informatics from the Technical University (TU) of Berlin (Germany) and an MS from Stanford University (USA). Before he was appointed to a professorship in Applied Informatics at the University of Hamburg, Professor Page worked as a scientific associate in the Environmental Information System Group of the German Federal Environmental Agency (in Berlin). Since joining the faculty of the University of Hamburg he has been engaged in research and teaching of computer simulation and environmental informatics. During the 1980s he co-founded and served as chairman of the Special Interest Group "Informatics for Environmental Protection" of the German Informatics society (GI). Professor Page has initiated and supervised many funded applied research and industry projects. He has published widely.

Wolfgang Kreutzer received his PhD in Applied Computer Science from the University of Frankfurt (Germany) in 1978, and has since worked for a number of universities in New Zealand, the USA, Canada and Sweden. He is now an associate professor at the University of Canterbury (New Zealand). Associate Professor Kreutzer has taught simulation and conducted research in related areas for many years. His current interests center on the use of visual representation and animation in simulation models, programming languages and multi-media systems.

Illustrations:

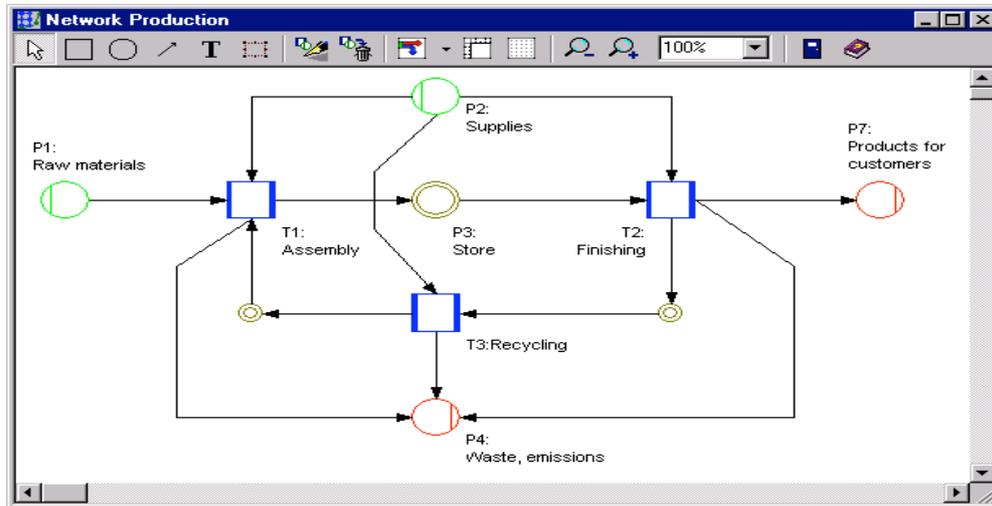


Figure 1: A simple Material Flow Network

Formerly known flow: Housing 7,000.00 kg 

Input Coefficients		Output Coefficients	
Steel sheet	7.00 kg	Housing	6.20 kg
Plastic parts	0.50 kg	Metal waste	1.20 kg
Electr. Energy	4,000.00 KJ	Plastic waste	0.10 kg

 Steel sheet 7,112.03 kg
Plastic parts 564.52 kg
El. Energy 4,516.14 GJ

 Metal waste 1,354.84 kg
Plastic waste 112.90 kg

Process level:
 $7,000.00 / 6.2 = 1,129.00$

Figure 2: Simple transition specification which uses production coefficients, and its application to building; based on Schmidt, et al. (1997), page 775.

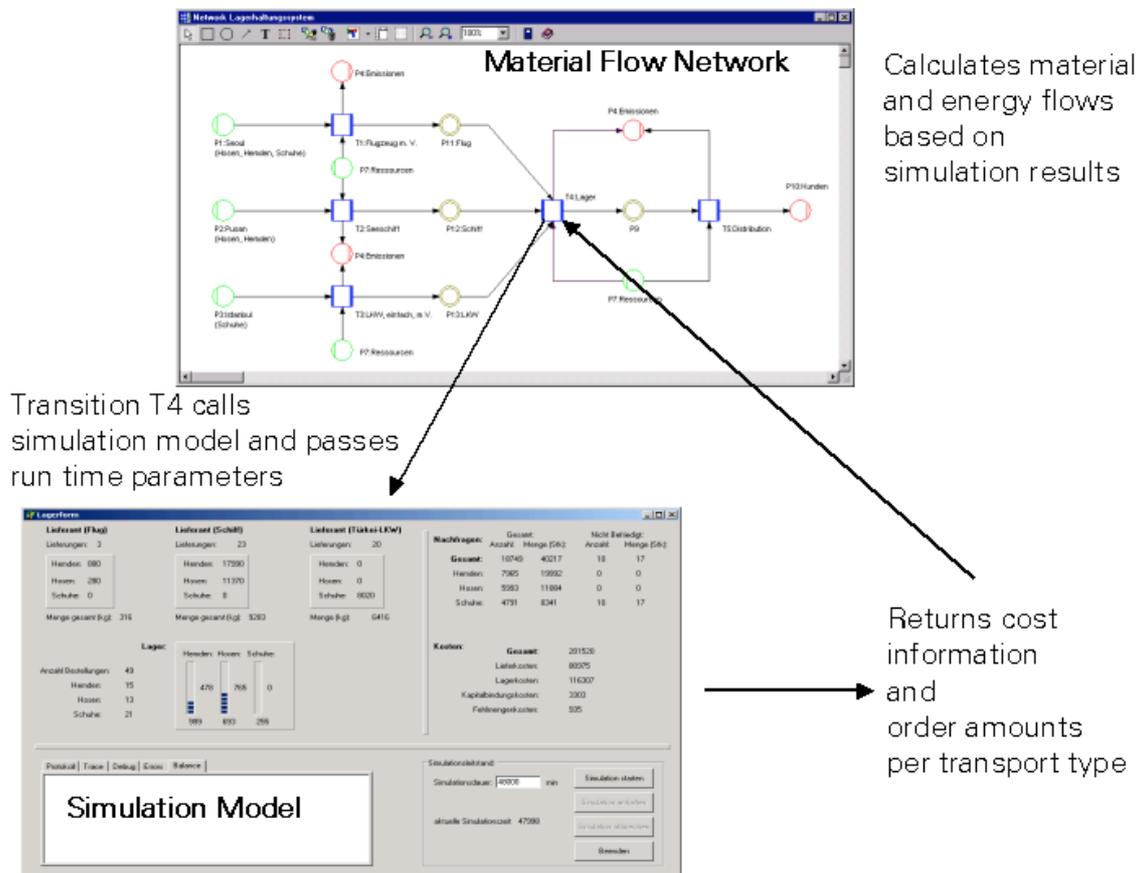


Figure 3: Linking a Discrete Event Inventory Simulation to a Material Network; see Wohlgenuth, et al. (2001).

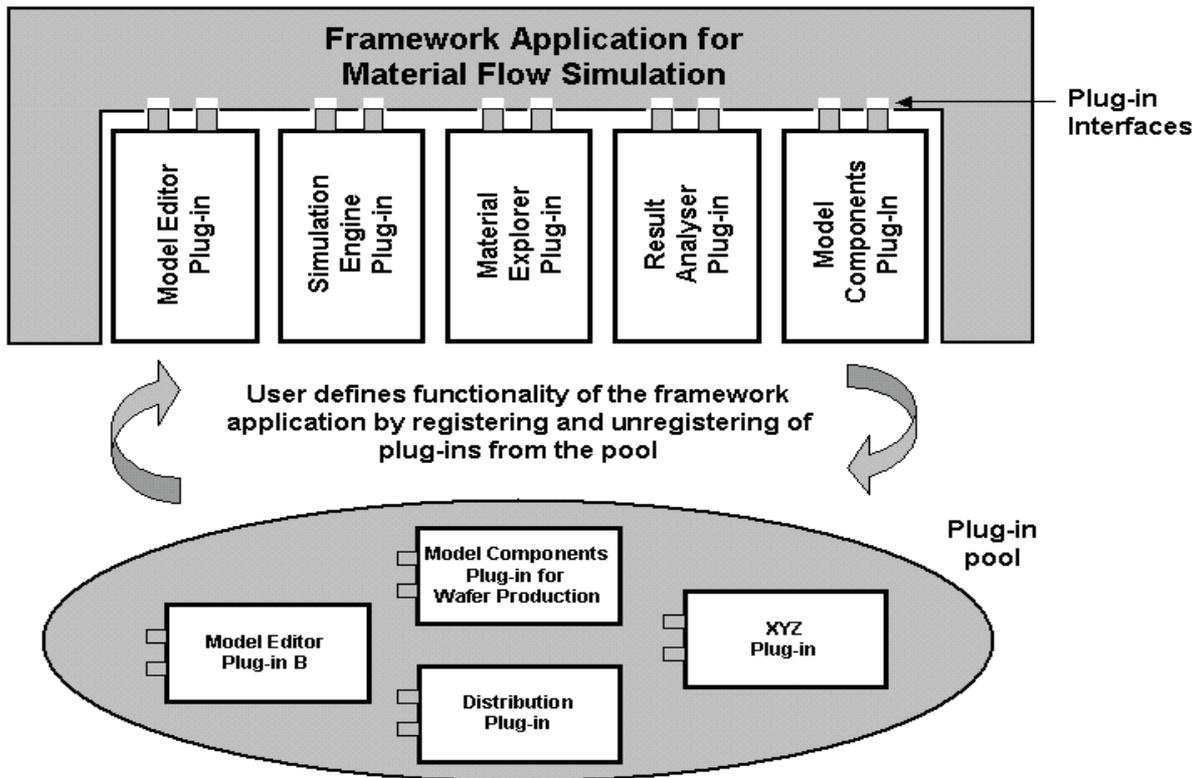


Figure 4: Building an Application Framework for Material Flow Simulations from Component-Based Plug-ins

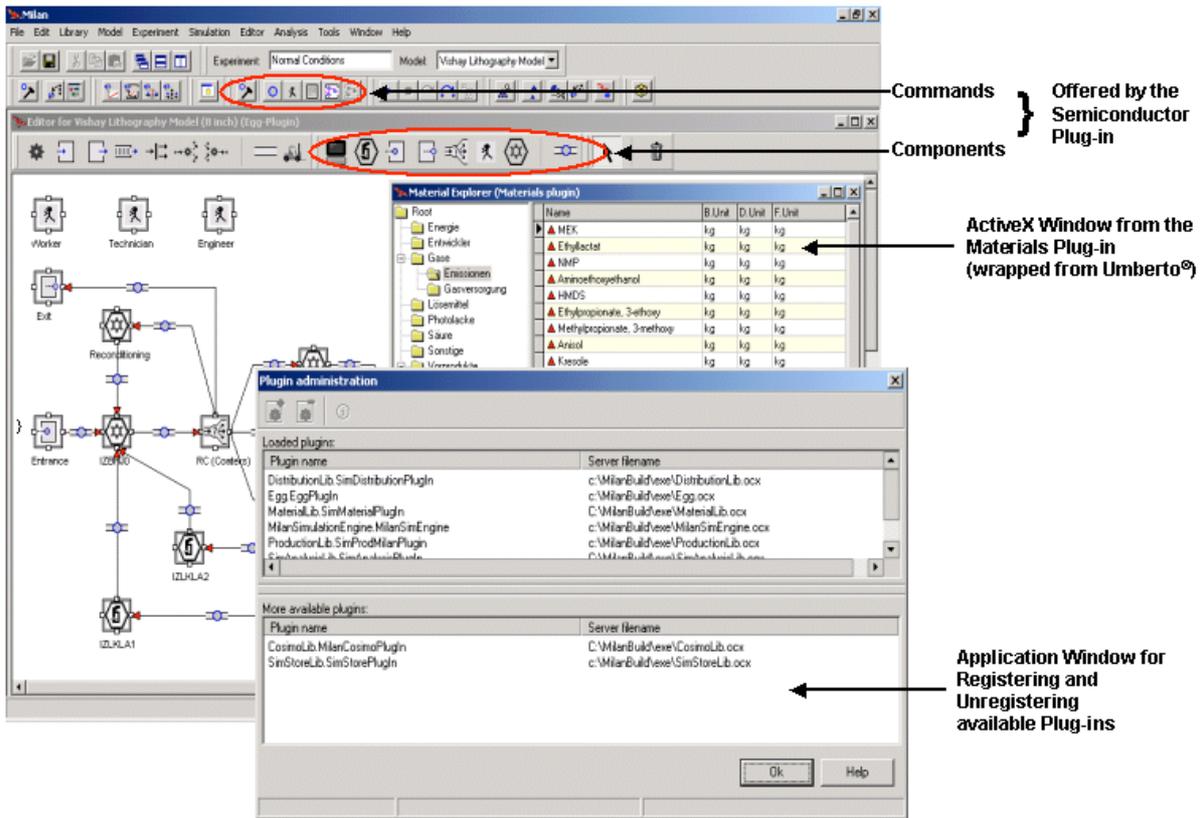


Figure 5: Commands and Components for Plug-ins - as seen by the Milan Framework Application

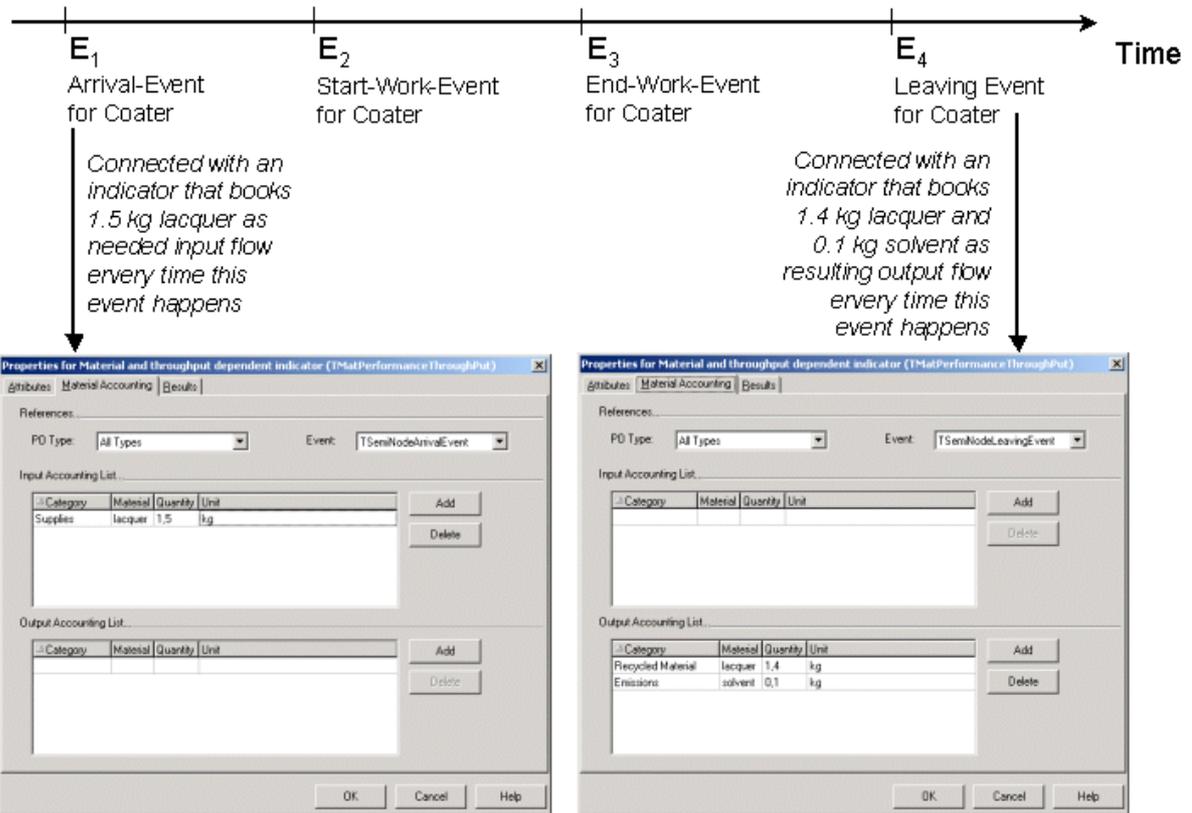


Figure 6: Relationships between Material Indicators and Events for a Coater

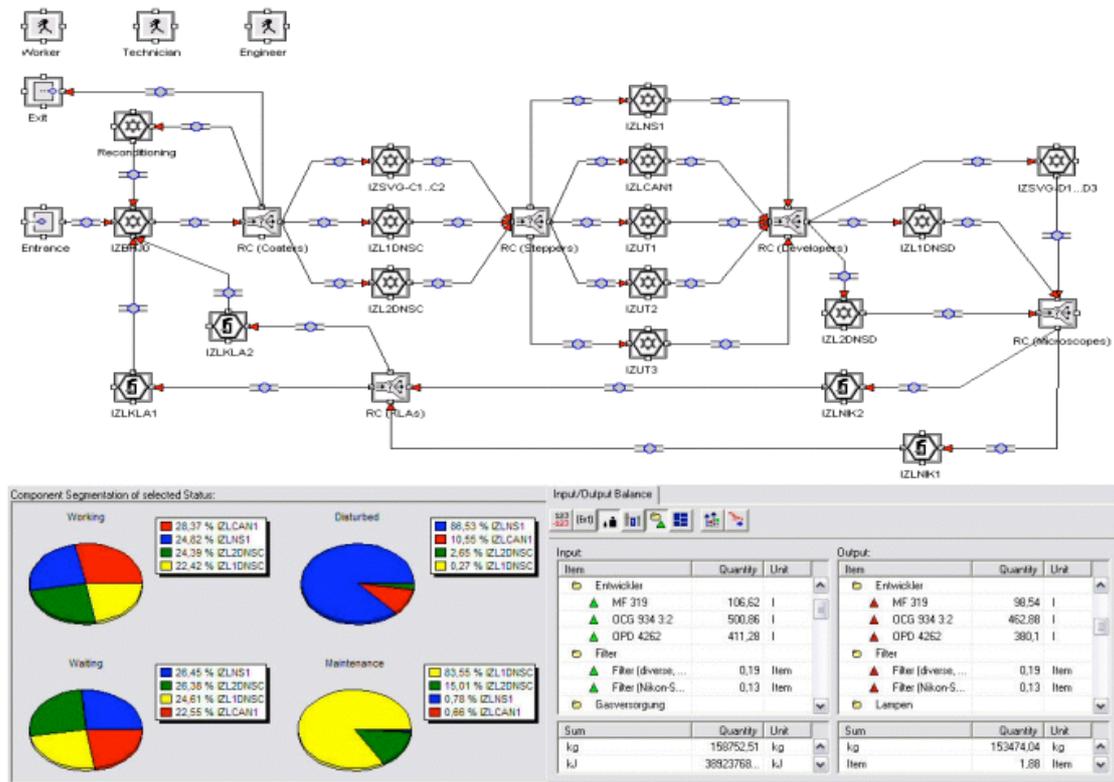


Figure 7: Results of a Lithography Simulation and its Material Flow Analysis