Simulating the Impact of Information Flows in Networked Organizations

Peter Peters
RWTH Aachen

Matthias Jarke
RWTH Aachen

Follow this and additional works at: http://aisel.aisnet.org/icis1996

Recommended Citation
http://aisel.aisnet.org/icis1996/30

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 1996 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
SIMULATING THE IMPACT OF INFORMATION FLOWS IN NETWORKED ORGANIZATIONS

Peter Peters
Matthias Jarke
Lehrstuhl für Informatik V
RWTH Aachen

Abstract

Information and communication technology are crucial enablers of networked organizations, but effective Cooperative Information Systems remain difficult to design because there are no tools to assess their short-term operational effectiveness and long-term impact on organizational knowledge. We present three contributions toward a solution of these problems: 1) we identify information flow categories as a crucial conceptual abstraction; (2) we show how successful simulation models for long-term analysis of team performance can be extended to include the crucial problems of cross-unit information management and exception handling; (3) we exploit recent results on multi-simulation to enable an integration of these long-term models with queuing-type simulation models usually applied in operational business process analysis. The approach has been implemented in an interactive analysis environment called MultiSim, and has been tested in a case study in the production industry.

1. INTRODUCTION

Cooperative information systems (CIS, De Michelis et al. 1996) is a term coined for a new generation of modular distributed information systems intended to support the vision of networked organizations in which semi-autonomous units continuously adapt their patterns of interaction to changing customer needs and organizational goals (Brynjolfsson and Mendelson 1993; Scott-Morton 1994).

The frequency and distribution of change activities does not imply that careful information systems planning and impact analysis are now obsolete. On the contrary, they become much more frequent and must therefore be better supported than before. The danger of simplistic analysis is amply demonstrated by the observation that about 50% of all business process reengineering projects did not lead to the perceived improvements (Hall, Rosenthal and Wade 1993). One important reason for the failure is, according to Hess, Brecht and Österle (1995), that the process abstractions chosen covered only short-term operational effects and ignored the importance of processes that support learning and the collection of organizational experience. Not surprisingly, impressive short-term cost savings were achieved but disaster followed in the medium term because the company lost its strategic advantage of superior knowledge in some domain.

In this paper, we therefore advocate a broader perspective toward organizational impact analysis for CIS which addresses both the long-term and the short-term effects of changing information flows in networked organizations. Three ideas are put forth to achieve this ambitious goal:

1. We start from information flows between organizational units as our basic conceptual primitive which is then linked into traditional IS frameworks by means of differentiating between operational task-oriented information flows, exchange of corporate memory, and promotion of strategic ways of thinking.

2. We build on industrially validated simulation models for the case of software project analysis, and extend them on the one hand by specific modules for analyzing information management, exception handling, and cross-unit information flows, on the other by consideration of short-term operational issues such as workload.
3. We combine recent research in multi-model simulation and repository technologies to enable practical support for the integrated simulation of both long-term and short-term effects.

In section 2, we present a brief state-of-the-art in impact analysis for organizational information systems. Section 3 describes the above approach in some detail while section 4 presents a case study in industry which serves as a demonstration and first validation of our approach. Section 5 offers some conclusions.

2. INFORMATION IMPACT ANALYSIS: A BRIEF REVIEW

The analysis of IS impact on business performance and the resulting IS design decisions are complex tasks. The IS impact on the performance of networked organizations depends on its adaptation to a variety of tasks such as transaction processing, decision making, and communication, or organizational values such as organizational philosophy and managerial flexibility (Wilson 1988; Ein-Dor and Jones 1985; Raymond, Paré and Bergeron 1993). Methods appropriate for specific problems in the context of IS impact analysis can be organized into descriptive, analytical, and simulation methods (Ein-Dor and Jones 1985; Banker, Kauffman and Mahmood 1993).

No matter which analysis tools are used, IS impact analysis always starts from empirical methods (surveys, Delphi studies) using perceptual (growth, profitability) or objective measures (ROI, ROA) (Brynjolfsson and Hitt 1993; Bergeron and Raymond 1995; Chun and Huff 1993). In networked organizations, empirical research is appropriate to capture IS impact perception of the different stakeholders but can hardly give a global picture. As long as general mechanisms behind the interaction of business and IS are poorly understood, expert knowledge provides the only general way to assess the IS impact on organizational performance. Vice versa, case studies or portfolios provide valuable insights how specific results were achieved by a company (e.g., Goldstein 1993; Lucas and Olson 1994), but their transfer to other business environments depends on the skills of IS managers.

Based on the results from empirical research, analytical methods such as statistical decision theory (Raiffa 1968), team theory (Marschak 1955), Cost-Benefit-Analysis (CBA) or econometrics are of varying importance for IS impact analysis. The first two analyze the value of information in decision situations and do not capture the impact of the IS as such. Furthermore, they are helpful mainly in laboratory situations, since handling of the created equations becomes far too complex in real world situations (Ein-Dor and Jones 1985).

Classical investment analysis methods such as CBA assess functional productivity by monitoring and assessing certain key financial indicators that drive IS investment. This “management by numbers” (Dean 1987) is not concerned with the dynamics of the firm in a competitive market situation where flexibility and process-orientation are important aspects. Only recently, CBA is being extended by inclusion of strategic visions and business intangibles (Toraskar and Joglekar 1993).

Econometric productivity assessment became popular with the COCOMO model of Boehm (1981). Brynjolfsson and Hitt analyze the relation between firm level output and IS spending and Laskowitz (1994) relates a firm’s output to collaborative decision making structures and technology. The explanatory power of econometrics is relatively low, since the user must have deep knowledge of theory behind the equations to analyze the condensed information provided by them. Furthermore, the dynamics of the interactions between IS and business are not captured.

The environment of networked organizations is dominated by continuous change. Analytical methods are of limited help since they only deal with static and mainly monolithic environments. Simulation, the classical method for computer-based analysis of dynamics in complex systems, has been used in the context of IS analysis and organizational performance only to a limited extent: for the behavioral analysis of IS design with respect to correctness and completeness and performance analysis of teams.

The first group of simulation approaches deals with the analysis of IS designs in business situations, e.g., Deiters, Gruhn and Striemer (1995) and Plexousakis (1995). They use discrete event simulation to describe and analyze behavioral aspects of IS specifications. The major goal is to match the IS design with functional customer requirements. If the simulation couples business process and IS design, the assessment can include the alignment to functional aspects of business processes (Oberweis, Scherrer and Stucky 1994).
A second group of simulation methods focuses on project performance and organizational learning. They simulate complex feedback behavior that determines the dynamics of organizational performance using System Dynamics (Forrester 1961; Senge 1990). Abdel-Hamid and Madnick (1991) showed how factors such as schedule pressure or adding new workforce influence the dynamics of software development projects and provide performance factors that are influenced by the available information.

While the first group of simulation approaches focuses on short-term behavior (“is the actual process supported in an appropriate way?”), the second group deals with the mid-term and long-term performance of a group (“what factors influence group performance or learning processes?”). We found no examples where the long term influence of IS on organizational performance is analyzed by simulation. Such an analysis would only be significant if statements in terms of throughput or capacity measures could be made in order to combine the long-term impact of information with the short-term business variables to be controlled. In the following, we describe an approach to deal with the combined dynamics of short-term and long-term impacts of IS in networked organizations.

3. A SIMULATION MODEL FOR INFORMATION FLOW IMPACT ANALYSIS

In order to study the productivity impact of IS jointly with the long-term effects of available information on organizational learning and change processes, we have developed our simulation method around three key ideas:

1. A formal conceptual model of information flows within a networked organization that includes a categorization of information flows according to their impact on organizational performance (Peters 1996).

2. A simulation model of group performance in software projects by Abdel-Hamid and Madnick that captures variables which influence the productivity of a group in a given working situation.

3. The multimodeling approach by Fishwick (1995) that allows the combination of several simulation techniques in order to jointly simulate continuous (learning processes) as well as discrete factors (schedule pressure, workload, business variables).

Starting from a description of the information flow categories identified, we illustrate how the existence of information flows influences interaction in a networked environment. To simulate this influence, we developed a model of department performance. The specific realization of IS impact analysis in this model is presented in section 3.3. Finally, we present the multi-simulation formalism and tool that provides the technical basis for the design and execution of heterogeneous simulation models.

3.1 Categorization of Information Flows

In the tradition of the CRIS methodology (Olle et al. 1991) business process descriptions for IS design consist of tasks and objects consumed and produced by those tasks. In a cooperative business modeling project with industrial engineers throughout Germany (Jarke, Jeusfeld and Szczurko 1993; Peters and Jeusfeld 1994), we observed that this model neither considers the distribution of organizational units (agents) nor different methods of performing tasks, and extended it accordingly. The crucial observation for this paper is that this extension induces a differentiation of the exchanged information objects in operational task information flows determined by methods used to solve the tasks and in exchanges of more general knowledge among agents (cf. Figure 1). Additionally, both kinds of information flows are further influenced by high-level strategies. We found ample evidence in the literature that the following distinction of information flows makes sense (Peters 1996):

1. Task Information drives and controls the business processes in organizations (Hammer 1992; Olle et al. 1991; Scheer 1993). For example, the structure of a product in manufacturing industries. This information is commonly understood within an organization. Its transfer between departments or tasks is based on standardized formats. The effects of task information are short term and local. It results from one department, is transferred to the next step, and can initiate a new task. If task information is missing or late, the process is stopped or, in the extreme, canceled. After task execution, it is stored in the IS and might become part of the corporate memory.
2. **Corporate Memory** about products and processes results from accumulated execution and analysis of business processes (Senge 1990; Vennix et al. 1994; Harmsen, Brinkkemper and Oei 1994). Capturing this corporate memory can ensure higher quality and efficiency of a process execution (Chen, Liou and Weber 1992; Fine 1986; Pentland 1994). Shared knowledge enables higher productivity and a faster learning curve of the company (Adler and Clark 1991; Cooprider and Victor 1993). In contrast to task information, corporate memory cannot be formalized easily. It consists of informal documents, such as experience collections or stories, and of formal ones, such as design rules or process models. It needs rich communication support that reduces equivocality and uncertainty in communication (Daft and Lengel 1986). In manufacturing industries, the production and design know-how is becoming obsolete about every three years. Therefore, the effects of corporate memory are of mid-term range. The effects can be global, meaning that collected corporate memory can influence processes in various other parts of the organization.

3. **Strategies** define a common context according to which tasks are organized and information is interpreted (Hammer 1992). They consist of a set of visions, policies and goals under which an organization or department operates. Examples are TQM, JIT, or BPR. Strategies result from long-term experiences (Burgelman 1988) or theories. They are implemented to ensure long-term competitiveness and should influence every part of the organization.

These categories form the groundwork for information flow impact analysis in this paper.

### 3.2 The Dynamics of Information Impact

According to the differing contents and ways of processing, the performance impact of exchanging task information and corporate memory among organizational units must be analyzed using different criteria. The impact of task information is determined by **short term effects** such as transaction costs, timeliness, and completeness. The exchange of corporate memory influences performance by **long term effects** on process quality, effectiveness, or personnel qualification.

**Task information** impact criteria are typically defined by discrete, quantitative measures, describing short-term, local effects, for instance the time needed to collect or produce information for a given, identifiable task. Since the exchange of task information is always connected to the flow of tasks along business processes, these criteria are related directly to the business process (and therefore monetary or time-related business criteria). The dynamics of such systems are usually analyzed by Petri-Net or Queuing System simulation (e.g., the examples of the previous section).

The analysis of **corporate memory** is more problematic, because its effects are related to long-term feedback loops within an organization: information is accumulated, condensed and transferred to the organizational units where it is needed. The impact of those strategic learning processes is hard to measure in task-related time and money. These feedback loops work by the way they influence business variables that produce time-and-money effects. These effects cannot be captured by the exchange of discrete units of information. Therefore, corporate memory is viewed as a continuous effector and the rate of this effect is determined by the availability of corporate memory and its understanding by the communicating partners.

This is why we selected SD for the long-term IS impact analysis based on the corporate memory concept. The SD metaphor for system description is a set of **levels** which are connected by pipes. The flow of non-discrete **resources** through the pipes is regulated by valves. The **flow rate** at these valves depends on information about the state of the overall system. The calculation of the rate is decomposed into smaller steps by a network of **auxiliaries** which perform simple calculations on incoming system values (levels, rates, **constants**, or already derived values). Therefore, SD cannot be used for keeping track of specific resource items such as tasks, but is suitable for the description of complex relations which can be approximated by functions over non-discrete resources. The description of impact behavior over time is typically described in such a fashion (e.g., learning curves, predator-prey relationships, search behavior).
In their work on software project dynamics, Abdel-Hamid and Madnick developed and validated a set of System Dynamics (SD) models for the major factors that influence a team’s software development productivity and validated them in numerous industrial applications. The step from a single-group, single-project model of Abdel-Hamid and Madnick to a multi-group, multi-task situation required an extension by models that deal with error management, information management and discrete exchange of tasks among the multiple groups in the networked organization. The resulting department performance model (see below) provides the basis to interrelate models of cooperating departments by information flows and task exchange.

The overall simulation model combines SD and Queuing Systems in order to represent discrete and continuous flows between department performance models. They are coupled by three types of links (cf. Figure 1):

Figure 1. The Mapping Between Conceptual and Simulation Model

1. The discrete flow of tasks along the business processes is represented by a Queuing System. The service time of a queue is not generated using a distribution, but is based on the planned service time of the task currently performed and the state of the SD performance model of the department.

2. The effects of exchanging corporate memory between cooperating departments are represented by specific SD auxiliaries between the performance models (information management submodel).

3. The flow of errors and rework tasks between departments is represented by SD auxiliaries between error management submodels.

Since the flow of task information is closely related to the flow of tasks and is considered priceless in the context of transaction systems (Ein-Dor and Jones 1985), it is not depicted as an extra flow. The costs of task information are defined by the information access and production costs per task. The effort to produce and distribute task information or access it from the predecessors in the business process is encoded in the information management model.

The combination of queuing systems for the representation of task flows and SD models for the representation of organizational feedback loops allows for the combination of situation dependent effects such as throughput or meeting of schedules (not observable in SD due to non-discrete resources) with long-term processes such as organizational learning (which cannot be modeled with acceptable effort by Queuing Systems or Petri-Nets). This enables a direct coupling of operational aspects of IS performance with strategic aspects such as supporting organizational strategies.
Peters, Jarke and Mandelbaum (1996) describe in detail how the conceptual model of information flows and the approach of Abdel-Hamid can be combined to form a model of department productivity in a federated environment. The resulting department performance model consists of the following parts (cf. Table 1):

**Table 1. The Role of the Models in the Simulation Approach (cf. Figure 1)**

<table>
<thead>
<tr>
<th>System</th>
<th>Purpose of Analysis</th>
<th>Resource</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning and Controlling</td>
<td>- definition and control of schedule</td>
<td>- man days</td>
<td>- workload</td>
<td>- schedule - workforce needed</td>
</tr>
<tr>
<td>Queue</td>
<td>- management of task flow</td>
<td>- tasks</td>
<td>- new tasks - schedule</td>
<td>- workload - performed tasks</td>
</tr>
<tr>
<td>Manpower Allocation</td>
<td>- allocation of manpower to performance models</td>
<td>- man days</td>
<td>- workforce</td>
<td>- fractions of available manpower</td>
</tr>
<tr>
<td>Task Productivity</td>
<td>- task productivity - effects of schedule pressure</td>
<td>- tasks</td>
<td>- avail. manpower - schedule - corporate memory</td>
<td>- performed tasks</td>
</tr>
<tr>
<td>Error Management</td>
<td>- error and rework rate related to workforce, schedule pressure, avail. information</td>
<td>- errors</td>
<td>- tasks performed - corporate memory - schedule pressure</td>
<td>- errors not found - rework effort needed</td>
</tr>
<tr>
<td>Information Management</td>
<td>- manpower needed for task information production - manpower needed for information management - effects of reuse of information</td>
<td>- corporate memory - task information</td>
<td>- tasks performed - manpower available</td>
<td>- information avail. - manpower needed</td>
</tr>
</tbody>
</table>

1. The *Human Resource Management* (HRM) model. The agent role in the conceptual model is filled by the department workforce. The available workforce calculated in the HRM defines the manpower that can be used in the Method Performance model.

2. The *Planning and Controlling/Queue* models. The Queue describes the flow of discrete tasks in and out of the department and provides the actual workload for planning. The Planning and Controlling model compares workload and schedule with the actual performance parameters (*work completed*) and defines the current schedule pressure, the need for adding new workforce, and the service time needed for the job currently serviced in the Queue.

3. The *Method Performance* model. It describes how manpower is spent on various jobs. These jobs are described in four submodels concerning task productivity, manpower allocation, error management, and information management.
   a) The *Manpower Allocation* model distributes the available workforce over the jobs in the following three models according to company policies.
b) The Task Productivity model evaluates the amount of a task performed per day according to the structure of the workforce and the “history” of productivity, i.e., the schedule pressure history or workforce exhaustion.

c) The Error Management model provides the means to analyze by which rate errors are generated, detected and reworked in a department and determines the effort needed to perform those tasks.

d) The Information Management model is based on the task information and corporate memory categories. It is described in detail in the next subsection.

3.3 The Information Management Submodel

The Information Management (IM) model describes the effort necessary to access, provide and manage information flows. It focuses on the role of reuse in information production, the generation of corporate memory from task information and the effects of available information on department performance. The detailed structure of the simulation model is depicted in Figure 2.

Figure 2. The Information Management Model
According to the information categories, two major resource flows were defined: the task information resource flow and the corporate memory resource flow.

Task information is represented as consisting of task documents generated and distributed in close connection with the performed business processes. The time necessary to produce task information cannot be predefined but is closely related to the job productivity: information productivity increases proportional to the job productivity. This situation is defined in the lower left of Figure 2.

The reuse of existing task information plays an important role for higher quality and higher productivity of information generation in manufacturing (Pfeifer 1993). The system of auxiliaries in the upper left of Figure 2 describes this influence:

1. In the reuse step the engineer searches for helpful information in the task information base. This search is described by the triangle $\text{INRETI, FTREAC, INUSAB}$: $\text{INRETI}$ is the time necessary to find information, $\text{FTREAC}$ defines the fraction of daily manpower for information production available for search, and $\text{INUSAB}$ describes the success of this search which depends on time and information availability. Additionally, success and failure of accessing task information was embedded into the system by increasing $\text{FTREAC}$ every time the access was successful and decreasing $\text{FTREAC}$ every time it was not. This feature describes system acceptance: if a newly introduced information base does not provide the information the user needs at a demanded speed, the tool and the information base will not be accepted, even if the situation gets better over time.

2. Based on the reusability of information, the nominal times for the other two information production steps, accessing information along the task information flows ($\text{INACTI}$) and information design ($\text{INDETI}$), can be adjusted.

3. The values for $\text{INRETI, INACTI}$, and $\text{INDETI}$ are added up in the $\text{INGETI}$ variable. This variable is now used to adjust task information productivity by comparing the time available ($\text{DMPINFO}$) to the time needed ($\text{INGETI}$).

The corporate memory resource flow is based on the resource corporate memory units, which is defined as a fraction of the produced task information. Managing corporate memory deals with all work that is not related directly to producing task information, such as maintaining the information base or generating corporate memory from task information. Typically, such work is not embedded into day to day work, but is performed in defined slots, e.g., once a week or once a month. Under the policy assumed here, the time available for those slots is reduced or enlarged not in relation to productivity but in relation to the actual schedule pressure.

Information management productivity now depends on the relation between the time normally needed ($\text{NFMPIM}$) and the time available ($\text{AFMPIM}$). The effect of the IMPRD value is twofold:

1. It defines the intensity of IM tasks, i.e., the rate of corporate memory production and information base management ($\text{TIDERT, CMDVRT, CMDERT}$).

2. It defines the quality of the corporate memory influence ($\text{CMINF, CMTRANS}$). Only if the corporate memory is updated and managed properly over time does the information fit the user needs. The $\text{ACIMPRD}$ describes the average productivity which provides a measurement for the overall quality of the corporate memory.

The influence of corporate memory on other parts of the model is defined by two auxiliaries: $\text{CMINF}$ describes the influence within the department (on productivity, information generation, error generation, and hiree assimilation). $\text{CMTRANS}$ describes corporate memory flows to other departments as explained in section 3.2. The effects of those flows are defined by a variable that mixes the quantity ($\text{CORMEM}$) with the quality ($\text{ACIMPRD}$) of an information flow.
3.4 The MultiSim Environment

In order to support a close contact between customer and IS impact simulation and to establish the model as a learning environment, the simulation modeling and execution environment should allow for a user-friendly and fast iteration of the modeling-simulation-analysis cycle. Additionally, this environment must allow the modeling and execution of heterogeneous simulation models such as the one presented in the previous sections.

To reach this goal we combined meta modeling and repository techniques for model integration (ISO/IEC 1990) with the multimodeling approach by Fishwick that enables interaction between simulation techniques implementable as discrete event techniques. The resulting simulation environment MultiSim facilitates the graphical modeling and analysis with multiple simulation techniques such as SD and Queuing Systems (cf. Figure 3):

1. It uses the structure of simulation technique models stored in a repository of conceptual models to adapt the editor to the simulation technique selected by the user, (cf. Appendix A for a repository description).

2. It provides a graphical design environment for the definition of simulation models which are checked by modeling constraints defined at the simulation technique layer and are stored at the simulation model layer.
3. It provides a simulation run execution environment that retrieves models from the repository, allows initialization according to the experiment, performs the simulation, and stores the results in either a relational database or a file system.

4. It contains facilities for the display and comparison of simulation experiment results.

Besides the interest of the participating engineers in the results of the simulation experiments, the fun that the MultiSim environment provided by playing with the simulation model led to the success of the simulation as a learning environment in the case study presented in the next section.

4. ANALYZING INFORMATION FLOWS: AN INDUSTRIAL CASE STUDY

An SD/Queuing model consists of about 100 variables. Such a model is typically underspecified, which means that “driving the wrong screws” could lead to acceptable overall results with respect to the variables of interest. Careful calibration and validation is necessary before the models can be used for analyzing organizational information flows. About 70% of the variables (HRM, Task Productivity, Manpower Allocation, and Planning) were validated by Abdel-Hamid in numerous case studies (e.g. Abdel-Hamid and Madnick 1991; Abdel-Hamid, Sengupta and Ronan 1993). A first validation of the new parts was performed during an industrial case study.

The case study was performed in a medium-sized manufacturing company, which produces system solutions for packaging machines. Since those machines consist of mechanical parts and control software, there are two main business processes performed by six semi-autonomous departments at three different sites (cf. Figure 4). One consists of the mechanical design and the manufacturing of machines which then are assembled and tested together with the control software that is developed in a parallel process. In the case study we focussed on the mechanical engineering process highlighted in gray.

The company was applying for ISO 9000 certification and therefore needed to analyze and reorganize their business processes. We supported their information management organization and were able to do a first calibration of our simulation model at least for the context of the specific company (cf. Appendix B). Afterwards, we used the validated models as an “information management flight simulator” by which changes to the system infrastructure and information flows can be simulated and analyzed. An example is given below.

Figure 4. Proposed Changes of the Company’s Information Flows
Simulating the Impact of Information Flows in Networked Organizations

4.1 The Change Situation

Meeting their schedules is critical for our case study company since late arrival of packaging machines leads to the paralysis of whole production lines for their customers, the food industry. Schedule problems in the design department were considered the reason for the overall schedule meeting problems the company experienced at that time. The introduction of a new CAD system was considered a starting point for a possible solution (cf. Figure 4).

The system was only the cornerstone for the reorganization of information exchange between the departments: a faster distribution of drawings and support for their collection, storage and organization was planned. The drawings were often changed during manufacturing and assembly due to design errors without feedback to the design department. Therefore, the collection of corporate memory about changes at the places where they occur together with a better access to drawings in a CAD database should lead to faster design and a reduction of design errors by reuse of previously corrected designs. All in all, the CAD system was a new node in the information flow network around which the overall information exchange had to be reorganized. For this reorganization several scenarios were considered possible.

We conducted twelve interviews with engineers from different departments about the local effects that these changes might produce with respect to information management and error production. With conservative predictions we performed a set of twelve experiments (four for every possible scenario) where we analyzed the local and the global behavior of the changed environment. Figure 4 describes the scenario presented in the remainder of this section.

Figure 5a) shows the simulation results for the design department against the base case simulation (from model validation) with respect to the business variables throughput and schedule meeting. The results indicate that the implemented changes show no positive effects in the first half year. Afterwards the throughput results are significantly better than in the base case situation and finally, for the last two tasks of the sample, the deadlines are met.

A first analysis shows an overlay of two effects: Better support of task information flow leads to better performance. The additional effort needed to manage the collected design drawings consumes these positive effects at first. But after some time of collecting corporate memory, the reuse of drawings and design error knowledge leads to an acceleration and quality improvement of the design process.

A closer look at the simulation results indicates that this description of the dynamics is still too simple. There are other internal effects that influence the design throughput. Additionally, there are positive effects that influence the performance from outside the design department along the feedback loops. Our simulation approach allows for a deeper analysis of these effects.

4.2 Local Effects

A major goal of the planned changes was the reduction of design errors. Figure 5c shows the change of the error generation rate based on the proposed changes. Two major effects were observed:

1. The area under the curve of the experiment is smaller, because the design tasks are performed faster and the curve falls back to its default value earlier. The peaks of the error generation rate are smaller, too. This effect grows over time.

2. These observations sum up to a significant decrease in the overall number of errors. A thorough analysis of the simulation runs identified two major effects that were responsible for this decrease with respect to local causes.
First, the intended effects *reuse of drawings and usage of error information* have an impact on the curve. As Figure 5d shows, the available information, task information as well as corporate memory, is growing steadily. With a growing amount of information, the influence of corporate memory will grow. This leads to the movement of the experiment curve away from the base case, but does not explain why the effects are that strong after a short period of time.

The secondary effects are more complicated. The following *positive feedback loop* was identified: The decreased error generation rate leads to a decreasing amount of rework; this in turn reduces the manpower allocated for rework tasks. The less manpower is needed for rework, the more is available for task performance. This increases the speed of task development and hence reduces the schedule pressure. The decreasing schedule pressure now has two effects: First, it reduces the error generation rate directly and, second, it increases the manpower available for information management (cf. Figure 5d), which in turn increases the amount and quality of available information. This is a first example of how complex dynamics sum up to a locally observable effect.

Figure 5. Prognosis on IS Changes
Before we illustrate the global effects, we take a look on the information dynamics of the experiment which are depicted in Figure 5d. It shows the development of the levels of task information and corporate memory as well as the main function that couples them, the daily manpower for information management. Four observations are of interest:

1. The amount of task information grows with the number of tasks produced.

2. The information management manpower decreases whenever schedule pressure is growing. It is at zero at every deadline that was not met in the experiment. Additionally, the increasing peaks of the available man days indicate that the schedule pressure is decreasing over time.

3. The amount of corporate memory grows in relation to the task information and the DMPIM. The curve also indicates that every time DMPIM is available again, the process catches up with the amount of task information produced during the DMPIM depletion.

4. The amount of corporate memory is increasing even after all jobs are finished. The reason for the additional amount is the flow of corporate memory from other departments.

4.3 Global Effects

Figure 5b depicts the manpower needed for rework in all three departments. Both curves show that the amount of time spent on rework tasks increases along the production processes. This observation is in line with classical problems of quality management: errors are detected and reworked in the late phases of production were the costs are higher than in the early phases.

The first effect of the change described in the experiment is one across departments in the direction of product flow. Error reduction in the design phase leads to a decreasing error escape rate which in turn results in less manpower spent on rework in the late steps of the production process. Even if the manpower needed for managing the corporate memory about errors adds to the overall manpower needed in design, the strong positive effects on the performance of later departments along the process would justify the change implementation.

The second observation is a feedback effect that influences the productivity of the design department. A closer look at the curve for rework in design shows that a lot of tasks are sent back to design. Normally, the base case line would go down to the default value after day 240, the end of the planned design jobs (as indicated by the forking of the base case curve). But rework tasks are still delegated back from the assembly and manufacturing department.

Even if the corporate memory could not show its full effect after nine projects in the experiment, rework in the design department is reduced significantly. Avoiding past errors in design led to less rework in the assembly department and, due to sinking rework delegation, in the design department, too. This in turn leads to higher productivity, lower schedule pressure and less errors per task. A large scale positive feedback loop is implemented that leads to higher overall process quality.

5. CONCLUSION

In this paper, we have presented a simulation approach that deals with the long-term impact of IS on organizational performance. The presented theory on information flows emphasizes the combination of short-term and long-term effects of information on organizational performance. The simulation model reflects this statement by combining queuing and SD
Peters and Jarke

simulation techniques to include effects on short-term variables such as throughput and workload and long term organizational feedback behavior as it is induced by knowledge transfer. The tool concept necessary to build and analyze these multimodels combines results from simulation research with developments in model integration and repository technology in the presented MultiSim tool.

The case study illustrates how modeling of feedback loops by information flows can be used to analyze the effects of IS changes in an industrial setting. It highlights that changes of IS should not be restricted to new systems (CAD), but should be planned corresponding to the effects this system can have on the exchange of information along business processes and feedback loops. The simulation method showed its value by giving the management insights into the complex dynamics behind the IS impact on productivity.

Obviously, further work needs to be done to validate the structure and contents of the simulation model in order to generalize the approach beyond the presented company. To serve as an integral part of an information flow design and analysis environment for cooperative information systems in networked organizations (Jarke et al. 1996), the changeable parameters must be reduced to a number manageable by the ordinary user.

6. REFERENCES


Simulating the Impact of Information Flows in Networked Organizations


**APPENDIX A: STRUCTURE OF THE MULTISIM REPOSITORY**

A language for the combination of simulation techniques has to provide a taxonomy of modeling elements independently of a specific simulation technique. Our language at the topmost repository level is derived from the analysis of simulation techniques of Fishwick (1995) and other taxonomic approaches such as SES (Zeigler 1984). According to those approaches three major elements of a discrete event simulation exist:

- entities describe the structure of the system to be analyzed in the simulation model,
- activities describe the internal or interacting behavior of the entities over time by events, and
- conditions which constrain the behavior of the overall system.

These elements form the backbone of the Discrete Event Simulation Language (cf. Figure 6). The system structure is characterized by the Entities, their behavior, and their connections. Every Entity can have a set of state describing attributes and always has an initialization. We have identified two Entity classes, the Stations that manipulate or store items and the Load manipulated by the Stations. Both elements are divided into two categories, depending on their active or passive role in the model.

Multimodeling allows that a Station itself consists of a simulation model. This is enabled by the refinement attribute. The transfer of Load to the submodel is ensured by so-called TransferEvents. If the refinement is homogeneous, the transfer consists of a simple passing of the Load, otherwise the Load is transferred into the loadtype by a transfer rule. These TransferEvents cannot only be used to connect models vertically along the hierarchy of models but also laterally between models on the same level. In this case, it is possible to pass just information between models by TransferEvents (systems might interact without passing load).
The coupling of Stations is based on Events that initiate activities to be performed by a Station. With Conditions we are referring to information that is evaluated across the flows of load by using state information from multiple Stations. Conditions can form a network such as in System Dynamics, where information exchange between different resource flow systems is of major importance. Such networks are evaluated whenever a Station that activates the network changes its state.

The DESL provides the concepts for the definition of the building blocks of simulation techniques such as System Dynamics or Queuing Systems by instantiation as depicted in Figure 6. We will use the System Dynamics example to illustrate this technique definition process.

In SD the rates take the active part in the simulation of flows: they determine the filling or depletion of levels. Therefore, Rate is an instance of ActiveStation and Level is an instance of PassiveStation. Sources and Holes determine the system borders and are treated as PassiveStations with infinite capacity. Information transfer between flow systems is described by Constants and Auxiliaries which are instances of SimConstant or SimVariables, respectively. Externals indicate lateral information exchange between SD models by a TransferEvent. Finally, HetTransferEvents allow the exchange of Load between the Queuing Systems and the SD models that define their service time.

In MultiSim, the simulation technique models configure the editor according to the simulation technique. Embedding the model into the editor ensures the quality of the developed models with respect to the defined technique semantics.
APPENDIX B: SIMULATION MODEL VALIDATION

The philosophy of our modeling approach is that structure and local behavior of processes can be defined and estimated by the agents that perform the work with high quality. The emanating global dynamics have to be simulated and compared with the company’s reality. We therefore initiated a three step validation process.

During the first step, twelve interviews with department stakeholders of the three analyzed departments were conducted. We started with six unfocused interviews where we aimed at statements about the general situation of information management within their department. Then we analyzed the newly developed parts of the SD model structure by asking the engineers and workers about the relevance and correctness of auxiliaries, their relations, and the resources. These interviews led to a refinement of the overall model structure.

Afterwards, a questionnaire about the value of the variables was filled in by engineers not involved in the interviews (Return rate: 12 of 30). Multiple statements were collected about the self estimation of workload and overwork rates, the allocation of time to their various jobs, their personal error management, assimilation times of new hires and support for training, structure of task spectrum, time per task, meeting of schedules, information access along the business processes, and IS structure and usage. The models were parameterized according to the questionnaire results.

In a second step, the dynamics of the submodels were validated using black-box and plausibility tests with example processes. Besides a calibration of the model dynamics, this validation step provided the first results with respect to our paradigm of simulation usage: The engineers had to explicate assumptions they had about their company and learned about the business process dynamics of which only vague generalizations existed prior to simulation.

Finally, in order to check the results of the interviews against the company’s reality, the whole model for one business process (design – manufacturing – assembly) was tested using real production data of a previous year as a base case. The mix of projects resembles the situation at the company: The managers estimated the fraction of standard projects at about 60%; 30% were semi-standard and 10% were innovative projects. The simulation model was initialized with planned production schedules and expected task duration. The simulated throughput of projects at the Queue submodel of the departments was then compared to the real completion dates of the company.

Figure 7 shows this comparison of project throughput. The simulation results resemble the original throughput sufficiently. An exception are some “holes” (e.g., manufacturing around days 100, 150 and 180, or assembly around day 210). At these points one project passed by another along the process which could not be modeled in the queuing system. Therefore, the simulation is late at first and then catches up after the next project.

![Figure 7. The Validation Base Case](image-url)