J. ENG. DESIGN, 2001, VOL. 12, NO. 3, 177–199

# A decision support tool for predicting the impact of development process improvements

PETER CRONEMYR†*, ANNA ÖHRWALL RÖNNBÄCK‡ and STEVEN D. EPPINGER§

This paper presents a method for simulating the impact of improvements to the engineering design process. The method can be used by managers and teams to prioritize the most valuable process improvements among several suggested ones, before they actually take place. The method is based on the design structure matrix (DSM) developed by Steward (1981), and an extension of DSM called the work transformation model developed by Smith and Eppinger (1997). We introduce two new concepts, total process time and simulated to-be/as-is ratio. Two applications are presented. The first, a gas turbine blade development process, illustrates the estimated gain of a process improvement, and evaluates the actual implementation. The second application, a buyer-supplier product development project, shows how the method could be used as a decision support tool in an inter-organizational context. Input to the process simulation comes from process descriptions and estimates of anticipated effects of process change at the activity level. Output shows the effect of such a change on a total process level.

## 1. Introduction

### *1.1. Iterative development process*

The development process is often described as a sequence of project phases from conceptual design to production ramp up (see, for example, Andreasen and Hein 1987, Wheelwright and Clark 1992, or Ulrich and Eppinger 1995). While there have been quite extensive efforts to shift from sequential to parallel or concurrent tasks, the problem of iteration between coupled tasks is seldom addressed. Iteration within and between project phases is common in modern product development. Iterative development processes are discussed by Eppinger *et al.* (1994), Smith and Eppinger (1997), and MacCormack (1998).

Iteration is a somewhat negative word since it implies re-doing work. Hence the comment 'We have to get rid of iterations by doing it right the first time' is, or at least used to be, quite common in development process re-design. In development of high-tech products, where specialists from several domains have to be involved at the same time, iteration is not a negative word. Some iterations are of great value

Revision received March 2001.
† ALSTOM Power, SE-612 82 Finspong, Sweden
‡ Department of Management and Economics, Linköping Universitet, Institute of Technology, SE-581 83 Linköping, Sweden
§ Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
* To whom correspondence should be addressed. e-mail: peter.cronemyr@power.alstom.com

*Journal of Engineering Design*
ISSN 0954-4828 print/ISSN 1466–1387 online © 2001 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/0954482011005554 7

and even necessary. The problem is that iterations often take a long time to converge. Furthermore, the convergence criteria may change, or there may not be convergence criteria at all. To shorten the development time for a development project with iterations, Ulrich and Eppinger (1995) suggest two approaches.

(i) *De-couple tasks to avoid iterations*: If there is a well-understood interface between two interacting components of the product, it may be possible to de-couple development tasks to avoid iterations. This may be achieved by specifying the interface in advance.
(ii) *Perform faster iterations*: Many of the concurrent engineering tools available today may be used to perform coupled tasks more quickly, i.e. to perform more iterations quickly.

This paper presents a method for simulating the impact on development time of suggested process improvements such as avoiding or speeding up iterations. Fast iterations are required for fast developers, i.e. companies that are flexible and capable of rapidly adapting to new markets or technology conditions. Being a fast developer is the key to successful product development in many industries (Wheelwright and Clark 1992, Smith and Reinertsen 1998).

## 2. Modelling development processes and simulating process improvements

While much literature discusses the product development process on a highly aggregated level, consisting mainly of product development phases, the process maps that are used in this method are at a more detailed activity level.

### 2.1. Flow charts

Flow chart techniques, e.g. Gantt and PERT charts (Hillier and Liebermann 1986), are implemented in popular project management software such as MS Project. Planning of the activities is commonly carried out with the critical path method (CPM) where the sequence of activities taking the longest time determines the total project time. A problem, well known to PERT/CPM users, is that coupled-task iterations cannot be modelled. Instead, the time for iteration loops is estimated. In a Gantt or PERT flow chart, the iterations are therefore kept inside 'black boxes' of grouped activities. Such flow charts are not very useful for mapping iterative processes since the iterations are invisible, and consequently uncontrolled. Figure 1 illustrates sequential, parallel, and coupled activities. Figure 2 shows an example of an iterative process.

### 2.2. Design structure matrix

The design structure matrix (DSM), as developed by Steward (1981) and extended by Eppinger *et al.* (1994), is a technique based on representing the relation between any two activities as a binary (to, from) representation in the matrix cell. Each activity is represented by a row of input relations and a column of output relations. By partitioning the matrix, i.e. re-arranging the rows and columns in such a way that the activity relations are transferred to the lower triangle of the matrix, it is possible to find sequential, parallel and coupled activities (or activity blocks) in the matrix.[1]

[1] In this application of DSM, no partitioning is necessary. For more information on partitioning algorithms, see the DSM web site (http://web.mit.edu/dsm).

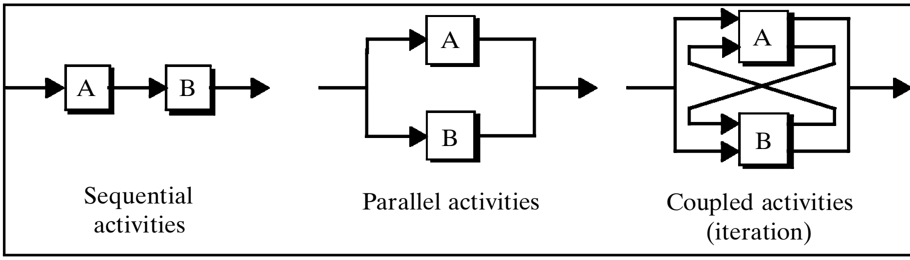


Figure 1. Possible sequences for related activities (adapted from Eppinger *et al.* 1994).

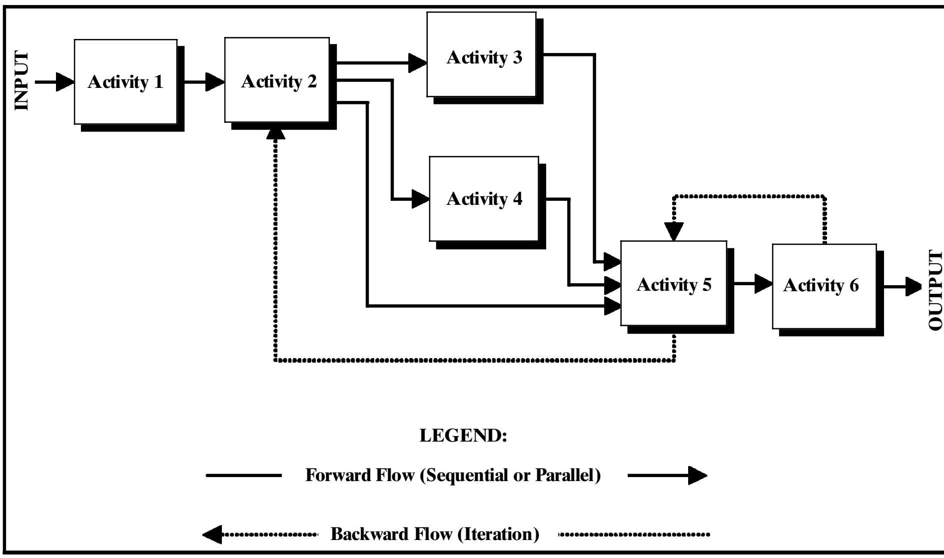


Figure 2. Process map of an iterative process.

To transform a flowchart of an iterative process into a DSM, one starts to rearrange the activities and arrows as indicated in figure 3. Then, when the activities are on the diagonal, with input arrows horizontal and output arrows vertical, it is easy to draw the DSM as can be seen in figure 4.

The marks in the matrix indicate what activity output is needed as input to other activities. This is an activity-based DSM (Eppinger *et al.* 1994). Other uses of DSM include parameter-based, function-based (Pimmler and Eppinger 1994), people-based, or team-based (McCord and Eppinger 1993) DSM analysis.

The marks can be exchanged for different icons to show different types of dependencies, e.g. input data needed to start an activity, input data needed to finish an activity, or input data needed to check an activity. Different icons or numerical values can be used to show different strengths of dependencies, e.g. weak dependency, medium dependency, or strong dependency.

### 2.3. *Work transformation model*

The work transformation model (WTM) (Smith and Eppinger 1997) is an extension of DSM. In a work transformation model, numbers are used instead of binary marks to show the strength of dependency expressed as rework fractions, with 0.0

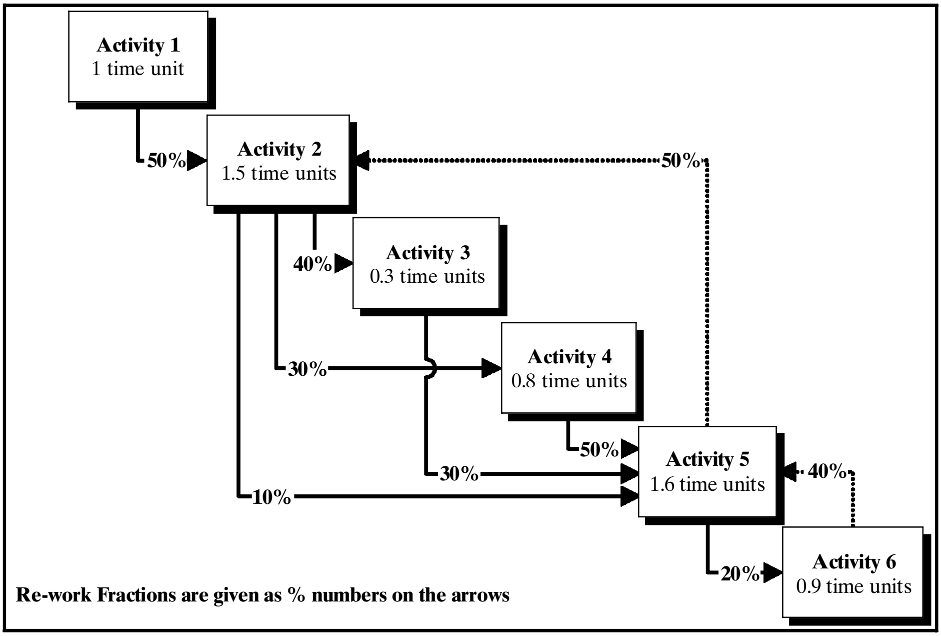


Figure 3. Rearranging the process map: put activities on the diagonal, make input arrows horizontal, and output arrows vertical.

meaning no rework and 1.0 meaning 100% rework when new information is transferred from an input activity.

In the first iteration loop, the activities are carried out without proper input data or perhaps with *guessed* input data. In the second and following iteration loops, the output data from the previous iteration are used as input data. This is called work transformation. If the process is convergent, then rework decreases as the iteration progresses. When the convergence criteria are satisfied, the iteration stops.

In a WTM, the DSM is split into two matrices: the work transformation matrix **A**, and the task time matrix **W**. In figure 5,the task times and rework fractions from figure 3 have been transformed into **A** and **W** matrices.

In the transformation matrix **A**, the off-diagonal elements represent the amount of rework that is generated for a certain activity when another activity is complete. For example, fully repeating activity 2 will lead to 30% rework of activity 4.

The numbers in the diagonal matrix **W** represent the amount of time it takes to execute the activities the first time.

### *2.4. Assumptions*

The work transformation model assumes that all activities are carried out in parallel at the same time (Smith and Eppinger 1997). The assumption of fully parallel activities is acceptable for many highly coupled, iterative processes, but does not represent a sequential process. Often there are some sequential activities within a parallel iteration process, but this may not affect the model results since these activities are inside a loop and will still be repeated many times. What may be a problem, however, is an activity that is only performed after a sub-process has converged, e.g. continue after accepted review. In that case, the matrix has to be divided into sub-matrices for the analysis.

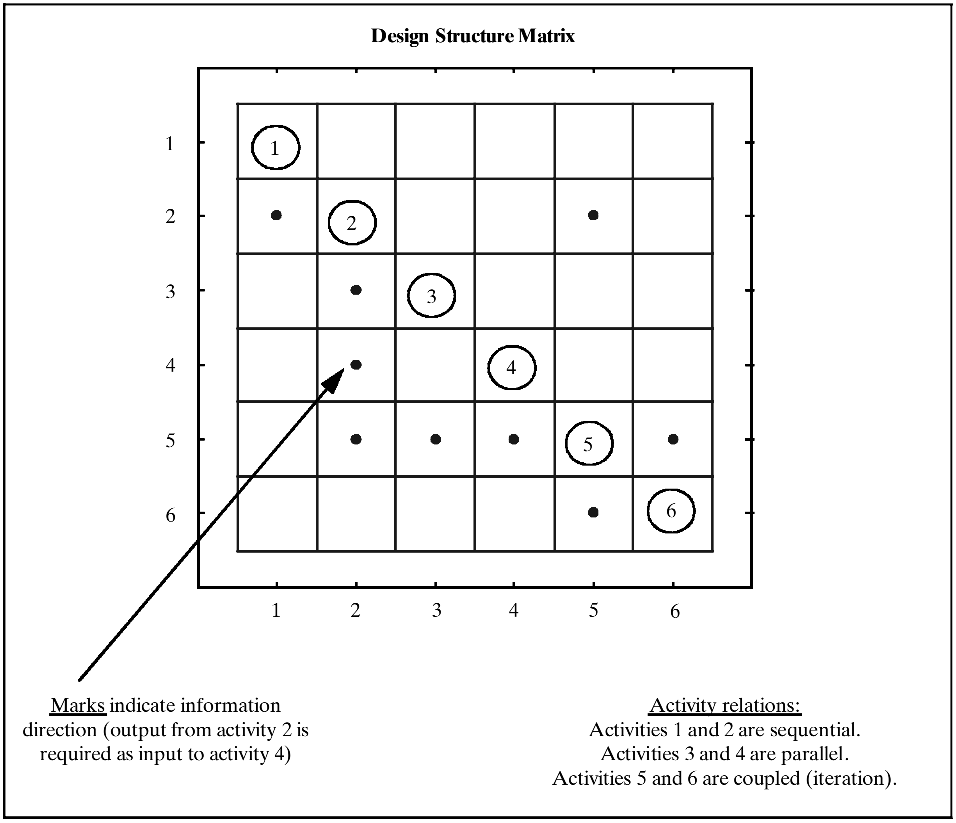


Figure 4. The design structure matrix (DSM).

The parallel assumption may look severe but, as was found in practical applications, it has a small influence on the final results given as total process time (TPT) and simulated to-be/as-is ratio (STAR). If one sub-iteration is converging quickly and another sub-iteration is converging slowly, the process does not converge until the slow iteration has converged. Then the activities in the fast sub-iteration have been executed several times with 'close-to-zero' rework. That may look strange but does not change the final result expressed as TPT and STAR.

Another assumption is that the matrix **A** is static and linear. It does not change with time and there are no conditional relations. In a changing process, these are somewhat heavy assumptions and the WTM may not be reliable. To analyse sequential iterations, conditional relations, and non-linear timing characteristics, more general modelling techniques, such as signal flow graphs, may be used (see, for example, Eppinger *et al.* 1997, Andersson *et al.* 1998).

Another fundamental assumption is that there is no noise, i.e. no stochastic variation, in the process. Another approach has to be used to deal with this, e.g. robust process design (Phadke 1989, Bergman 1992, Bergman and Klefsjö 1994).

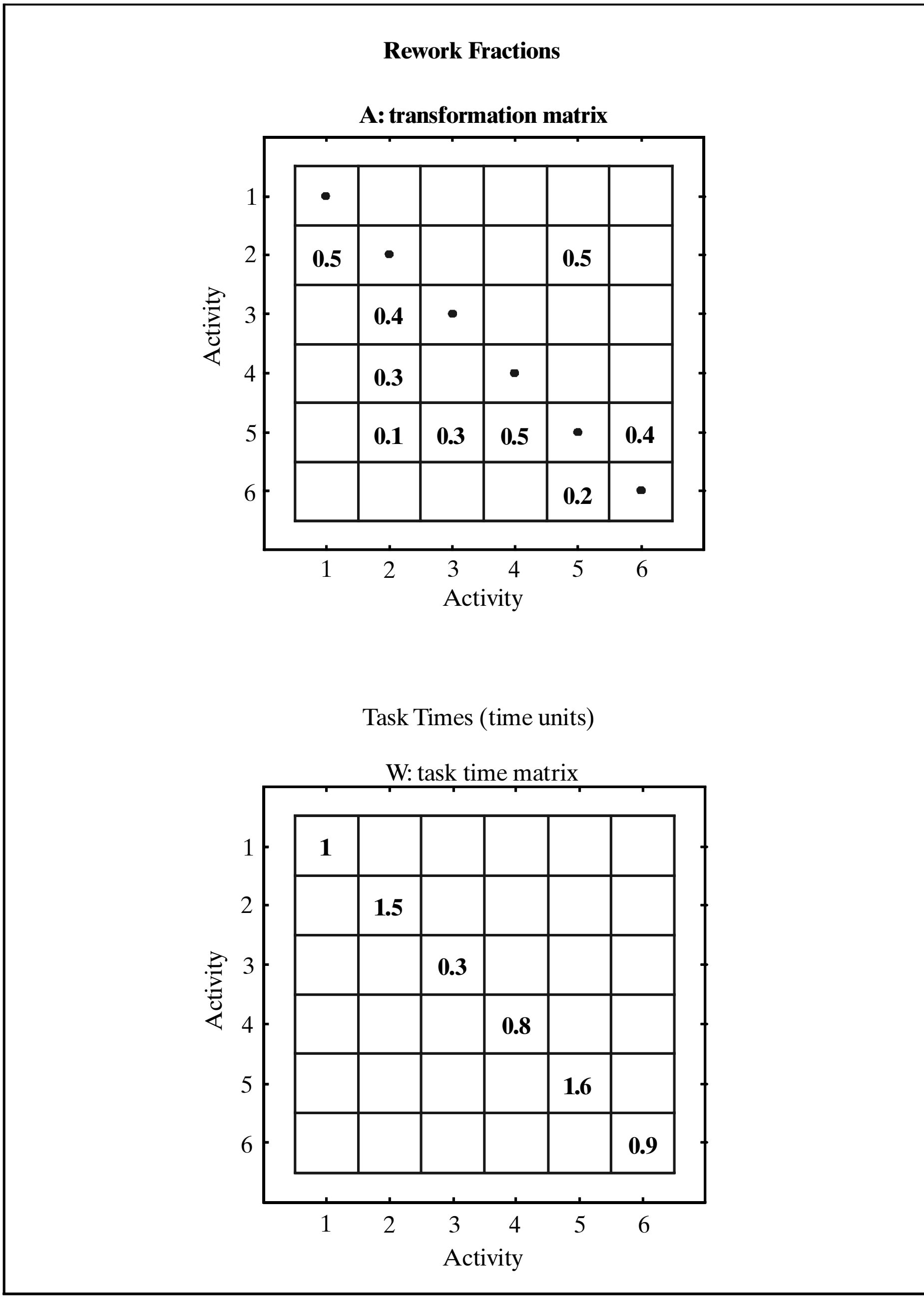


Figure 5. The two matrices of the work transformation model (WTM).

*2.5. Total process time TPT*

The work vector $u_i$ describes the amount of work done in each step $i$. The amount of work in the initial step ($i = 0$) is 1 for all activities, meaning that all activities are carried out once, thus:[2]

$$u_0 = [1 \ 1 \ 1 \ 1 \ 1 \ 1]^T \tag{1}$$

Due to the work transformation, the amount of rework in the following iteration loop is:

$$u_{i+1} = \mathbf{A}u_i \tag{2}$$

For a convergent process (see later), the total work vector is:

$$U = \sum_{i=0}^{\infty} u_i = \sum_{i=0}^{\infty} \mathbf{A}^i u_0 \tag{3}$$

By introducing eigenvalue decomposition, the infinite series can be solved exactly. Because the eigenvector matrix is often ill-conditioned, it may not give a robust solution. (For more details, see Smith and Eppinger 1997. However, eigenvalue decomposition is introduced for another purpose in the section 2.8.) Instead, as was found in practical applications, the series may be truncated after only very few iterations. The simulations presented in this paper (made with the software Matlab[3]) were often truncated after 10 iterations, satisfying a chosen convergence criterion.

The total work vector may be scaled with the actual duration of each activity, i.e. the task times in the diagonal matrix **W**. Thus, the total process time vector $\mathbf{W}U$ is:

$$\mathbf{W}U = \mathbf{W}\sum_{i=0}^{\infty} \mathbf{A}^i u_0 \tag{4}$$

and the scalar TPT is the sum of all elements in the total process time vector:

$$\text{TPT} = \sum_{j=1}^{n} (\mathbf{W}U)_j \tag{5}$$

where $n$ is the number of elements in the total process time vector $\mathbf{W}U$.

*2.6. Convergence of iterations*

In the first iteration step, all activities are fully carried out, i.e. they are done once ($u_0$ equals a vector of ones). The process time for the first step is the sum of the diagonal elements of **W**. The work increments will decrease in the following steps if the process is convergent. A sufficient, but not necessary, criterion for convergence is that the entries in either every row or in every column of the transformation matrix **A** sum to less than one. In the example in figure 5, this is true for all columns but not for the rows. As can be seen later in figure 8, the process in the example is convergent. After only 10 iterations, the TPT reaches a stable value (in this example,

[2] The dimension is equal to the number of activities, $n$.

[3] Matlab, Trademark of The MathWorks Inc.

15.9 time units). In real product development processes, iterations with very small changes near convergence are not performed. Instead, the process is stopped after the major iterations are completed.

### 2.7. *Simulated to-be/as-is ratio STAR*

The calculated TPT is a measure of the total amount of working time, i.e. effort, required for the process to converge. TPT has units of work (e.g. person-hours).[4] However, TPT expressed as an absolute value is of somewhat limited value. Instead, TPT for the given as-is process and TPT for the changed to-be process are related. The change introduced to the process, e.g. a new software or a new work method, is a 'small disturbance' to the system. The process map stays the same but the numbers change. Hence, the STAR is a sensitivity measure of a suggested improvement as a linearization around an existing process.

$$\mathrm{STAR} = \frac{\mathrm{TPT}_{\text{to-be}}}{\mathrm{TPT}_{\text{as-is}}} \tag{6}$$

STAR is a relative measure of the overall impact of a suggested process improvement.

### 2.8. *Identifying potential for improvement*

One question that emerges is which improvements to suggest. There may have been several suggestions already before the process map was developed and those could of course be examined, but there is another way to find out where potential improvements could have most effect.

Each eigenvalue of the transformation matrix **A** represents the convergence rate for one possible iteration mode in the process. The iteration mode with the slowest convergence rate is identified by the largest eigenvalue that always is positive and real (Smith and Eppinger 1997). The eigenvalues and eigenvectors of **A** are found by decomposing **A** according to:

$$\mathbf{A}\boldsymbol{V} = \boldsymbol{V}\mathbf{D} \tag{7}$$

where **D** is a diagonal matrix of the eigenvalues of **A**, and $\boldsymbol{V}$ is the corresponding eigenvector matrix. Note that it is not necessary to invert the eigenvector matrix that often is ill-conditioned. The eigenvalues and eigenvectors presented in this paper were easily solved by the use of Matlab.

The eigenvalues (the diagonal elements of **D**) and the first eigenvector (the column in $\boldsymbol{V}$ corresponding to the largest eigenvalue) of **A** for the iterative process in the example are given in figure 6. As can be seen, the largest eigenvalue is positive and real. Furthermore, the eigenvector associated with this eigenvalue has positive (and real) elements. Each of these elements characterizes the relative contribution of one activity in the slowest converging iteration mode. The interpretation of the other eigenvalues with corresponding eigenvectors is more difficult and hence no more than the first mode is used here. (Smith and Eppinger (1997) present a more thorough interpretation.)

The interpretation of the slowest converging iteration mode is that activity 5 has

[4] To compute calendar time instead of effort, Smith and Eppinger (1998) propose modified equations.

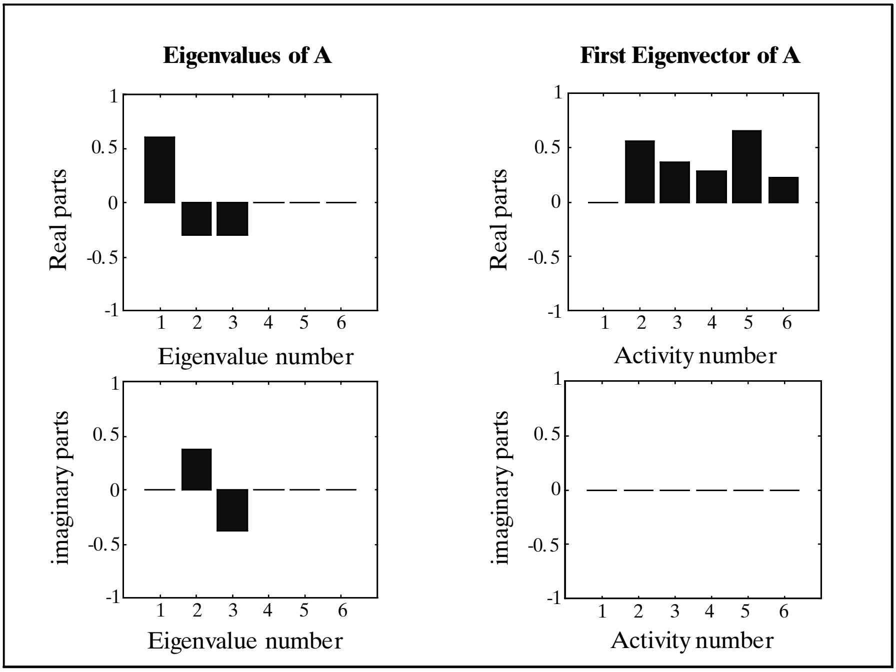


Figure 6. Eigenvalues and first eigenvector of A.

the largest contribution. Also, by looking at the task time matrix in **W** (figure 5) we see that activity 5 has the longest task time. It is evident that, to reduce the total process time and hence get a lower STAR, we should concentrate on activity 5, its task times and its rework fractions.

*2.9. Comparison of suggested improvements*

Two different improvements in activity 5 are suggested (figure 7). Improvement #1 is a make-less-iterations improvement, e.g. better interface to activity 5. All rework fractions in row 5 and in column 5 of the **A** matrix have been divided by two. Improvement #2 is a work-faster improvement, e.g. faster tools for activity 5, hence the task time for activity 5 in the **W** matrix has been divided by two.

In figure 8, the TPT, the $U$ vector (total work vector) and the $\mathbf{W}U$ vector (total process time vector) after 10 iterations are given.

In the left part of the figure, the TPT, $U$ and $\mathbf{W}U$ are given for the as-is process and for a suggested improvement #1. In the right part, the as-is process is compared with suggested improvement #2.

For the make-less-iterations improvement, the work as well as the amount of time decrease considerably for activities 2 and 5, and some also for 3, 4, and 6. The TPT after 10 iterations is 10.1 time units and, hence, the STAR is 10.1 / 15.9 = 0.64.

For the work-faster improvement the $U$ vector stays unchanged but the TPT decreases because the total time required by activity 5 decreases, giving TPT = 12.9 time units and STAR = 12.9 / 15.9 = 0.81.

One of the most interesting things to notice is that improvement #1, in addition to giving a lower STAR, also gives a much faster convergence than the as-is process and improvement #2. Look especially at the time/iteration curves (lower curves) in

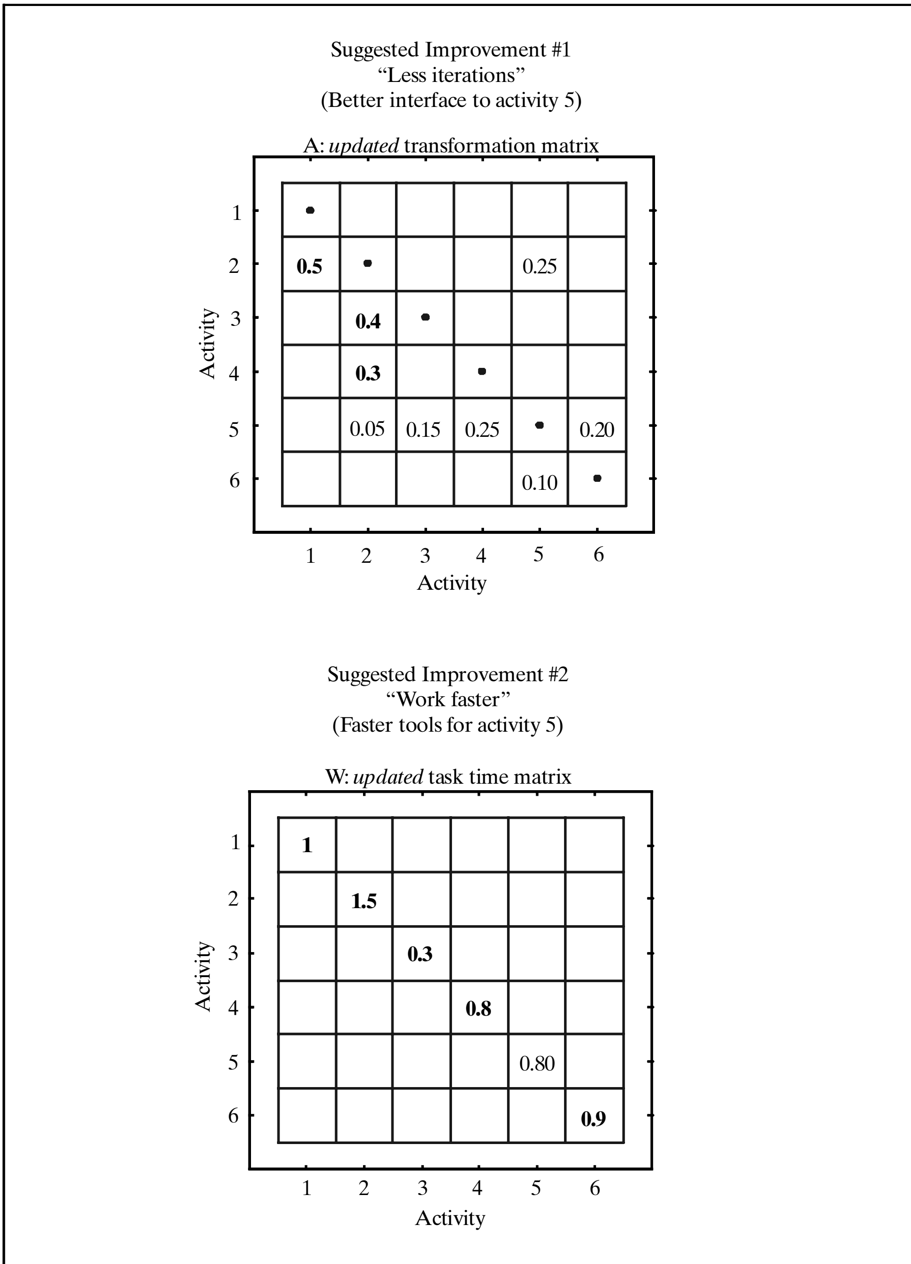


Figure 7. Two different improvement suggestions.

figure 8. The number of necessary iterations decrease by half. In reality, the most successful improvements are combinations of fewer iterations and working faster. A combination of the two improvements in this example, i.e. to use both the new **A** matrix and the new **W** matrix, would shorten the TPT even more, giving a STAR = 0.54, i.e. a total time reduction of 46%.

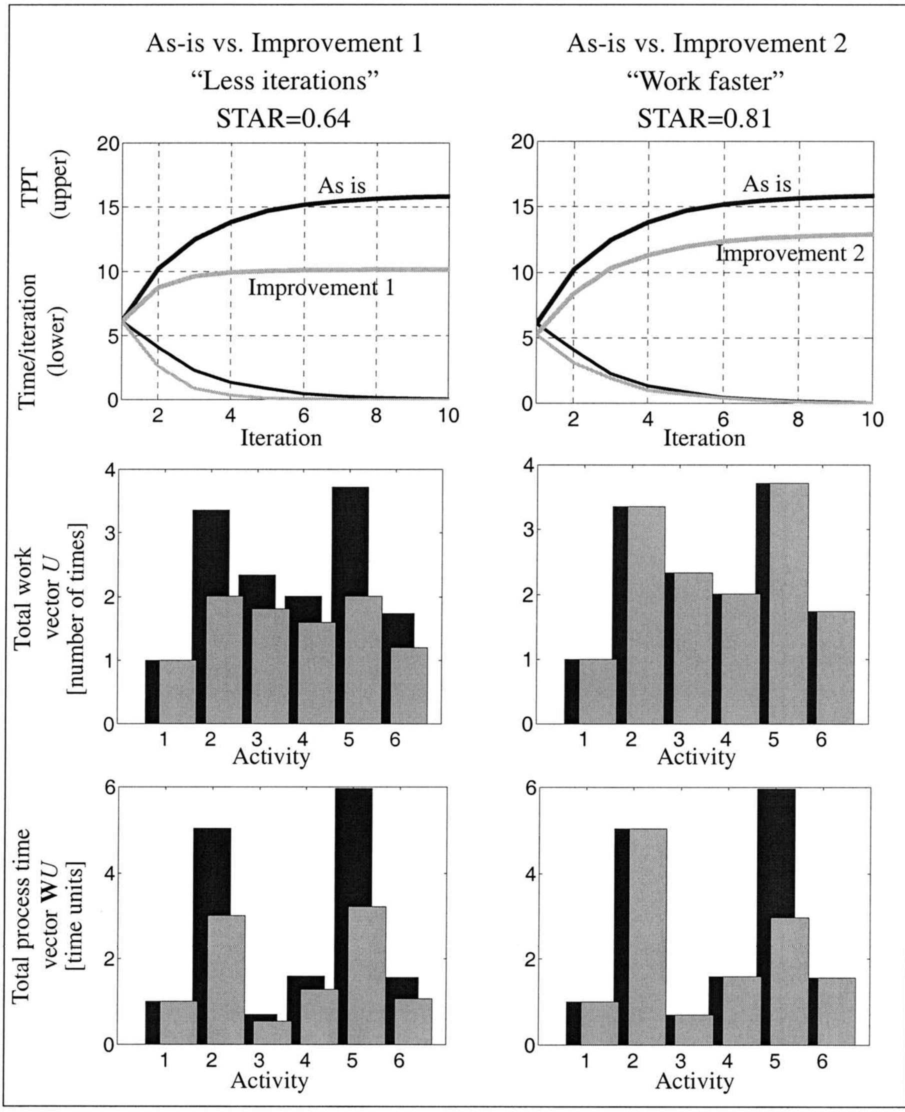


Figure 8. Simulation results from two different suggested process improvements (as-is, dark bars at back; improvements, lighter bars in front).

## 3. Application to gas turbine blade development

In this section, we describe an application of this method to a gas turbine blade development process at ABB STAL.[5] Among the most critical components of a gas turbine are the blades. Due to the extreme technical demands on gas turbine blades, the development of the blades is a very complicated, coupled process. It includes several specialists from several technical domains. In an effort to achieve increased integration, a project was launched at ABB STAL with the mission of mapping the

[5] Since 1999, ABB STAL has become a part of ALSTOM Power (www.se.alstom.com).

as-is process and to suggest improvements in a to-be process map for gas turbine blade development.

### 3.1. *As-is process maps*

The as-is process map (figure 9), was developed over several months by a small project team, consisting of seven engineers normally working with gas turbine blade development (among them, one of the authors of this paper). The main process consisted of three sub-processes: aerodynamic design, mechanical design and verifying mechanical integrity. The main reason for this division was that the activities of those sub-processes were carried out by engineers from different departments with names similar to those of the sub-processes. By looking at the process map in figure, 9 it is obvious that the activities and the sub-processes are highly coupled. Hence the map was transformed to a DSM (figure 10). Also, a task-time matrix **W** and a transformation matrix **A** for the as-is process were created but, since the numbers are not for public display, only the binary DSM is shown in this article.

### 3.2. *Simulations of suggested improvements*

Only a short while after the process team had begun working with the as-is map, they discovered things that could be improved. Even though those suggested improvements often were obvious and easy to fix, they had not been suggested earlier since they dealt with matters that had fallen between the responsibilities of the different departments. Some of these improvements were implemented at once, since there was no reason to wait if it could easily improve the process.

Other process improvements that were suggested from looking at the as-is map were more extensive and could not be implemented at once because the project group did not have the authority to implement them. Such improvement suggestions included change of computer-aided design/computer-aided engineering (CAD/CAE) software and earlier manufacturing involvement.

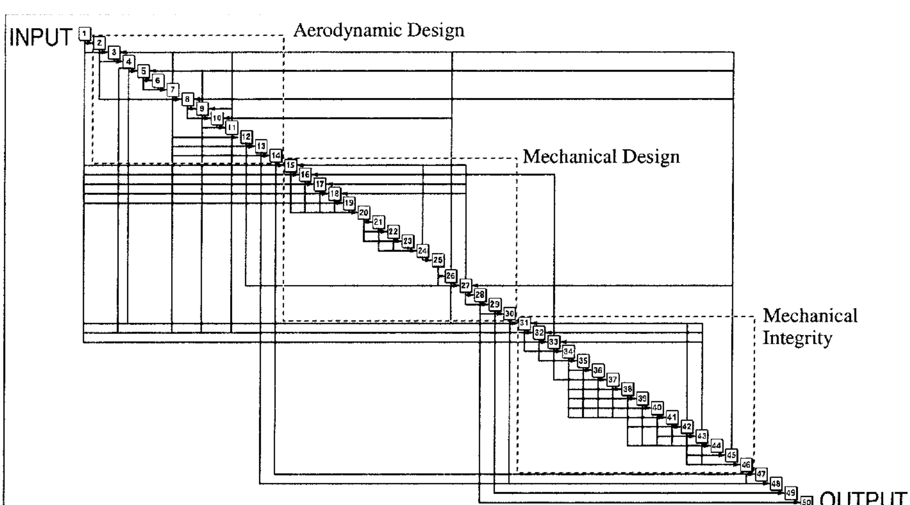


Figure 9. As-is process map for the gas turbine blade development process (courtesy of *ABB Stal* 1997).

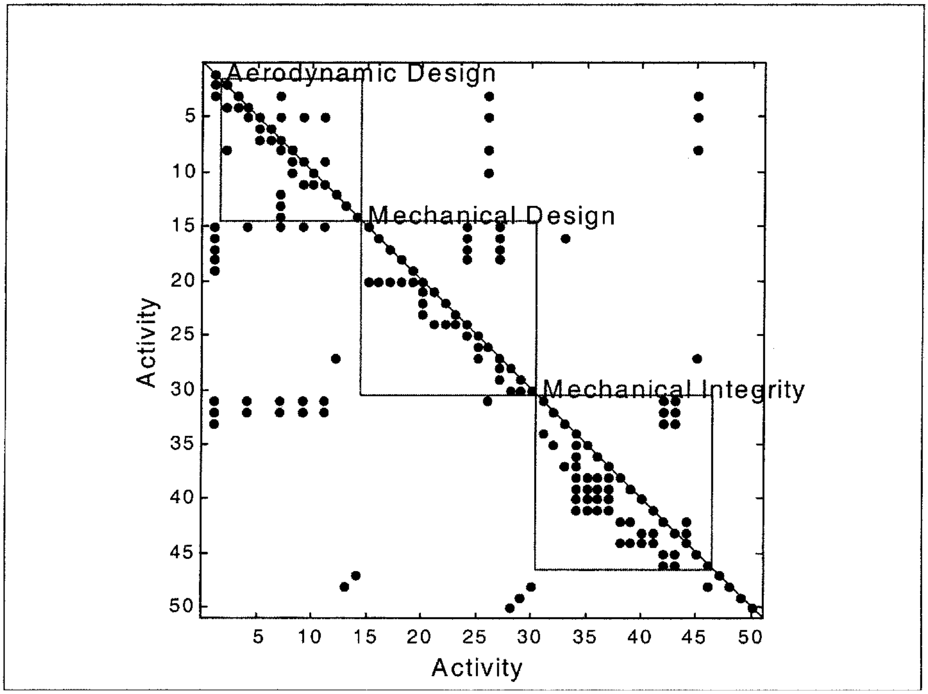


Figure 10. DSM for the blade development process. Numbers in **A** and **W** matrices are indicated by marks (not for public display).

The discussion on choice of a common CAD/CAE software had been going on for a while. The stress engineers claimed that, if the mechanical designers and the stress engineers used a common CAD/CAE software utilizing parameterized geometry (the one that the stress engineers already used for finite element pre- and post-processing), it would shorten the development time considerably. People in other departments thought other aspects of the choice of CAD/CAE software were more important and did not really think that the stress engineers should have opinions on the tools within mechanical design.

Analysis of the highest eigenvalue and the corresponding eigenvector of the **A** matrix showed that the slowest converging iteration mode consisted of activities, with a lot of rework in each iteration, mainly within mechanical integrity (see figure 11). It also consisted of some activities, but without a lot of rework, within mechanical design. This meant that the stress engineers had to re-do a lot of work every time a changed geometry was delivered from the mechanical designers. This supported the suggestion of a common CAD/CAE software for these two domains. Therefore, a study was made with the purpose of measuring the task times and rework fractions for finite element modelling for different CAD/CAE software combinations at the departments.

The results from the study, presented in table 1, were implemented in the **A** and **W** matrices (see figure 12) and used for TPT calculations. Results from simulations showed that the simulated to-be process, compared with the as-is process, has STAR = 0.83 (see table 2). The overall development time for the gas turbine blades would decrease by 17% by switching the CAD/CAE software of the mechanical design department. By inspecting the total work vectors in figure 13, we notice that the

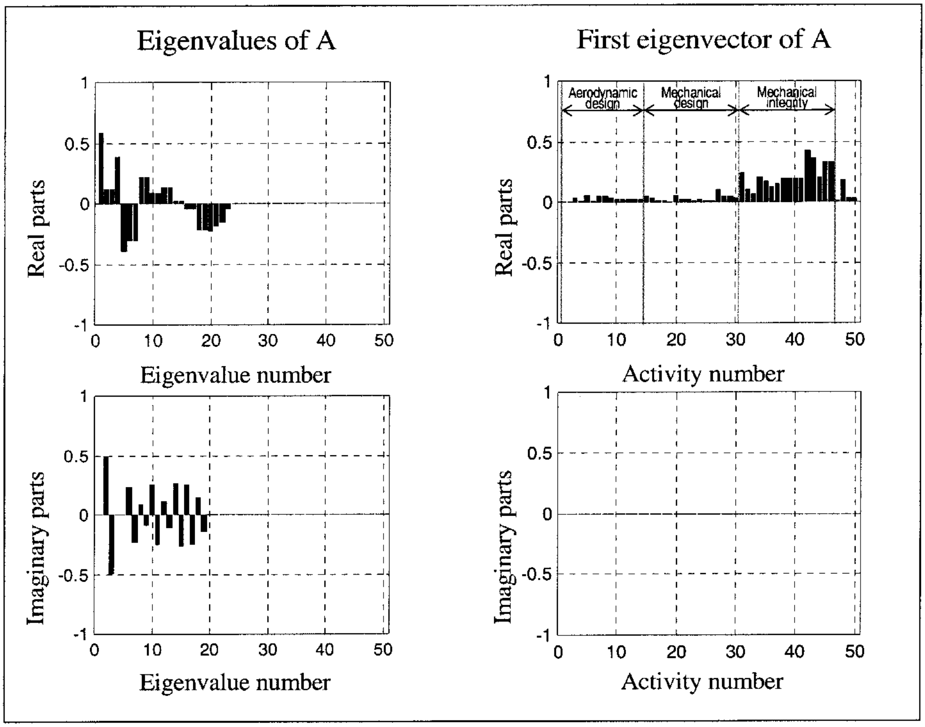


Figure 11. Eigenvalues and first eigenvector of **A** for the blade development process.

| Modelling time for different processes | Create first finite element model | Updating blade geometry | Ratio |
|---|---|---|---|
| *As-is process* | | | |
| Different CAD/CAE software (not parameterized geometry) | 80 (time units) | 56 (time units) | 0.70 |
| *To-be process* | | | |
| Common CAD/CAE software (not parameterized geometry) | 20 (time units) | 1 (time unit) | 0.05 |

Table 1. Results from study of blade modelling times.

| Suggested improvement | $TPT_{\text{to-be}}$ (time units) | $TPT_{\text{as-is}}$ (time units) | STAR |
|---|---|---|---|
| Common CAD/CAE software | 193 | 231 | 0.83 |

Table 2. STAR for suggested improvement: common CAD/CAE software.

amount of work decreases considerably for the stress engineers within the mechanical integrity process (activities 31–45) but not that much within the other two subprocesses. There were also opinions that the new software was a better CAD software, and hence the total work in the mechanical design process would decrease. Furthermore, the use of common geometry models would increase the cross-disciplinary cooperation, but none of those matters have been taken into account in

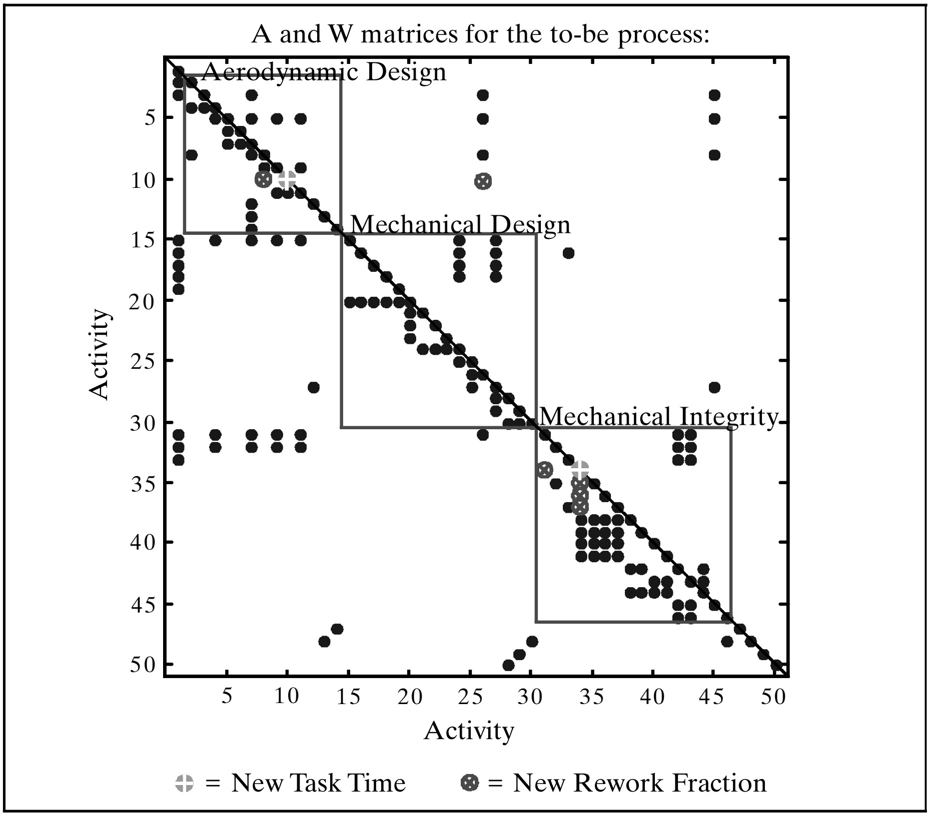


Figure 12. Improvement suggestion: common CAD/CAE software.

the present paper. Only new task times and rework fractions for finite element modelling have been included in the simulation. Still, 17% of the total person-hours for such a development project is a lot of money, hence the company decided to make the suggested investment.

Installing the common CAD/CAE software for the gas turbine development team led to many problems, as one could expect. To change the team software, they also had to adapt the process. After some months, the people had adapted to the new software and the simulated improvements were realized. The stress engineers confirmed the to-be finite element modelling times from the study and the engineers in the team thought that the iterations had become faster.

It is not *quantitatively* proven that the faster iterations were actually making the process time shorter. Since the engineers are in the middle of the process, there is no way for them to measure the new total process time. *Qualitatively*, however, the engineers and project managers expressed in discussions that ‘things got finished faster’.

## 4. Application to buyer–supplier cooperation in aircraft development

In the military aircraft industry, high complexity of products combined with shrinking national defence budgets (Augustine 1983) have forced companies to cooperate in product development. The tradition to keep the supplier at an arm’s length distance, both for cost reduction and security reasons, is now rapidly changing

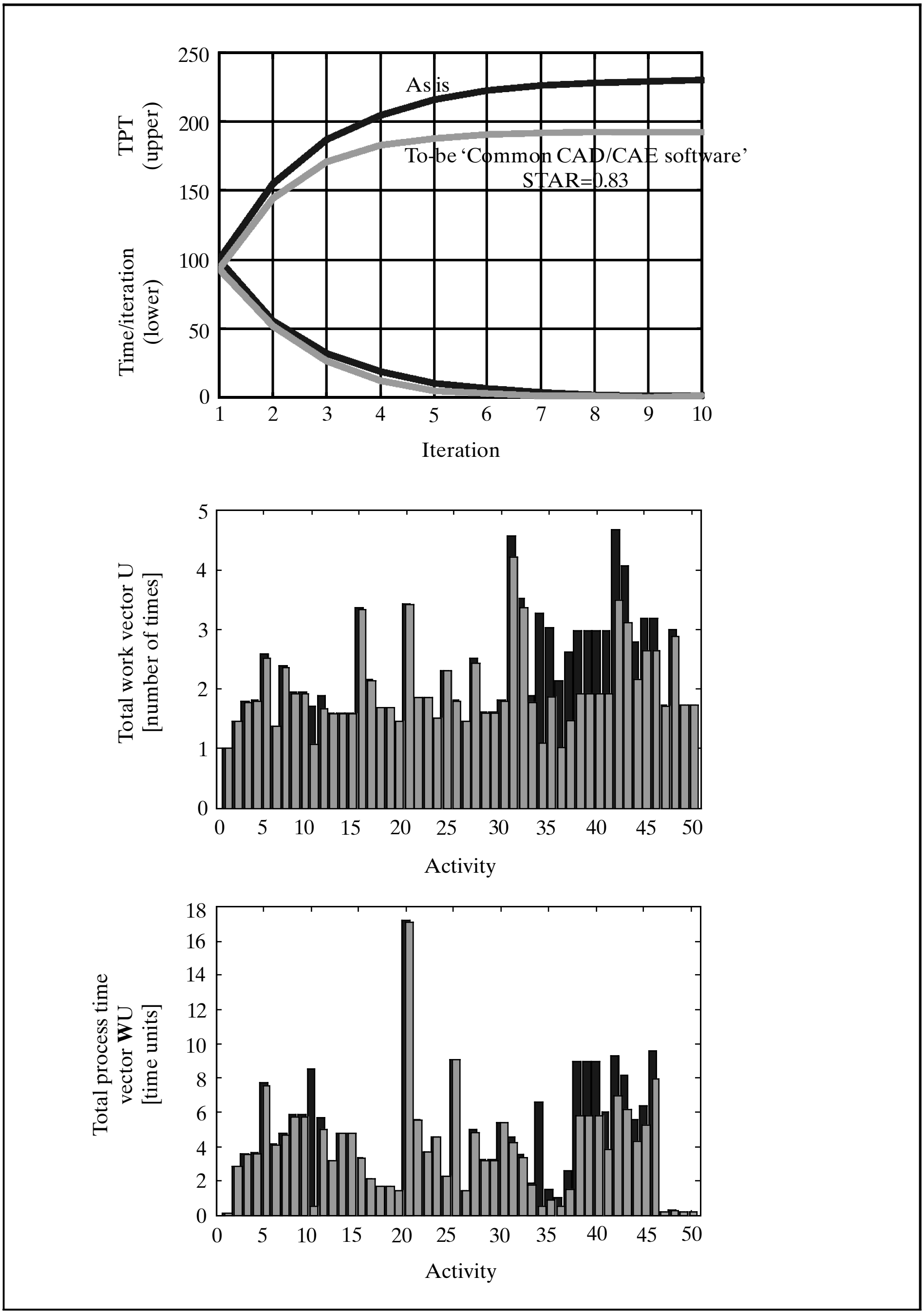


Figure 13. Simulation results from the suggested process improvement: common CAD/CAE software.

towards a partnership behaviour built on trust between the parties. Such collaboration between organizations and over national borders requires efficient communication. This is emphasized in collaborative product development, especially in the early phases where the uncertainties are considerable about technology choice, product characteristics, or teamwork cooperation methods, and contracts cannot specify all details of the forthcoming product (Helper 1996).

In studies carried out in the aerospace industry during 1997–1999,[6] it was found that engineers needed better communication support, and saw the benefits of a web-based communication platform as a complement to telephone, fax, e-mail communication, and physical meetings. However, although prototypes had been constructed, a full-scale tool had not, at the time of the study, been successfully implemented. One important reason was a hesitation to further invest in information technology tools since the return on investment in information technology (IT) during the 1990s had been difficult to measure. Therefore, both managers and product development engineers asked for better decision support tools to estimate expected effects of IT investments.

*4.1. Process map according to plan*

The process map of figure 14 shows the collaborative product development activities between a systems integrator and its major supplier. It is generalized mainly from the studied development cases.[7] The cases are described from both the buyer and the supplier perspective, where project activities (boxes 1–28) are conducted by integrated product development teams (IPTs), one in the supplier company and one in the systems integrator company. The arrows show the dependencies between activities.

It was found that development was carried out at the partner companies separately, with some common activities where the parties meet. The common activities are shown in the middle, together with some of the most important documents that are exchanged, such as the Request for Information and the Request for Quotation in the beginning, the Contract and the Requirement Specification, and the Final Certificate at the end of the project. To give an overview of the process activities, they are grouped into three main phases: the concept phase (activities 1–9), the design phase (10–21), and the test and integration phase (22–28). In the DSM (see figure 15), the dependencies between the activities in the three phases are shown.

*4.2. Simulations of suggested improvements*

The suggested improvements[8] presented in this paper are based on the idea that a common project web site for communication between the systems integrator and the major supplier could be used in all steps of the design phase, after the contract

---

[6] This included approximately 50 interviews and meetings with engineers and project managers of three aerospace companies in Sweden, US and France, where the collaborative product development activities for subsystems of Swedish JAS 39 Gripen were mapped (see Sundstrand 1993, Saab 1998, Microturbo 1998), time consumption on activity level was estimated, and process improvements were discussed. These studies were complemented with studies of web application development and implementation at Saab AB and at Boeing Commercial Aircraft Group. Some additional data come from Linnarsson and Molin (1998). Refer to Öhrwall Rönnbäck (1999) for a complete review.

[7] The IPTs included between 30 and 70 engineers at each company.

[8] Estimated task times and rework fractions derive from meetings and discussions with product development managers and engineers (see Öhrwall Rönnbäck (1999) for a complete list of estimations).

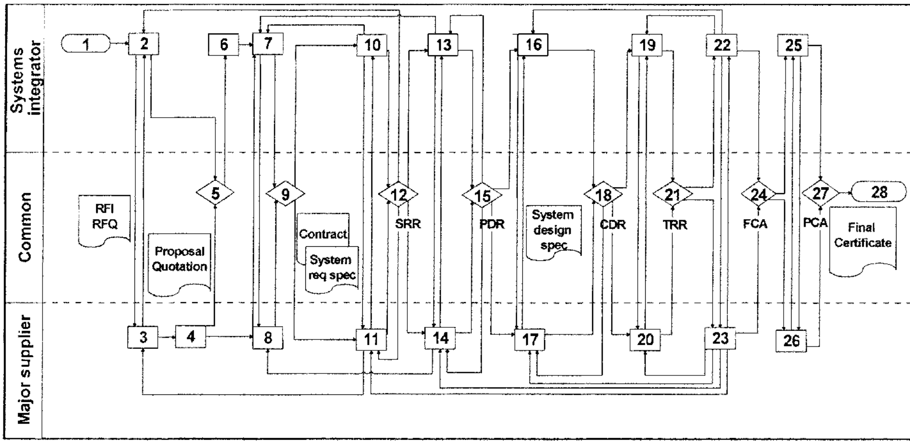


Figure 14. Process map for the collaborative system development.

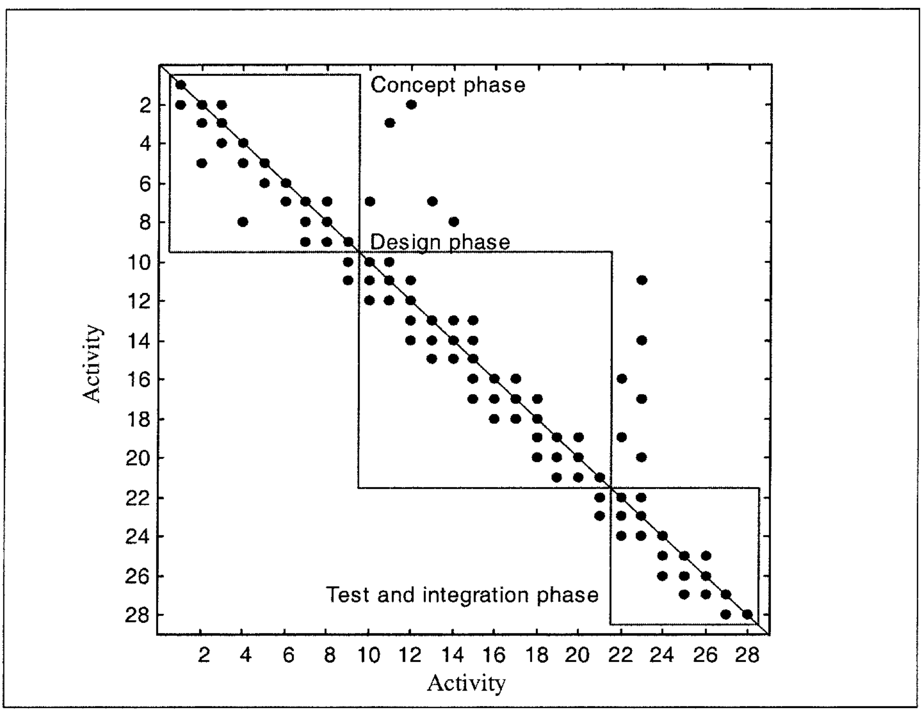


Figure 15. DSM for the collaborative development process.

is signed, i.e. after the concept development phase. The interviewed engineers suggested the functions stated in table 3 of a common web-based platform.

Of these, the first five are quite easily implemented in a web-based communication platform,[9] while exchange of product data is more complex, requiring

[9] Prototypes containing this functionality had already been produced during the time of the study in the research project LARP (www.liu.se/org/imie/larp), for M.Sc. courses at the university department (www.eki.liu.se), and in the com-onweb™ project (www.com-onweb.com).

| Number | Function | Description |
|---|---|---|
| (1) | Project data | Contact lists and photos of people involved in the project, action lists, schedule, documents (such as project specifications, minutes of meetings, etc.), budget and resource planning, project history, and national specifics, such as a calendar where holidays are marked |
| (2) | Update alert and filtering | Access levels and filtering, in order to direct information to the right person and avoid problems of information overflow |
| (3) | Informal communication forums | Electronic notice boards, sorted on different discussion themes, equipped with text search engines |
| (4) | Work process and rules | Project management methods, work process descriptions, etc. |
| (5) | Links | Interactive links to common information sources, such as standardization organizations, common suppliers, customer sites, universities, etc. |
| (6) | Product data | Requirement specifications, change requests, engineering data, such as early product models, CAD data, simulation digital prototyping, integration and test results, configurations, production planning, and cost estimations |

Table 3. Functions of a common web-based platform as suggested by interviewed engineers.

common standard formats and an inter-organizational product data management system. At the time of the study, commercial full-scale tools supporting this were not found in use, and therefore the simulation was carried out for functions (1)–(5).

The to-be process is called the 'common project web' in figure 16, where the simulated changes from the as-is to the to-be process are shown. The following assumptions in the to-be process were made. At each company, one or two persons ('web masters') were added with the task of helping project leaders and teams to manage project information and to teach them how to use the project web. Information published on the project web makes it quickly available for the person who needs it, enabling time reductions of product development activities, with larger gains at the supplier. The common project web gave no time reduction for review meetings. Today, misunderstandings in the later phases often derive from lack of communication in the early phases. In the to-be process, more iterations (larger dependencies between activities) were estimated in the early phases of the development process, but fewer in the late phases, since increased information sharing between the IPTs enhances early conceptual discussions.

The results (see table 4) show that the total process time of 224 time units with the current communication tools could be reduced to 150 if a common project web is used, although the number of persons is increased by three to four persons throughout the project. As a decision support tool, this method can provide guidance on questions such as:

- If the resources are limited, which tools, or part of a tool, should be implemented?

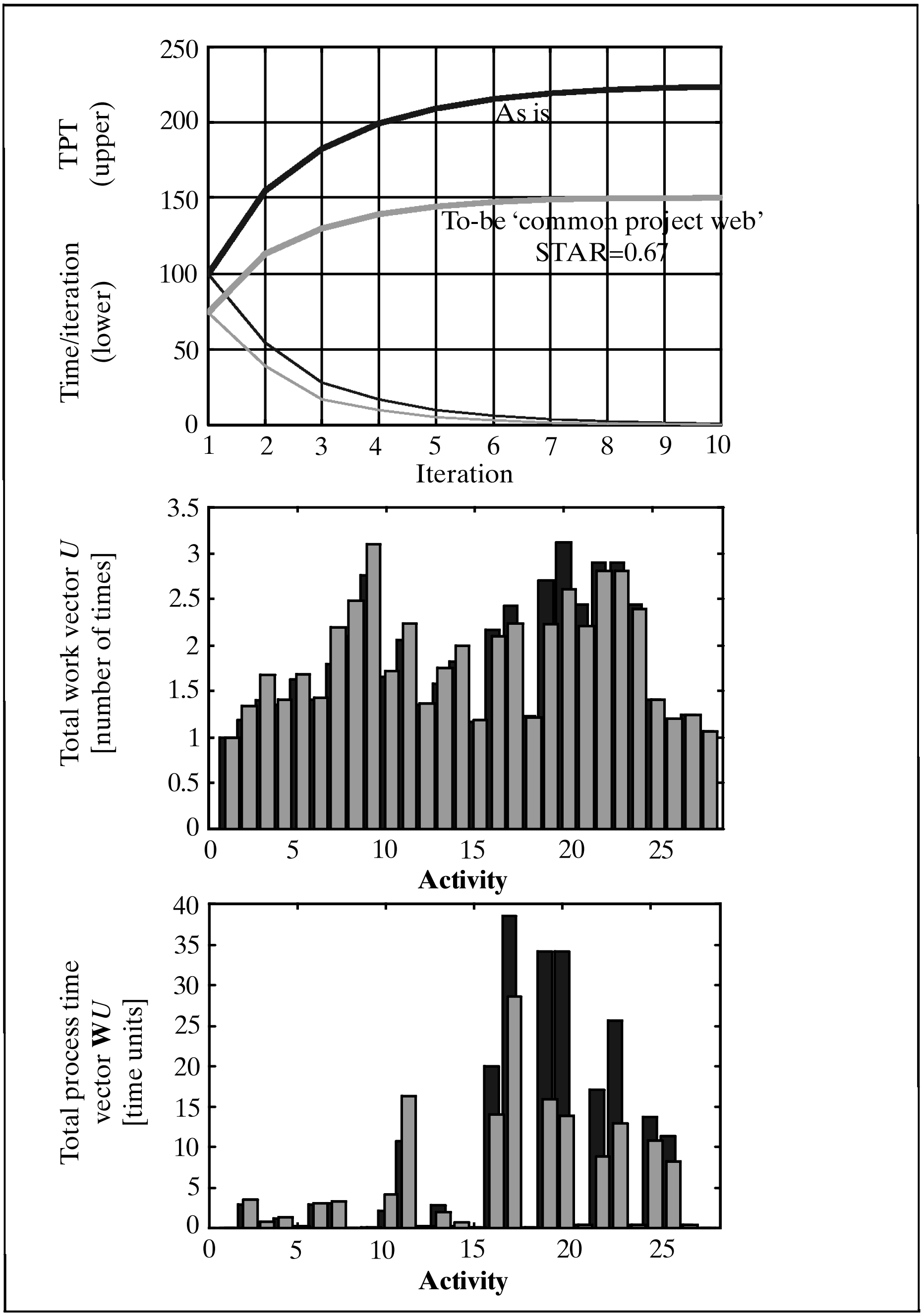


Figure 16. Simulation results from the suggested process improvement: common project web.

| Suggested improvement | $TPT_{to\text{-}be}$ (time units) | $TPT_{as\text{-}is}$ (time units) | STAR |
|---|---|---|---|
| Common project web | 150 | 224 | 0.67 |

Table 4. STAR for suggested improvement: common project web.

- Is it worth introducing a new tool late in a project? Where does it give most effect?
- Can we afford support personnel?

*4.3. Potential for process improvement*

When a process simulation indicates high economic impact, the investment decision will most likely be prioritized on top management's agenda, and other related issues will be considered in relation to potential savings. For investment in a web-based tool, an issue to discuss could be Internet security compared with the company's current information management procedures.

The simulated improvements based on a common project web with quite simple functionality indicate a 33% shorter development process. When more extensive web-based exchange of product data is realized, larger time gains can be expected. Cost gains are expected if a larger part of the new system concept can be evaluated with digital prototypes, communicated over the common web.[10] The expensive tooling and production ramp-up activities would then create opportunities for better accuracy, and at lower cost. To integrate customer and sub-tiers in the common communication platform could also lead to substantial improvements. With the process simulation method, such gains can be estimated by analyzing the development activities. The gains can then be related to anticipated costs for implementation, training, and the collocation required early in the product development process to make the solution effective.

It is clear that the simulated to-be/as-is ratio is an important indicator of how web-based communication tools could change product development cycle time. However, it does not answer questions of long-term consequences on the relationship between partners. That needs to be investigated through in-depth case studies.

## 5. Discussion and conclusions

Task-based DSM is one of very few tools for mapping and visualizing iterative processes. From a DSM, it is easy to get a holistic view of a complex development process where flow charts become too complicated.

In this method, processes are studied at a very detailed level. The WTM and the STAR concepts are easy to use for process improvement simulations. The assumptions made in the method are linearity and stability of the process and fully parallel task execution. The parallel assumption may look severe but, as was found in practical applications, that has a small influence on the final results given as TPT and STAR. If these assumptions correspond to the actual process, the output provides good guidance for further process development and improvement.

[10] Tools such as inter-organizational PDM systems are now becoming available on the market (e.g. Eurostep's Share-A-Space and PTC's Windchill), and results pointing in this direction are being presented, e.g. by Boeing (Holly *et al.* 2000).

Of course, the method demands a detailed process map. One could argue that the main process improvements come from the knowledge gained from the process mapping itself, i.e. once the map exists, the suggestions for improvements become obvious.

On the contrary, it is the authors' experience from process development that it is difficult to get the authority to implement major changes in an organization if it cannot provide an estimate of the payback on the investment. The STAR is such an estimate, since it quantifies the impact from a suggested improvement on the overall process performance, expressed as reduction of total process time. Hence, it is a good tool for speeding up improvements.

**Acknowledgements**

The authors would like to thank the sponsors and tutors at ENDREA (The Swedish Engineering Design Research and Education Agenda) and IMIE (International Graduate School of Management and Industrial Engineering). They would also like to thank cooperation partners ALSTOM Power (formerly ABB STAL), SAAB AB, Sundstrand Power Systems, and Microturbo SA.

The main research work for this paper was carried out during 1997–1998. Some of the material in this paper has previously been presented in Cronemyr *et al*. (1999), Cronemyr (2000) and Öhrwall Rönnbäck (1999). Peter Cronemyr contributed to the extension of the theory of WTM and the ABB application. Anna Öhrwall Rönnbäck contributed to the Saab application. Steven D. Eppinger introduced DSM and WTM in an ENDREA doctoral student course and has guided this research work.

**References**

*ABB Stal*, 1997, The blade development process, Report, November.

Andersson, J., Pohl, J. and Eppinger, S. D., 1998, A design process modelling approach incorporating non-linear elements. Procedings of the ASME Design Theory and Methodology Conference, 13–16 September, Atlanta, GA.

Andreasen, M. M. and Hein, L., 1987, *Integrated Product Development* (London: IFS Publication Ltd).

Augustine, N. R., 1983, *Augustine's Laws, and Major System Development Programs* (Reston, VA: American Institute of Aeronautics and Astronautics).

Bergman, B., 1992, *Industriell Försöksplanering och Robust Konstruktion* (*Industrial Design of Experiments and Robust Design*) (Lund: Studentlitteratur) (in Swedish).

Bergman, B. and Klefsjö, B., 1994, *Quality from Customer Needs to Customer Satisfaction* (Lund: Studentlitteratur).

Cronemyr, P., 2000, Towards a learning organization for product development. Licentiate Thesis, Division of Quality Technology and Management, Department of Mechanical Engineering, Linköping University, Linköping, Sweden

Cronemyr, P., Öhrwall Rönnbäck, A. and Eppinger, S. D., 1999, Process improvement simulations using the work transformation model. Proceedings of ICED99, 12th International Conference on Engineering Design, 24–26 August, Munich, Germany.

Eppinger, S. D., Whitney, D. E., Smith, R. P. and Gebala, D. A., 1994, A model-based method for organizing tasks in product development, *Research in Engineering Design*, **6(1)**, 1–13.

Eppinger, S. D., Nukala, M. V. and Whitney, D. E., 1997, Generalized models of design iteration using signal flow graphs, *Research in Engineering Design*, **9(2)**, 112–123.

Helper, S., 1996, Incentives for supplier participation in product development. In *Managing Product Development*, edited by T. Nishiguchi (New York: Oxford University Press).

Hillier, F. S. and Liebermann, G. J., 1986, *Introduction to Operations Research*, 4th edition (Oakland, CA: Holden-Day).

HOLLY, M., COYLE, J. and KOSHIBA, D. A., 2000, The virtual product environment. Proceedings of the Global Product Devlopment, 6th Conference on Product Models, 6–7 November, Linköping, Sweden.
LINNARSSON, A. and MOLIN, K.-Å., 1998, Prerequisites and enabling factors for successful buyer–supplier partnership. Master's Thesis 1998: 23, Department of Management and Economics, Logistics and Transport Systems, Linköping University, Sweden.
MACCORMACK, A., 1998, Towards a contingent model of product development: a comparative study of development practices. Proceedings of the 5th International Product Development Management Conference, 25–26 May, Como, Italy.
MCCORD, K. R. and EPPINGER, S. D., 1993, Managing the integration problem in concurrent engineering. MIT Sloan School of Management, Cambridge, MA, Working Paper, No. 3594.
*MICROTURBO*, 1998, Diagramme type d'initialisation d'un programme (P.A.Q. 5–8-A-51, Indice A), and Etapes et jalonnement du devéloppement produit (P.A.Q. 5–8-A-55, I.A.Q. 0–1-D-53).
ÖHRWALL RÖNNBÄCK, A., 1999, Collaborative product development and IT communication infrastructures — a study in the Swedish aerospace industry. Dissertation No. 33, Licentiate Thesis, International Graduate School of Management and Industrial Engineering, Linköping University, Sweden.
PHADKE, M., 1989, *Quality Engineering Using Robust Design* (Englewood Cliffs, NJ: Prentice Hall).
PIMMLER, T. U. and EPPINGER, S. D., 1994, Integration analysis of product decompositions. Procedings of the ASME Conference on Design Theory and Methodology, Minneapolis, MN, pp. 343–351.
*SAAB*, 1998, Product development process, Draft 98.09.21, 21 September.
SMITH, R. P. and EPPINGER, S. D., 1997, Identifying controlling features of engineering design iteration, *Management Science*, **43(3)**, 276–293.
SMITH, R. P. and EPPINGER, S. D., 1998, Deciding between sequential and concurrent tasks in engineering design, *Concurrent Engineering: Research and Applications*, **6(1)**, 15–25.
SMITH, P. G. and REINERTSEN, D. G., 1998, *Developing Products in Half the Time*, 2nd edition (New York: John Wiley & Sons).
STEWARD, D. V., 1981, The design structure system: a method for managing the design of complex systems, *IEEE Transactions of Engineering Management*, **EM-28(3)**, 71–74.
*SUNDSTRAND*, 1993, MIL-STD-881B, Military Standard. Work Breakdown Structures for Defense Material Items, AMSC F6914.
ULRICH, K. T., and EPPINGER, S. D., 1995, *Product Design and Development* (New York: McGraw-Hill).
WHEELWRIGHT, S. C. and CLARK, K. B., 1992, *Revolutionizing Product Development. Quantum Leaps in Speed, Efficiency, and Quality* (New York, NY: The Free Press).