Heft 134

R. Heib, M. Daneva, A.-W. Scheer

Benchmarking as a Controlling Tool in Information Management

Oktober 1996

Table of Contents

1. Introduction

Information Management (IM) Controlling and benchmarking have been a primary focus on research in the current decade. The motivation for using IM-controlling tools are twofold. On one side, controlling provides coordination for the operative applications. On the other side, it ensures the cost-efficient utilization of information processing (Scheer, 1995). Simultaneously, high attention on benchmarking has become a part of the total quality movement, which swept through the manufacturing industry and spread to the service sector and non-profit organizations.

Given the heightened attention on both IM Controlling and benchmarking, it is surprising that little research has been conducted to examine the linkages between these two concepts in depth and to integrate them properly.

This paper presents how benchmarking fits with IM Controlling. Specifically, we explore the use of benchmarking as a controlling tool with respect to three main concerns: how benchmarking support management of business processes, application systems, and infrastructures.

2. Background

To establish the linkages between the concepts of benchmarking and IM, the following prerequisites are required:

- the creation of a holistic understanding of benchmarking.
- the development of a reference model for benchmarking capable to guide our research on its role in IM.

Some solutions regarding to these issues have been developed due to our previous work on benchmarking (Heib et al, 1996-a, 1996-b). A summary of our findings is presented in the next two subsections.

2.1. Deriving a holistic definition

The development of a holistic view to benchmarking is based on the analysis reported in (Heib and Daneva, 1995) where we considered 42 benchmarking definitions and derived 11 relevant dimensions that characterize the benchmarking process (Tab. 1.):

Dimension	*Instance*
Focus	Process-focused, Tool-focused
Benchmarking Goal	Radical Redesign, Incremental Improvement
Benchmarking Object	Product, Process, Function, Resource, Strategy
Application Context	Marketing, Research and Development, Controlling, Total Quality Management, Strategic Management, Reengineering
Organizational Implementation	Benchmarking Project Team, Routine
Information Source	Primary, Secondary
Benchmarking Network	Multi-Client, Single-Client
Benchmarking Partnership	Friendly, Unfriendly, Anonymous
Cultural Background	American, Japanese, European
Decision Level	Strategic, Tactical, Operational
Benchmarking Scope	Internal, Competitive, Cross-branch

Tab.1. Benchmarking-Dimensionen.

The presented characterization of benchmarking is further used as a basis for formulating a new benchmarking definition. Our motivation behind it was that the definitions available in the reference basically concern particular benchmarking aspects, and do not explicite the relation between company's goals and benchmarking. We propose to use the following definition:

Benchmarking is a business management tool for defining feasible change goals. It is a continual assessment of business objects against the best-in-class ones or a standard, based on measurable characteristics. It is aimed at keeping or regaining company's competitive edge.

2.2. Reference Model for Benchmarking

To ensure the holistic understanding of benchmarking and to provide a mechanism for structuring the information about benchmarking practices, a reference model for benchmarking is developed. It is an universally applicable model that is adaptable to the company's specific goals and describes feasible benchmarking approaches. The model does not focus on a particular benchmarking case, but at structures typical for a set of enterprises that might be classified according to common characteristics (Hars et al, 1992). Therefore, the development of a reference model results from a thoughtful analysis of both theoretical considerations and empirical studies concerning the problem domain. In our work, we account the theoretical analysis given in the previous section, as well as, several empirical benchmarking studies and some US Government and industry guides.

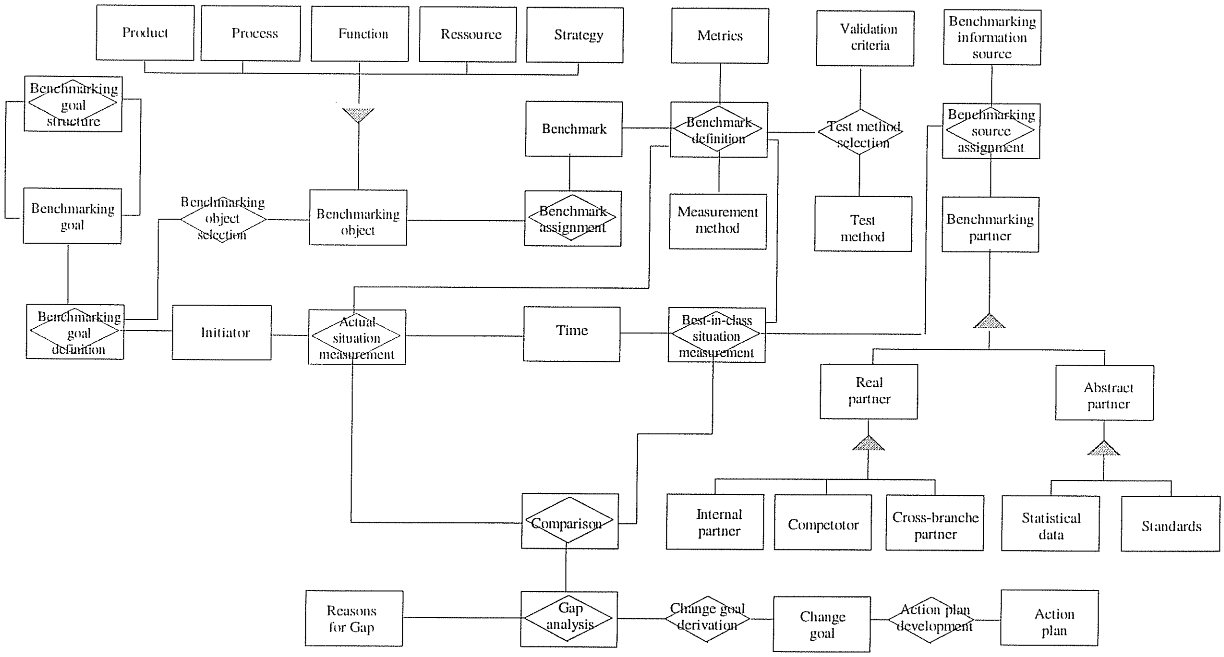

Fig.1. A reference model for benchmarking: the Data View.

To represent the benchmarking process, the ARIS (Architecture of Integrated Information System) methodology for information modelling proposed by (Scheer, 1992), is selected. Generally, the objective of ARIS is to facilitate the specification and implementation of information systems supporting business processes. The ARIS methodology predefines four descriptive views (data, function, organization, and control view) and three levels (requirement definition, design and implementation) components. For each level and each view a set of suitable and integrated description methods is previewed (Scheer, 1995).

We developed our reference model by focusing on the requirements definition level (Heib et al, 1996). The model consists of four description views: data, function, organization and control (process) views. The languages used for enterprise modelling are: extended Entity-Relationship Model, hierarchy diagrams, organizational charts, and extended process chain diagrams, respectively. Fig. 1. depicts the data view of the reference model. The function view and the process view are presented in Appendix 1 and Appendix 2, respectively. For further details with respect to the development of the reference model see the IWi-report 128 (Heib et al, 1996).

3. IM-Controlling through Benchmarking

Controlling coordinates the management of processes by providing decision information to implement efficient and effective processes. To realize that goal controlling provides a set of methods and tools which can be used within the different management subprocesses. The concept of controlling is independent on the type of processes which have to be managed. E.g. controlling concepts for logistics and production management or marketing have been developed.

This paper focuses on information management controlling. The target of IM-controlling is to coordinate the information management processes in order to implement efficient and effective enterprise-wide information processing. Information management consists of the following sub-processes:

- **Business Process Management**
  based on the strategic enterprise targets, business process management defines the structure of the business processes constituting the basis for their implementation by workflow and application systems. it ensures the monitoring and continuos improvement of the business processes.
- **Application Systems Management**
  Application Systems Management ensures the development of individual software systems and the selection of standard software systems. Monitoring the life cycle of the enterprises' application systems ensures the maintenance of software systems, the release management of standard software and the acquisition of new software systems.
- **IT Infrastructure Management**
  IT Infrastructure Management ensures the provision of the resources necessary for enterprise-wide information processing. It includes the organizational units and employees responsible for information processing and their processes like software development. additionally, it comprises the hard- and netware resources and the IS environment like case-tools, programming languages, operating systems and graphical interfaces.

IM-controlling provides different methods and tools to support the subprocesses of information management. This paper focuses on benchmarking as a tool for IM-controlling. To develop a documented and disciplined procedure for studying the potentiality of benchmarking as an IM-controlling tool, we used the holistic benchmarling definition and the reference model. Our objective was to define a systematic and structured research method that could be applied to any of the three IM-levels, i.e. to business process management, application system management and infrastructure management. The procedure includes the following steps:

1. *Systematize benchmarking goals.*
2. *Identify relevant objects to be benchmarked.*
3. *Assess the applicability of the current benchmarking practices in IM-controlling.*
4. *Find out typical illustrative examples for benchmarks.*
5. *Identify potential problems and further research opportunities.*

We followed this procedure with respect to all IM-levels. Our findings are reported in next sections and then summarized in Table 6.

3.1. Benchmarking in Business Process Management

3.1.1. Business Process Benchmarking

To study the interface between benchmarking and business process management, two issues should be considered: business processes and business process models.

Benchmarking Goal

Business process benchmarking is principally concerned with company's efforts to achieve long-term competitive and customer advantages. One way that benchmarking is very useful, is the identification of non-value added enterprise's activities. This leads to target activities for removal or reduction which are adding cost but not value to company's processes. Another way to employ benchmarking as a process controlling tool is to use it in sensitive (what-if) analysis: by conducting benchmarking, we can explore the costs associated with the TO-BE business situation, and thus to determine how much improvement would be gained by the change. Analogously, one can examine a set of alternative TO-BE scenarios and to prioritize them on the basis of cost for improvement actions.

To summarize, benchmarking studies on business processes have two main purposes:

- to state realistic and competition-adequate business process goals.
- to create process vision (Davenport, 1993).

Benchmarking Objects

Process-related benchmarking studies primarily compare business processes as a whole. In case of complex business processes, it is reasonable to divide the process in manageable and logically structured subprocesses. Benchmarking studies on subprocesses can be conducted, then.

State of the Art Bechmarking Practices

Nowadays, both practitioners and academicians agree with the statement that process benchmarking is especially fruitful if it is applied to enterprises from different branches (Mertens et al, 1995, Camp, 1994, Davenprort, 1993). Davenport even suggests that looking for process innovation does not obligatory mean searching for best practice: „even a poorly performed process in a poorly performing company can have innovative aspects“. Thus, it is essential to benchmark distinctive uses of process innovations. Even relative narrow aspects of processes can be worthy of analysis. For example, several companies that have decided to reengineer their order management processes, have studied a division of AT&T whose personnel exploits laptop computers and portable networking to work without offices. Although this detail does not comprise a whole process, it can be considered as an essential element of an order management method.

The current systematic approaches to the development of performance measurement systems for business processes rely on the analysis of critical success factors (Brenner and Keller, 1994, Rolstaadas, 1994). Following Brenner, these factors fall in three groups: strategic, environment and process-specific. The first group involves factors concerning process as a whole, e.g. time to market, process profit, etc. Next, the environment factors concern two process levels: performance and implementation. The process performance level refers to the benefits the process delivers to its customers. In contrast, the implementation level deals with all elements (resources, inputs, outputs) included in the process. Benchmarks referred to the performance level are:

- time: it is a measure of speed with which some benefits are delivered to the customer. For example, bid preparation time (Hirsch, 1994).

- cost: it reflects the question how much a company should spend to deliver a benefit.
- quality: it expresses the compliance of the process with customer's requirements.
- flexibility: it addresses the adaptability of a process to customer needs.

Next, some benchmarks relevant to the implementation level are: capacity, information flow, efficiency, know-how.

Finally, process-specific success factors can be derived from analysis of responsibilities and interests of the participants in the business process („stakeholder analysis").

An alternative approach to design of process benchmarks is proposed by Fried, who develops process performance measures based on the elements of Total Quality Management, Nominated Group Technique, and Systems Thinking (Fried, 1994).

Example of Benchmarks/Metrics

An useful example of process benchmarking is presented in (Hirsch, 1994), where bid preparation processes conducted in three multinational companies (ABB, Krüger Engineering, and Guehring Automation) are evaluated and compared. Significant is the fact that - although the ABB subdivision in Srommen (Norway), produces rolling stock for railways, Krüger Engineering, Copenhagen, is engaged in environmental protection, and Guehring Automation, Frohnstetten (Germany), manufactures grinding machines, the anticipated bid preparation bottlenecks are similar and thereby independent of the product. The enterprise processes were examined regarding the following issues: bid project management, inquiry assessment, product design, cost estimation, product scheduling, sales price fixing, bid document compilation.

The reported case example confirms the hypothesis that business processes of different enterprises can be compared each with other even if the product and/or the branch are different.

Potential Problems

Throughout this section, we have emphasized that without process benchmarking, it is hard to know what to reengineer and what type of improvement to pursue. Successful process benchmarking relies on:

- exact definition of criteria for dividing business processes in subprocesses.
- controlling methods that should enhance the efficiency of the benchmarking process itself (Weber et al, 1994).

Reliable solutions regard to these two problems, can render process benchmarking initiatives in a primary vehicle for IM-controlling.

3.1.2. Process Model Benchmarking

Benchmarking Goal

Comparisons of business process models can serve for many purposes, some of which are enumerated below:

- to ensure model quality in efficient and effective way.
- to validate business process models.
- to control the process of enterprise modelling.
- to anticipate fallacies and pitfalls in integration of submodels and/or customization of reference models.
- to support feedback among the modellers and customers.
- to facilitate cost and schedule estimation with respect to the modelling process.
- to optimize reference models and customer-specific models.

Benchmarking Objects

Examined objects can be either integrated business process models, or submodels, e.g. data models, functional models, etc.

State of the Art Bechmarking Practices

Till now there exist a few references reporting about comparative model analysis (King, 1986, Hull, 1988, Maier, 1994). All of them focus on modelling methodologies (Entity-Relationship Model, Functional Model, and other conceptual modelling frameworks). There exist a few references only that deals with business process models themselves (Hars, 1993, Kesh, 1995). These lead us to the following hypothesis:

- benchmarking of process models provides information needed for conceptualizing business processes.
- the comparison between a reference model and developed process models can be considered as a model validation approach.

Recently, these hypothesis became a basis for workflow analysis and workflow automation. Model benchmarks that characterize the control flow complexity and the utilization of data resources may help the managers evaluate the benefits from workflow management to their processes.

Examples of Benchmarks/Metrics

A typical example of how to use benchmarking information on process models in the context of workflow management is reported by Heilmann (1994). Let us consider three process models evaluated on four benchmarks, as the author suggests (Fig.2). The studied measures are:

- Complexity: it points out whether there are complex and nested structures in the model.
- Detail level: it refers to the amount of functions needed to represent the process completely.
- Degree of personnel participation: it concerns how many workers are involved in the analyzed process.
- Autonomy: it refers to the complexity of the interface between the model under consideration and the other models.

For each benchmark there are three benchmark values: low, middle, and high. The rules how to evaluate benchmarks are given in (Heilnamm, 1994).

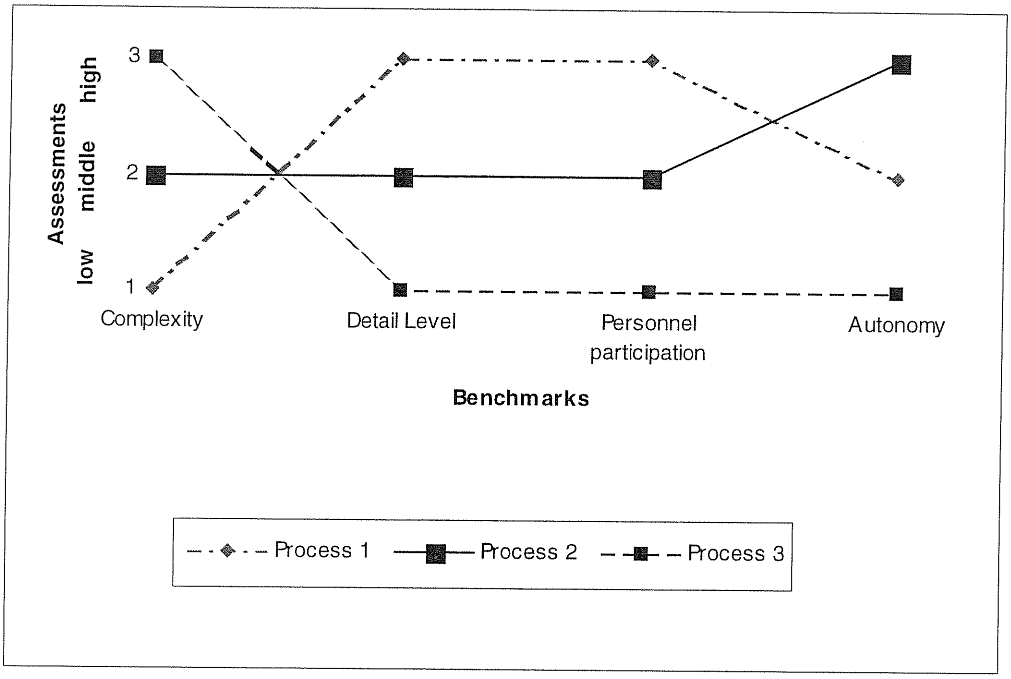

Fig.2. A comparison of three business process models.

The process 1 represents a typical routine workflow, for example order processing. The workflow management fits with such processes the best. Next, the process models 2 and 3 represent more sophisticated flows with limited number of co-workers and reduced interactions with other processes, for example customer consulting. Such processes are difficult to support by workflow management systems.

Another useful example of model benchmarks are the measures for controlling the development of reference models proposed by Hars (1993). These quantify different quality aspects of entity-relationship data models and are practically validated through their use in the CODE ESPRIT Project. The measures are defined as follows:

- Size: it denotes the quantity of elements needed to represent the business process.
- Complexity: it concerns the model structure and the ability of model elements to be interpreted ambiguously.
- Detail level: it refers to the extent to which the model represents the universe of discourse in detail.
- Modularity: it refers to the model's ability to be divided into smaller units.

Potential Problems

Benchmarking of business process models concerns the following problem domains:

- defining and systematizing model benchmarks (measures).
- checking the benchmark set regard to validation criteria.
- developing computer-based tools for supporting model benchmarking.

Our work on the interfaces between the benchmarking and IM takes some steps toward addressing these issues. Recently, we initiated a new project that aims at developing a framework and a tool for benchmarking business process models (Daneva et al, 1996). In contrast to Hars' research, the project deals with the ARIS process view and uses quality measures to control the development of event-driven process chains. For more information about this project see (Daneva et al, 1996).

3.2. Benchmarking in Application System Management

IM-controlling at the level of application management refers to strategic decision making on software life cycle and software development paradigms (Scheer, 1995). The interfaces between benchmarking and IM-controlling at this level are focused on both standard and application software systems.

Benchmarking Goal

Standard and application software system are benchmarked by the following classes of organizations each of which follows a different set of goals (Daneva, 1995-a):

- Software developers: To uncover software bottlenecks and optimize software processes.
- Software system integrators: To highlight strengths of the marketed products.
- BPR-teams: To select the most appropriate software for a given reengineering project.
- Software testing laboratories: To evaluate user satisfaction with the available software.
- Marketing agencies: To determine the market power of software products.
- Commercial computer magazines: To promote or to position a certain software product.

The above goals can be organized in two groups:

- to compare different software packages of a certain type in order to select the most capable of meeting some requirements.
- to compare different releases of one product in order to control the quality enhancement.

Benchmarking Objects

Typically, software benchmarking refers to the whole system. However, as the interest in software reusability has strengthened, the significance of benchmarking software components has increased too. Today, more and more customers commission specialized reviews that deal with software components and particular software product details. For example, a benchmarking study on the systems SAP R/3, Triton, Oracle Financials & Manufacturing, and PeopleSoft may deal with the whole packages as well as the components that support certain business processes.

The software product can not be differentiated from the benefits it provides the users with. Software documentation, service and support, user training programs, product guarantees, and software brand image are treated as complements to any software system. Therefore, each of them can be selected as a benchmarking object.

State of the Art Bechmarking Practices

The problem of software benchmark design and use has been discussed in detail in (Daneva, 1995-a). Generally, during benchmarking of standard software two basic activities should be carried out: assessment and comparison.

Software assessment means to construct an assessment specification and the report of measured values (SCOPE, 1991). The quantifiable software characteristics (benchmarks) are usually represented by a quality hierarchy. Its design and interpretation depend on the underlying quality measure system. Today, the most used quality system is this of McCall (1976), who exploits a hierarchy of software factors. According to him, any complex benchmark number is a sum of weighted subordinate benchmark numbers. An example of hierarchical measure system is given in Fig.4. It depicts Babini's complexity hierarchy (1995), which defines the overall system complexity by considering three factors: product, technology, and market. These are further defined by a set of elementary factors.

The elements at the bottom of the hierarchy are usually assessed by counting how many of a certain quality the product possesses. For example, the SCOPE-Project-team introduced countable software factors that can be used to compute the following characteristics: modularity, generality, portability, redundancy, integrity, complexity, execution and storage efficiency.

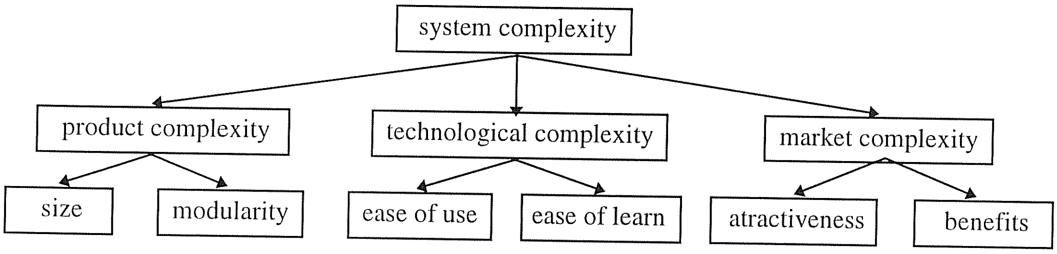

Fig.4. Babini's software complexity tree.

Next, software products are typically compared by using the additive-weight method. This practice is accepted by some U.S. benchmarking organizations such as DATAPRO, VARBUSINESS, National Software Testing Laboratory. Another usable method for software selection is Anderson's algorithm (Anderson, 1989) which uses superiority and inferiority criteria for defining a rank number for any system of a given software class. However, the major drawback to these methods is that they disregard the role of best-in-class product against which the other software systems should be assessed. When these methods are used, the benchmarker need to conduct finer comparisons on a evaluation-by evaluation basis, in order to establish the position of any product regard to the best-in-class-software. This situation is resolved in (Daneva, 1995-a), where an algorithm for software comparing with respect to a best-in-class benchmarking object, is discussed. The algorithm uses measures of similarity in quality between each examined product and the best-in-class one, and ranges software packages according how well they compete with the best-in-class package.

Examples of Benchmarks/Metrics

Typical illustrative examples of software benchmarking studies are the product reviews presented in Computer World, PC Magazine, and Byte. The reviews are based on generally applicable set of the following software attributes: *price, functionality, performance, quality of output, ease of use, ease of learn, quality of documentation*. Some of these characteristics are common for all software types, while others such as functionality highly depend on the analyzed software class. For example, the functionality can be defined for specific user group who is interested in benchmarking study. In such a case, the functionality should cover all software options which the users is supposed to use the most in the system.

Potential Problems

The current problems related to standard software benchmarking can be grouped in three classes concerning the software product model, the quality measure system and the measurement approaches:

- till now there exist no unifying approach to product modelling. Only particular solutions addressing specific software management purposes (certification, assessment) are proposed (SCOPE, 1991).
- the existing quality measure systems for software are based on product issues. With a few exeptions they do not concern the user and his requirements. There exist no sound model of user satisfaction from standard software products.
- current evaluation practices are based on check-lists or expert judgments. This often leads to confusions and misunderstanding about the extent to which a certain product exhibits a quality characteristic. To this end, it will be of potential interest to refine SCOPE's pragmatic measurement approach and to adapt it to benchmarking purposes.

3.3. Benchmarking in IT-Infrastructure Management

IM-controlling in infrastructure management is defined as a process at operative level that includes controlling IS's units (application analysts and programmers), managing repositories, coordinating the communications between internal and external networks (Scheer, 1995). Hence, the relevant dimensions we should study at this IM-level are: software processes, development environments and IS organizations.

3.3.1. Software Process Benchmarking

Benchmarking Goal

Software process benchmarking is considered to be a feasible way to improve software management and raise software quality and productivity (Cunnan, 1995). The main purpose of benchmarking for a software company is to learn about its own technological opportunities by learning about other's similar operation. This lies on the notion of controlling the process of software development and use through benchmarking (Daneva, 1995-b).

Benchmarking Objects

The target objects are the software development process, or specific functions.

State of the Art Bechmarking Practices

There are two basic types of benchmarking practices in software business:

- qualitative assessments that imply the examination of company's methods, tools, and process steps applied in software development and maintenance (Jones, 1996-a), and
- building quantitative figures that compare quantitatively company's data against either branch average numbers or best-in-class achievements.

While the assessments are usually performed by means of on-site interviews and check lists, the comparative fugues are based on measurements on productivity, quality, schedule and cost indicators.

Two widespread assessment models of great importance for software industry are the Capability Maturity Model (CMM) developed by Software Engineering Institute at Carnegie Mellon University (Paulk et al, 1995), and the BOOTSTRAP Model developed by European Software Institute (Kuwaja et al, 1995). Both of them describe the range of expected results that a software organization can achieve by following a software process. The CMM and BOOTSTRAP assessment levels are defined by using questionnaires about the implementation of important software practices in the organization.

Another assessment framework is now being produced from the SPICE-project which is supported by ISO and is aimed at developing an international standard for software process assessment (SPICE, 1993). Organizations will be able to use this standard:

- to determine the capability of software suppliers,
- to define whether company's process is aligned with and supports the organization's business needs, and
- to determine developer's ability to undertake new projects.

More recently, C. Jones from Software Productivity Research (SPR) has advanced another way to look at software assessment (1996-a). It is similar to the CMM concept, but it uses different assessment scoring method, which accounts for the relative position of a company with respect to branch average organizations. Thus, the SPR assigns rank 1 to the excellent end of the scale, while the CMM uses 5 for excellence (Tab. 2)

SPR Assessment Scoring		CMM Assessment Scoring	
Level	*Interpretation*	*Level*	*Interpretation*
1	Excellent	1	Initial
2	Better than average	2	Repeatable
3	Average	3	Defined
4	Below average	4	Repeatable
5	Poor	5	Optimizing

Table. 2. SPR's and CMM's assessment scoring techniques.

The qualitative assessment reports do explain why the companies vary from industry norms, but they provide insufficient information about to what extent a software leader or lagger defer from the average organization. To cope with this problem, project performance indicators have been developed. These can be structured to two levels. At the top level, we consider project productivity and quality benchmarks quantifiable due to data extracted from a sample of company's projects. Such benchmarks can examine a 100 % sample or a subset of projects above a certain minimal size. To ensure comparability between organizations, the data are segmented so as software projects are compared against projects of the same type, for example, military systems, business information systems, engineering applications.

Next, the bottom level of process benchmarking refers to activity-based measurements. These deal with specific kinds of work performed during software development (Jones, 1996). Such measurements are difficult to collect, but are much more valuable for an organization to set feasible improvement goals. The main benefit from these benchmarks is that they provide information which can not be revealed from less granular studies. For example, the production of software documents is known to be a major cost driver. To create a complete picture for the efforts and costs associated with the paper documents one need to study all activities like describing software architecture, developing specifications, producing test plans, writing user manuals, etc.

For designing process indicators, the software engineering community has developed different frameworks. The most of them are submitted by ESPRIT-consortia like AMI, METKIT, MARMAID, MUSiC, PYRAMID. Others are proposed by private consulting companies that offer workbenches to support the extraction of quantified information from project deliverables. Currently, there are more than 60 workbenches for software evaluation and project estimation in the market. Some of the most popular are Checkpoint (SPR), Metricate (SPC), Function Point Workbench (Charismatek Software), Cosmos, etc.

Examples of Benchmarks/Metrics

To illustrate how process benchmarks can be used, we reproduce here some results obtained by SPR when two software processes have been compared. For the first process the estimating, planning, tracking and quality control activities are performed manually, and for the second process - automatically. Two representative samples of projects have been analyzed with respect to how successful the projects are finished. Four possible projects' end states are considered: canceling, delaying, on-time end and early end. For the projects of the first sample, the above mentioned activities have been performed manually, and for the project of the second sample - automatically. The comparative figures are given in Table 3. It is pointed out, that 78% of the projects are finished on time when the estimation, planning, tracking and quality control activities are performed automatically.

Factors	Cancel	Delay	On-time	Early
Manual estimates Manual planning Informal tracking Minimal quality control	40%	45%	15%	0%
Automated estimates Automated planning Formal tracking Optimal quality control	1%	2%	78%	19%

Table 3. Probability of software project success or failure associated with management factors.

Potential Problems

The scope of software benchmarking is very wide: it encompasses all software aspects that influence the development of well engineered product at low cost and within time schedule. It is also a very complex problem domain because of its relative youth, of its multidisciplinary nature, and because of the great diversity, variability, and complexity of participants in software development process (managers, engineers, users). Some high potential area of interest in software benchmarking are:

- refining the existing formalisms for software process modelling. Current models aim to express what steps a process consists of, or how they are to be accomplished. In order to reengineer a process, we often need to have a deeper conceptualizing about the process - an understanding that reveals the whys behind the whats and the hows (Milopoulus, 1994).
- development of reference models for software business, which should serve as standards in process comparing.
- systematizing software process metrics on the base of classification criteria.
- development of reliable measurement methods for software processes, and procedures to collect data to analyze the process execution against process objectives.
- development of methods for analyzing process models with respect to its internal consistency, completeness, and correctness.

3.3.2. Hardware Benchmarking

Benchmarking Goal

Although hardware systems are benchmarked for a variety of reasons, the reason most commonly cited is to identify the right computer platform that will solve their problem properly at a lowest price. This motivates the information technology experts to develop benchmark tests for two purposes:

- to compare different computer systems: the benchmark tests should provide performance evaluation of different systems on different platforms running the same application, for example SQL on an IBM AS/400 versus Informix on HP 9000
- to compare different machines in a computer family: the benchmarking study should reveal how computer systems are ranked within a computer family.

Benchmarking Objects

Typical objects under consideration are computer platforms. These can vary in size, peripherals attached, number of users, and scalability.

State of the Art Bechmarking Practices

General guidelines about how to conduct computer benchmarking are provided by several world known consortia: Transaction Processing Council (TPC), System Performance Evaluation Corporation (SPEC), Business Application Performance Council (Gray, 1993). They provide benchmarks that measure and record the performance of a system, as well.

Many benchmarking professionals (for example bechmarkers from DEC, HP, and IBM) conduct studies by following TCP's guidelines. According to them, the main objective of any benchmarking exercise is the production of a document named Full Disclosure Report which meets the requirements of some benchmark specifications. Next, the TPC recommends to build a benchmarking team by including two types of personnel: performance experts that process repetitive benchmark executions, and produce the performance results, and audit specialists who establish exact workplace for executing all procedures specified in the audit section of a certain benchmark.

Lately, the workstation vendors designed numerous benchmarks typical for intensive numeric computations. The most successful among them are the SPEC's CINT92 and CFT92 suits which include 20 different metrics. The reason behind the design of all these metrics lies in SPEC's assumption that no single number can characterize workstation performance. SPEC has also developed a UNIX System Development Multitasking (SDM) benchmark that studies UNIX users editing, searching, printing, compiling and building software systems. Currently, the SPEC experts are working on the LADIS file server benchmark for UNIX systems.

Examples of Benchmarks/Metrics

Computer benchmarks are classified along two dimensions (Crawford, 1994):

- Generic/ Domain specific.

  Generic benchmarks are often used as a rough estimate of the relative system performance and price/performance of a system. The performance is typically a throughput metric (work/second), while the price is the cost over some period of ownership metric. Together they give a price/ performance ratio. They are usually supplied by vendors with the announce of new products. All such performance characteristics are approximate and does not consider specific application peculiarities. In contrast, the domain-specific benchmarks try to model the computer workload for specific applications for a given problem domain (e.g. engineering applications CAD/CAM, CASE, etc.).

- Standard/ Custom.
  Currently, the TPC-A and TPC-B are the only generally recognized benchmarks that check different aspects of database performance. Additional tests (TPC-C, TPC-D) to fill the breach left by TPC-A and TPC-B, are now under development. Next, custom benchmarks model the functionality of a real commercial system by using a subset of the user application. The strength of the standard benchmarks is that they have a fixed methodology, and thus, they produce comparable results with a known workload. However, the workload does not usually match exactly the customer environment. In such cases, custom benchmarks may has a significant advantage over the standard ones in terms of applicability to specific environments.

Potential Problems

There are two basic problems in computer benchmarking:

- how to develop a common understanding (agreements) between the benchmarkers and the customers (Crawford, 1994). In a lot of cases the benchmarkers' interpretation of the benchmark is different than this of the people who commission the benchmarking study (people actually interested in the benchmarking results).
- how to enable an uniform implementation of the benchmark tests. Although the most of benchmarks (all TCP) contain „Audit Checklist“, the requirements presented in these sections are not sufficient for ensuring an unifying implementation approach. This problem is partially resolved by using supplemental checklists that helps benchmarkers interpreting audit requirements unambiguously, and determining procedures for proving the correctness of the benchmark implementation.

3.3.3. Benchmarking Development Environments

Benchmarking Goals

Benchmarking practices are used to gain more insight into the development environment IS-organizations are operating in.

Benchmarking Objects

Typical objects for benchmarking are:

- CASE-tools
- Programming languages
- Operation systems
- Graphical Interfaces

State of the Art Bechmarking Practices

Since the above objects can be considered as software products, the benchmarking approaches used for standard software systems are suitable for studies on company's development environment, as well. However, in the '90s, a new approach to benchmarking CASE-tools, programming languages, and operation systems has emerged. This new evaluation methodology uses intensively the function points metric (Jones, 1996-a) to quantify two important features of the studies objects:

- the completeness provided by the development environment, and
- the organization's productivity that occur from using this environment.

Function point analysis is first made public by A.Albrecht of IBM in 1979. The function point measure refers directly to the business requirements which a software system should meet. Therefore it could be used successfully for benchmarking any particular aspect of company's development environment. Two recent studies on CASE-tools and programming languages are reported by SPR (Jones, 1996-b, 1996-c). Both of them report productivity ranges and averages collected by SPR due to analysis of more than 1000 software projects.

Examples of Benchmarks/Metrics

In this section we review briefly benchmarking issues with respect to some of the above given target objects. These are summarized in Table 4, where we report some of the most cited benchmarking goals together with the benchmarks relevant to them:

Benchmarking object	Benchmarking goal	Benchmarks
CASE-tools	To find out the most appropriate CASE-tool for building a Quality Management System	Project Management Specification Implementation Configuration Management Quality Measurement
Programming languages	To compare potentials of different programming paradigms	Abstraction Handling Capseling Dynamic Interaction Static Relation Building Polymorphism Modularity
Operation systems	To establish the optimal number of concurrently working users	Price/Performance Workload Parameters Concurrency Control Transaction Models Supported.

Table 4. Some benchmarking goals and benchmarks for IS environment.

Potential Problems

Basic problem in benchmarking company's development environment is how to derive a benchmark system specific for a certain IS-organization by using some standards, for example, ISO 9000, CMM, Baldrige National Quality Award.

3.3.4. Benchmarking IS-Organizations

Benchmarking Goals

Much of benchmarking research on IS-organizations concerns two indicators: the organization's efficiency and the effectiveness. Fundamentally, these benchmarking studies have focused on the following goals:

- to find out the best way of using the IS in the organization so that to optimize IS-benefits.
- to establish the best programs for promoting cooperation and communication within the IS-organization.
- to establish the best workable solution to combine information from different separate sources.

Benchmarking Objects

Typical objects for benchmarking are organizational units ranging from a single worker, project team, department, enterprise, to the corporation as a whole.

State of the Art Bechmarking Practices

Today's commonly accepted practices to evaluate the effectiveness and the efficiency of IS-organizations are based mainly on budgetary performance measures referring to stocks, throughput time, costs, investments, etc. The major drawback to all these approaches is that they have too little concern with employee-related qualities of work, responsibilities and innovative strengths. An alternative method for assessment of organizational units and creating measurement figures is proposed by Mertins et al. (1995) who organizes the benchmarks along to five dimensions: times, costs, attractive work places, quality, innovative potential. This allow us to benchmark organizations by regarding not only the organizations themselves, but all functions performed by the organization which contribute to the value adding process. Mertins' evaluation methodology employs the collection of objective data as well as making subjective judgments.

Other group of modern measurement methods are based on „stakeholder analysis" which involves the identification of stakeholder segments and the derivation of benchmarks that corresponds to the interests, „claims" and objectives of each group. For example, in evaluating company's software division, the stakeholder groups can be: project management, quality auditors, programmers, designers. Each stakeholder is assumed to express a consistent interest within the organization, but the group's interest might differ from one holder to another. A typical example of stakeholder analysis methods is Kaplan's „balanced scorecard" approach. It reviews the enterprise from financial, customer, innovation and internal business viewpoint and has the advantage to provide cross-company comparisons.

However, according to all stakeholder analysis methods, the most admired organizational units are these that have succeeded in balancing the competing claims of stakeholder segments. This measure of success expresses the performance of a company.

Concerning the comparison, both academicians and practitioners consider that a single productivity measure does not provide a comprehensive picture of organization's performance. In contrast, they propose to construct efficiency and effectiveness profiles reveal the interrelations and synergic effects of the particular measures.

Examples of Benchmarks/Metrics

All above discussed approaches employ hierarchical benchmark systems. The basic high level benchmarks are: *staff satisfaction, user satisfaction, budget performance, information availability and strategic congruence*. These are broken down to evaluation factors and metrics. Thus, for example, staff satisfaction, according to Scudder (1991), is made up by *job satisfactory index, training level, turnover ratio*. Some other examples for measures relevant to IS-organizations are given in Table 5.

Financial measures	Personnel measures	Marketing measures
Depreciation Capitalization of costs Provisions Added value	Job creation Employee productivity Development expenditure per employee	Sales volume Market share statistics Value for money Product Price/Performance

Table 5. Some organization benchmarks.

Potential Problems

Recent annual corporation performance reports of some large well-known companies like BCCI, Polly Peck, Trafalgar House, Blue Arrow, and Maxwell Communications, all turned out to be deliberately misleading (Brown, 1995). They do not exhibit consistency in evaluation methods making the interpretation and cross-comparison difficult. Thus, reliable characterizing the organizational units is becoming a surprisingly subtle and complex problem. Furthermore, one should examine other key issues on:

- how to combine the partial measures to paint a „true“ picture of the organization. The calibration methods should reflect the unpredictability of company's environment. Current measurement methods seem to be remarkably facile and even inappropriate (Brown, 1995).
- how to develop „industry standards“ as well as „standards for well managed“ firms to which a company has to be compared.
- how to define multiple levels of IS-performance. This means to identify all organization aspects that constitute the IS-effectiveness and efficiency, and to assign these aspects to certain organization levels.
- what are sources of reliable performance information for IS-organizations..

IM Subprocesses	*Benchmarking Goals*	*Benchmarking Objects*	*State of the Art*	*Examples of Benchmarks*
Business Process Management	• to state business process goals. • to create process vision.	**Business Processes Subprocesses**	• Fried's generic guidelines to process measure design • Brenner's process measure classification. • Cross-branch benchmarking approaches	Time Cost Quality
	• to ensure efficiently model quality, to validate process models, to control the process of enterprise modelling. • to anticipate pitfalls in integration of submodels and/or customization of reference models. • to optimize reference and customer-specific models, and to facilitate cost estimation for modelling.	**Process Models**	• Hars' measure set for reference models. • Workflow-relevant mesures.	Complexity Detail level Size Modularity
Application System Management	• to compare software of a certain type in order to select the most capable of meeting some requirements. • to compare different releases of one product in order to control the quality enhancement.	**Application and Standard Software**	• Assessment methods: SCOPE-methodology, hierarchical quality models (McCall) • Comparison methods: Additive-weight method, Anderson's approach, QR algorithm.	Ease of Use Ease of Learn Reliability Vendor Support
IT-Infrastructure Management	• to learn about company's own technological opportunities by learning about other's similar operation.	**Software Processes**	• Standards: ISO 9000, Capability Maturity Model, BOOTSTRAP Method. • Approaches proposed by ESPRIT-metric projects: AMI, METKIT, MARMAID, MUSiC, PYRAMID. • Workbenches: COSMOS, AMI, Metricate, Checkpoint.	Organization Maturity Technology Maturity Methodology Maturity
	• to compare different systems. • to compare different machines in a computer family.	**Hardware Systems**	• Methodologies proposed by SPEC, TPC, Business Application Performance Council	Price/performance Throughput
	• to gain some recognized quality appraisal. • to gain insight into company's development environment.	**Development Environments**	• Standards: ISO 9000, DIN Norms regard to Software Development. • Function point analysis.	Usability in Life Cycle Software Management Completeness
	• to optimize IS-benefits for the organizations. • to establish the right workable solution to combine information from different separate sources. • to establish the best programs for promoting cooperation within the IS organization.	**IS-organizations**	• Traditional business management evaluation. • Mertins' approach to organizational benchmarking. • „Stakeholder analysis", Kaplan's „balanced scorecard". and „informed spectator" method.	Customer Satisfaction Efficiency Job Satisfaction Financial Performance

Tab.6. Benchmarking as an IM-controlling tool: a summary.

4. Conclusions

Benchmarking studies relevant to IM levels are growing in terms of number conducted each year. Nevertheless, systematic approaches to examine the benefits from benchmarking as an IM-controlling tools have not been developed yet. The main contribution of our work is that we have established the interfaces between benchmarking and the three IM-levels. For each of them we have formulated specific benchmarking goals, relevant objects to be studied and a set of quantifiable indicators. The applicability of the state-of-the-art benchmarking practices has been examined, what lead us to identify potential problems and new research opportunities.

REFERENCE

1. Albrecht, A., Measuring Application Development Productivity; SHARE/GUIDE IBM Application Development Symposium, Monterey, California, Oct., 1979, pp. 83-92.
2. Anderson, E. , Heuristics for Software Evaluation and Selection, Software Practice and Experience, 8, 1989, pp. 707-717
3. Brenner, W., G. Keller, Business Reengineering mit Stamdard-Spftware, Campus Verlag, Frankfurt/Main, 1995.
4. Brown, M., S. Laverick, Measuring Corporate Performance, Longe range Planning, 27(4), 1994, pp 89-98.
5. Camp, R., Benchmarking, Carl Hansel Verlag, München, 1994.
6. Crawford, J., TCP Auditing: How to Do It Better, Qualerly Report, 1994, pp. 9-11.
7. Daneva, M., Software Benchmark Design and Use, In: Reengineering the Enterprise, Ed.: J.Brown, Chapman & Hall, London, 1995-a, pp. 20-29.
8. Grey, J., Benchmarking Handbook for Database and Transaction Processing Systems, Morgan Kaufman Publ.,Califormia, 1993.
9. Daneva, M., Benchmarking Practice for Software Quality Achievement, Proc. of 5th. Int. Conf. on Software Quality, Austin, Texas, Oct. 22-26, 1995-b, pp. 443-457.
10. Daneva, M., R.Heib, A.-W. Scheer, Benchmarking Business Process Models, Artikel in Vorbereitung.
11. Davenport, T.H., Process Innovation, Harvard Business School Press, Boston, 1993.
12. Fries, S. M.Saghezzi, Meßgröße für Geschäftsprozesse, Controlling, Heft 6, November/Dezember, 1994, S. 338-345.
13. Hars, A. Referenzmodelle, Gabler, Wiesbaden, 1993.
14. Hars, A., C. Kruse, R. Heib, A.-W. Scheer, Ways of Utilizing Reference Models for Data Engineering, Proceedings on CIM Conference FMEA'92.
15. Heib, R., M. Daneva, Benchmarking - eine Definitionsklärung, Management & Computer, 3.Jg, H.4, 1995, S. 227-230.
16. Heib, R., M. Daneva, A.-W. Scheer, ARIS-based Reference Model for Benchmarking, Veröffentlichungen des Instituts für Wirtschaftsinformatik, Heft 128, Saarbrücken, April, 1996-a.

17. Heib, R., M. Daneva, A.-W. Scheer, Benchmarking as a Controlling Tool in Information Management, IFIP WG 5.7 Workshop on Enterprise Modelling Techniques and Benchmarking, Bordeaux, 1996-b. To be published.

18. Heilmann, Workflow Management, HDM, 176/1994, S. 8-21.

19. Hirsch, B., D. Thoben, M. Krömker, A. Wickner, Benchmarking of Bid Preparation for Capital Goods, in Benchmarking - Theory and Practice, Ed.: A. Rolstaadas, Chapman & Hall, London, 1994.

20. Hull, R., R. King, Semantic Database Modelling: Survey, Applications, and Research Issues, ACM Computing Surveys, 19(3), September 1987, pp. 201-260.

21. Jones, C., Applied Software Measurement, McGraw Hill, 1996-a.

22. Jones, C., Patterns of Software System Failure and Success, International Thomson Computer Press, 1996-b.

23. Jones, C., Function Points Metrics to CASE-tools Assessment, American Programer, June, 1996-c.

24. Kesh, Quality Evaluation of Entity-Relationship Models, IEEE Transactions on Software Engineering, Vol.21, No 11, Nov, 1995, pp. 848-858.

25. Kuvaja, P., J. Similä, L. Krzanik, A. Bicego, S. Saukkonen, G. Koch: Software Process Assessment and Improvement - the BOOTSTRAP Approach, Blackwell Publishers, Oxford/UK and Cambridge, MA, USA, 1995.

26. Maier, R. Quality of Data Models, in Proceesings of Third International Conference on Software Quality Management, Seville, 1995.

27. McCabe, Th., A Complexity Measure, IEEE Transactions on Software Engineering, SE-2(4), Dec. 1976, pp.308-320.

28. Mertins, K., S. Kempf, G. Siebert, How Benchmarking Supports Reengineering, in Reengineering the Enterprise, Ed.: J. Brown, Chapman & Hall, London, 1995, pp. 72-82.

29. Paulk, M., B. Curtis, et al: The Capability Maturity Model, Addison Wesley, Reading, MA, 1995.

30. Rolstaadas A., Benchmarking - Theory and Practice, Chapman & Hall, London, 1994.

31. Scheer, A.-W., Architecture of Integrated Information Systems, Springer-Verlag, Berlin et al, 1992.

32. Scheer, A.-W., Wirtschaftsinformatik, Referenzmodelle für industrielle Geschäftsprozesse, 6. Auflage, Springer-Verlag, Berlin et al, 1995.

33. SCOPE ESPRIT Project 2151: „SCOPE - Technology for Evaluation and Certification of Software Product Quality“, Project Brochure, November, 1992.

34. Scudder, R.A., A.R. Kucie, Productivity Measures for Information Systems, Information & Management, 20, 1991, pp. 343-354.

35. SPICE: Software Process Improvement and Capability dEtermination (SPICE), Software Quality Journal, Ed.: Dorling, Vol. 2, No 4, December 1993, pp. 209-224.

36. Weber, J. H. Goedel, M. Hamprecht, Benchmarking als Instrument zur Gestaltung von Prozessen im Führungssystem, Management & Computer, 3.Jg, H.3, 1995, S. 115-121.

Die Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi) im Institut für empirische Wirtschaftsforschung an der Universität des Saarlandes erscheinen in unregelmäßiger Folge.

Heft 134: R. Heib, M. Daneva, A,-W.Scheer: Benchmarking as a Controlling Tool in Information Management, Oktober 1996

Heft 133: A.-W. Scheer: ARIS-House of Business Engineering, September 1996

Heft 132: J. Sander, A.-W. Scheer: Multimedia Engineering: Rahmenkonzept zum interdisziplinären Management von Multimedia-Projekten, Juli 1996

Heft 131: R. Heib, M. Daneva, A.-W. Scheer: ARIS-based Reference Model for Benchmarking, April 1996

Heft 130: R. Chen, V. Zimmermann, A.-W. Scheer: Geschäftsprozesse und integrierte Informationssysteme im Krankenhaus, April 1996

Heft 129: M. Nüttgens, V. Zimmermann, A.-W. Scheer: Business Process Reengineering in der Verwaltung, April 1996

Heft 128: P. Hirschmann, P. Lubiewski, A.-W. Scheer: Management von Konzernprozessen - Eine Fallstudie -, März 1996

Heft 127: J. Galler, M. Remme, A.-W. Scheer: Der Inseltrainer - Ein multimediales Lernsystem zur Qualifizierung in Planungsinseln, Januar 1996

Heft 126: P. Loos, O. Krier, P. Schimmel, A.-W. Scheer: WWW-gestützte überbetriebliche Logistik - Konzeption des Prototyps WODAN zur unternehmensübergreifenden Kopplung von Beschaffungs- und Vertriebssystemen, Februar 1996

Heft 125: M. Remme, A.-W. Scheer: Konstruktion von Prozeßmodellen, Februar 1996

Heft 124: M. Bold, E. Landwehr, A.-W. Scheer: Die Informations- und Kommunikationstechnologie als Enabler einer effizienten Verwaltungsorganisation, Februar 1996

Heft 123: P. Loos: Workflow und industrielle Produktionsprozesse - Ansätze zur Integration, Januar 1996

Heft 122: A.-W. Scheer: Industrialisierung der Dienstleistungen, Januar 1996

Heft 121: J. Galler: Metamodelle des Workflow-Managements, Dezember 1995

Heft 120: C. Kocian, F. Milius, M. Nüttgens, J. Sander, A.-W. Scheer: Kooperationsmodelle für vernetzte KMU-Strukturen, November 1995

Heft 119: W. Hoffmann, A.-W. Scheer, C. Hanebeck: Geschäftsprozeßmanagement in virtuellen Unternehmen, Oktober 1995

Heft 118: M. Remme, J. Galler, O. Gierhake, A.-W. Scheer: Die Erfassung der aktuellen Unternehmensprozesse als erste operative Phase für deren Re-engineering -Erfahrungsbericht-, September 1995

Heft 117: J. Galler, A.-W. Scheer, S. Peter: Workflow-Projekte: Erfahrungen aus Fallstudien und Vorgehensmodell, August 1995

Heft 116: A. Gücker, W. Hoffmann, M. Möbus, J. Moro, C. Troll: Objektorientierte Modellierung eines Qualitätsinformations-systems, Juni 1995

Heft 115: Th. Allweyer: Modellierung und Gestaltung adaptiver Geschäftsprozesse, Mai 1995

Heft 114: W. Hoffmann, A.-W. Scheer, M. Hoffmann: Überführung strukturierter Modellierungsmethoden in die Object Modeling Technique (OMT), März 1995

Heft 113: P. Hirschmann, A.-W. Scheer: Konzeption einer DV-Unterstützung für das überbetriebliche Prozeßmanagement, November 1994

Heft 112: A.-W. Scheer, M. Nüttgens, A. Graf v. d. Schulenburg: Informationsmanagement in deutschen Großunternehmen - Eine empirische Erhebung zu Entwicklungsstand und -tendenzen, November 1994

Heft 111: A.-W. Scheer: ARIS-Toolset: Die Geburt eines Softwareproduktes, Oktober 1994

Heft 110: M. Remme, A.-W. Scheer: Konzeption eines leistungsketteninduzierten Informationssystemmanagements, September 1994

Heft 109: Th. Allweyer, P. Loos, A.-W. Scheer: An Empirical Study on Scheduling in the Process Industries, July 1994

Heft 108: J. Galler, A.-W. Scheer: Workflow-Management: Die ARIS-Architektur als Basis eines multimedialen Workflow-Systems, Mai 1994

Heft 107: R. Chen, A.-W. Scheer: Modellierung von Prozeßketten mittels Petri-Netz-Theorie, Februar 1994

Heft 106: W. Hoffmann; R. Wein; A.-W. Scheer: Konzeption eines Steuerungsmodells für Informationssysteme - Basis für die Real-Time-Erweiterung der EPK (rEPK), Dezember 1993

Heft 105: A. Hars; V. Zimmermann; A.-W. Scheer: Entwicklungslinien für die computergestützte Modellierung von Aufbau- und Ablauforganisation, Dezember 1993

Heft 104: A. Traut; T. Geib; A.-W. Scheer: Sichtgeführter Montagevorgang - Planung, Realisierung, Prozeßmodell, Juni 1993

Heft 103: wird noch nicht verlegt

Heft 102: P. Loos: Konzeption einer graphischen Rezeptverwaltung und deren Integration in eine CIP-Umgebung - Teil 1, Juni 1993

Heft 101: W. Hoffmann, J. Kirsch, A.-W. Scheer: Modellierung mit Ereignisgesteuerten Prozeßketten (Methodenbuch, Stand: Dezember 1992), Januar 1993

Heft 100: P. Loos: Representation of Data Structures Using the Entity Relationship Model and the Transformation in Relational Databases, January 1993

Heft 99: H. Heß: Gestaltungsrichtlinien zur objektorientierten Modellierung, Dezember 1992

Heft 98: R. Heib: Konzeption für ein computergestütztes IS-Controlling, Dezember 1992

Heft 97: Chr. Kruse, M. Gregor: Integrierte Simulationsmodellierung in der Fertigungssteuerung am Beispiel des CIM-TTZ Saarbrücken, Dezember 1992

Heft 96: P. Loos: Die Semantik eines erweiterten Entity-Relationship-Modells und die Überführung in SQL-Datenbanken, November 1992

Heft 95: R. Backes, W. Hoffmann, A.-W. Scheer: Konzeption eines Ereignisklassifikationssystems in Prozeßketten, November 1992

Heft 94: Chr. Kruse, A.-W. Scheer: Modellierung und Analyse dynamischen Systemverhaltens, Oktober 1992

Heft 93: M. Nüttgens, A.-W. Scheer, M. Schwab: Integrierte Entsorgungssicherung als Bestandteil des betrieblichen Informations-managements, August 1992

Heft 92: A. Hars, R. Heib, Chr. Kruse, J. Michely, A.-W. Scheer: Approach to classification for information engineering - methodology and tool specification, August 1992

Heft 91: C. Berkau: Konzept eines controllingbasierten Prozeßmanagers als intelligentes Multi-Agent-System, Januar 1992

Heft 90: C. Berkau, A.-W. Scheer: VOKAL (System zur Vorgangskettendarstellung), Teil 2: VKD-Modellierung mit Vokal, Dezember 1991 (wird nicht verlegt)

Heft 89: G. Keller, M. Nüttgens, A.-W. Scheer: Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)", Januar 1992

Heft 88: W. Hoffmann, B. Maldener, M. Nüttgens, A.-W. Scheer: Das Integrationskonzept am CIM-TTZ Saarbrücken (Teil 2: Produktionssteuerung), Januar 1992

Heft 87: M. Nüttgens, G. Keller, S. Stehle: Konzeption hyperbasierter Informationssysteme, Dezember 1991

Heft 86: A.-W. Scheer: Koordinierte Planungsinseln: Ein neuer Lösungsansatz für die Produktionsplanung, November 1991

Heft 85: W. Hoffmann, M. Nüttgens, A.-W. Scheer, St. Scholz: Das Integrationskonzept am CIM-TTZ Saarbrücken (Teil 1: Produktionsplanung), Oktober 1991

Heft 84: A. Hars, R. Heib, Ch. Kruse, J. Michely, A.-W. Scheer: Concepts of Current Data Modelling Methodologies - A Survey - 1991

Heft 83: A. Hars, R. Heib, Ch. Kruse, J. Michely, A.-W. Scheer: Concepts of Current Data Modelling Methodologies - Theoretical Foundations - 1991

Heft 82: C. Berkau: VOKAL (System zur Vorgangskettendarstellung und -analyse), Teil 1: Struktur der Modellierungsmethode - Dezember 1991 (wird nicht verlegt)

Heft 81: A.-W. Scheer: Papierlose Beratung - Werkzeugunterstützung bei der DV-Beratung, August 1991

Heft 80: G. Keller, J. Kirsch, M. Nüttgens, A.-W. Scheer: Informationsmodellierung in der Fertigungssteuerung, August 1991

Heft 79: A.-W. Scheer: Konsequenzen für die Betriebswirtschaftslehre aus der Entwicklung der Informations- und Kommunikationstechnologien, Mai 1991

Heft 78: H. Heß: Vergleich von Methoden zum objektorientierten Design von Softwaresystemen, August 1991

Heft 77: W. Kraemer: Ausgewählte Aspekte zum Stand der EDV-Unterstützung für das Kostenmanagement: Modellierung benutzerindividueller Auswertungssichten in einem wissensbasierten Controlling-Leitstand, Mai 1991

Heft 76: Ch. Houy, J. Klein: Die Vernetzungsstrategie des Instituts für Wirtschaftsinformatik - Migration vom PC-Netzwerk zum Wide Area Network (noch nicht veröffentlicht)

Heft 75: M. Nüttgens, St. Eichacker, A.-W. Scheer: CIM-Qualifizierungskonzept für Klein- und Mittelunternehmen (KMU), Januar 1991

Heft 74: R. Bartels, A.-W. Scheer: Ein Gruppenkonzept zur CIM-Einführung, Januar 1991

Heft 73: A.-W. Scheer, M. Bock, R. Bock: Expertensystem zur konstruktionsbegleitenden Kalkulation, November 1990

Heft 72: M. Zell: Datenmanagement simulationsgestützter Entscheidungsprozesse am Beispiel der Fertigungssteuerung, November 1990

Heft 71: D. Aue, M. Baresch, G. Keller: URMEL, Ein UnteRnehmensModELlierungsansatz, Oktober 1990

Heft 70: St. Spang, K. Ibach: Zum Entwicklungsstand von Marketing-Informationssystemen in der Bundesrepublik Deutschland, September 1990

Heft 69: A.-W. Scheer, R. Bartels, G. Keller: Konzeption zur personalorientierten CIM-Einführung, April 1990

Heft 34: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS - Ein Ansatz zur Entwicklung prüfungsgerechter Software-Systeme, Mai 1982

Heft 33: A.-W. Scheer: Disposition- und Bestellwesen als Baustein zu integrierten Warenwirtschaftssystemen, März 1982, Vortrag anläßlich des gdi-Seminars "Integrierte Warenwirtschafts-Systeme" in Zürich vom 10. - 12. Dezember 1981

Heft 32: A.-W. Scheer: Einfluß neuer Informationstechnologien auf Methoden und Konzepte der Unternehmensplanung, März 1982, Vortrag anläßlich des Anwendergespräches "Unternehmensplanung und Steuerung in den 80er Jahren in Hamburg vom 24. - 25.11.1981

Die Hefte 1 - 31 werden nicht mehr verlegt.