

Heft 136

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

Benchmarking Business Process Models

Oktober 1996

Table of Contents

	Page
1. Introduction	1
2. The PROBE-project	2
2.1. Project objectives	2
2.2. Project strategy	2
2.3. The PROBE products	3
3. The PROBE solution	5
3.1. Background	5
3.2. The definition of quality in the PROBE-project	8
3.3. Designing a benchmark system	8
3.4. Quality issues suitable for benchmarking	9
3.5. The PROBE benchmark suite	10
3.5.1. An illustrative example	11
3.5.2. The Function Cohesion Indicator	13
3.5.3. The Event Cohesion Indicator	15
3.5.4. The Cohesion of Logical Connectors	17
4. Validation issues	19
5. The preliminary application	21
6. Conclusions	24
Reference	

1. Introduction

Enterprise modelling is an important premise for a successful business process (re)engineering (BPR) project (J. Brown, 1995). To facilitate this first step in any BPR initiative, the business process engineering community has developed different modelling methods and produced numerous tools implementing them. Although the commercially available BPR and CASE-tools support a lot of modelling activities (e.g. developing models, customizing reference models, simulating processes), today's modellers still meet difficulties in:

- introducing quality assurance principles into enterprise modelling;
- validating the produced business process models according to a disciplined and documented procedure.

The lack of comprehensive approaches to quality control of business process models and sound procedures for model validation results in producing inconsistent, incomplete and difficultly-interpretable and maintainable information models that impede the BPR-project participants.

This paper addresses the above issues and proposes to overcome the difficulties by a systematic and feasible solution based on the benchmarking concept. The work we describe is being carried out at the Institut für Wirtschaftsinformatik (IWi), University of the Saarland, in the context of the project „PROcess model BEnchmarking“ (PROBE), a research initiative supported by IDS Prof. Scheer Inc, Saarbrücken, Germany. The project aims at developing a computerized system (an extension of the ARIS-Toolset) capable of controlling the quality of process models via benchmarking.

2. The PROBE Project

2.1. Project Objectives

The present research project is based on the idea that the enterprise models could and should be improved by learning from the features of models recognized as standards or best-in-class ones.

Moreover, we state two basic 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.

This leads us to formulate the main objective of the PROBE-project: *to provide the enterprise modelling community with a comprehensive framework and a tool that allow controlling the quality of business process models via benchmarking.*

The modellers will be able to use the framework and the tool:

- *in model diagnose mode*, to help identify weaknesses and potential defects in business process models.
- *in progress measurement mode*, to help monitor the process of model improvement.

These use cases are summarized bellow:

Diagnosis: Given a set of enterprises, business processes of a given type have to be modelled. This results in developing a set of models which further can be compared against a branch-specific reference model that serves as a benchmark. Such a benchmarking study can reveal weaknesses and potential defects in the models. By focusing on areas that are deemed too complex, too large or improperly modelled, the modellers are able to substantially enhance their models.

Progress measurement: Once a diagnostics study has been completed and improvement actions has been performed, the next question to be answered is how well the modellers proceed. To measure the progress over time, different model versions should be benchmarked. This is aimed at obtaining comparative figures that represent graphically the current state of the model enhancing process.

2.2. The Project Strategy

To achieve the project objectives, we proposed a strategy including the following steps:

1. Proposal of a generic measure set (benchmarks). This involves the identification and operationalization of quality measures relevant to business process models.

2. Empirical validation of the measures. This implies:
 - (i) the application of the measures to process models developed by IWi, IDS Prof Scheer Inc, and SAP AG,
 - (ii) the study of the consistency between the measurements and expert ratings, and
 - (iii) the refinement of measure definitions, if necessary.
3. Development of a prototype system (an extension of the ARIS-Toolset) for automatic evaluation and comparison of business process models.
4. Testing the applicability of the PROBE benchmarking concept in other enterprise modelling projects.

2.3. The PROBE Products

The above given strategy is aimed at producing the following products (Fig. 1):

1. A procedure for benchmarking business process models.
2. A benchmark suite.
3. An architecture for software system supporting the process of benchmarking.
4. A prototype software system.
5. Benchmarking knowledge base/case studies.
6. A project vocabulary.

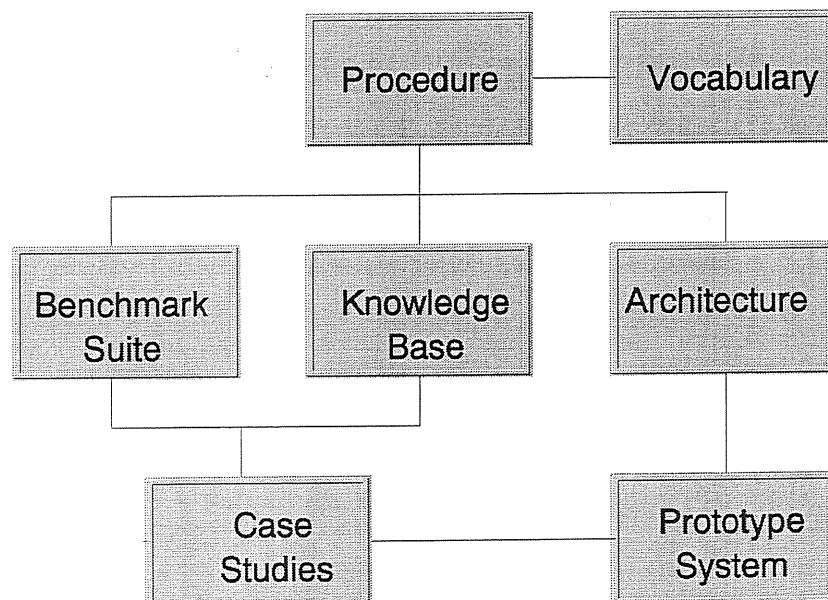


Fig.1. The products of the PROBE-project.

The *benchmarking procedure* describes a systematic sequence of steps to be followed in order to gain benchmarking information. The procedure organizes the temporal succession, iteration and parallelism of benchmarking activities. The *benchmark suite* specifies the measurement basis on which the business process models can be compared. It comprises (1) a benchmark system that defines the relationships between quantifiable model attributes and the model's features we are interested in, but which are not directly measurable, (2) a specification of a generic benchmark set, and (3) measurement instruments that specify the data to be collected.

The application of the *benchmarking procedure* can be time-consuming and error-prone, especially when a model includes many objects and relations among them. To facilitate the modellers an automation of the procedure is required whereby the benchmarks can be evaluated, the quantified information stored and benchmarking reports generated. For this purpose, the PROBE-project previews the development of an *architecture* of decision support system for benchmarking, as well as, a prototype system compliant with this architecture. The letter describes system components and the ways they behave, communicate, interact and coordinate each with other (Garlan & Perry, 1995). Based on this architectural description, a *prototype system* should be developed. It is a prototype in that its benchmark set is currently limited to several measures. In all other respects, the system should be fully operational. The preliminary goal with this prototype is to capture benchmarking information by means of conducting *case studies*. Any case study results from the application of the *benchmarking procedure* on a selected set of process models and a relevant subset of benchmarks. The case studies should be stored in the *benchmarking knowledge base* which is aimed at supporting the analysis of gaps between the benchmarked models and at helping make the benchmarking know-how reusable.

To conclude, the results of the PROBE-project are intended to help:

- *modellers*, who can validate business process models.
- *process designers*, who need to optimize reference models and customer-specific models.
- *quality managers*, who will be able to handle the model quality by means of quantitative data.
- *model customizing teams*, who will be facilitated to anticipate fallacies and pitfalls in integration of submodels and/or customization of reference models.
- *BPR-project managers*, who need to improve effort/cost estimation within the phase of business process analysis and modelling.

3. The PROBE Solution

3.1 Background

The development of a sound and workable solution to the problem of model quality evaluation is an ambitious goal whose achievement requires input from both business process modellers and research community. This is valuable in order to keep the products of the PROBE-project close to the practice. Therefore, we propose to develop our solution strategy on the base of a synergic combination between software metrics technology and business process engineering practice. We take account of both enterprise modelling experience and quality management approaches within industrial projects. Specifically, we have used the following major sources:

1. The ARIS methodology for enterprise modelling and information systems development (Scheer, 1995).
2. State-of the-art experience in quality evaluation of semantic models, e.g. Entity-Relationship Models (Batini, 1992, Heilandt & Kruck, 1993, Hars, 1993, Kesh, 1995).
3. The AMI approach to goal-driven derivation of software metrics (Pulford et al, 1995).
4. The concept of benchmarking as a controlling tool in information management (Heib et al, 1996-a, Heib et al, 1996-b) and the holistic approach to software benchmark design and use (Daneva, 1995).

The Architecture of Integrated Information System (ARIS) methodology for information modelling proposed by Scheer (1995) is the context where the benchmarking is intended to be applied as a controlling tool. Generally, the objective of ARIS is to facilitate the specification and the implementation of information systems supporting business processes. The ARIS methodology predefines four descriptive views (data, function, organization, and control view) and three levels (requirements definition, design and implementation) as depicted in Fig.2. Thus, it consists of 12 components. For each component a set of suitable and integrated description methods is previewed. The languages used for enterprise modelling at requirements definition level are: extended Entity-Relationship Model for describing the data view, hierarchy diagram for the function view, organizational charts for the organization view, and extended process chain (EPC) diagrams for the control (process) view. Any model identifies various categories of information in a statement of requirements and structures them so as to be easy accessed by the team responsible for developmental activities.

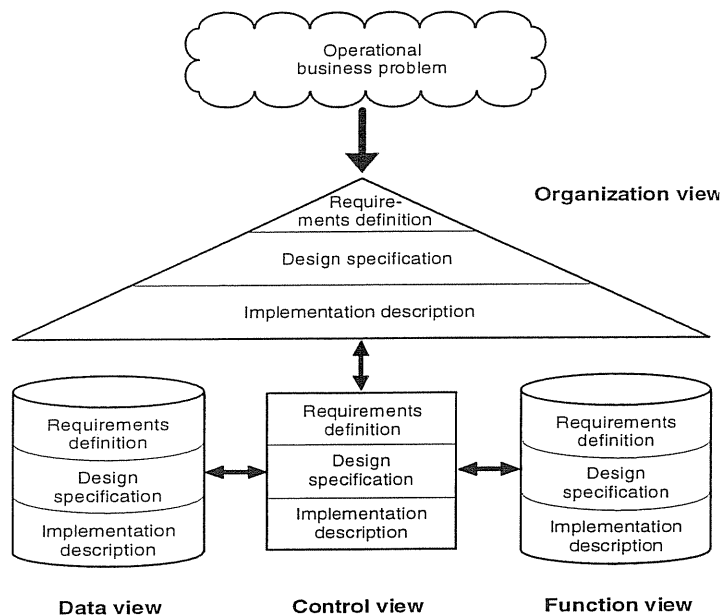


Fig.2. The ARIS methodology.

The application of the ARIS methodology is facilitated by means of the ARIS-Toolset (IDS Prof. Scheer Inc., Saarbrücken, Germany), whose functionality covers the creation and the modification of reference models, and the derivation and the configuration of specific enterprise models from reference ones.

Next, the PROBE-project focuses on the control view, i.e. the business process models we intend to study, are specified by using the EPC-formalism (Scheer, 1995). It comprises three basic modelling constructs: events, functions and logical connectors (*and*, *or*, and *exclusive or*). The functions are considered as time-consuming activities that create or modify objects. The events are both conditions for the execution of functions, and results. The events refer to points in time and are defined as data items that should be available at the start and at the end of functions' execution. In the EPC-schemes, the functions are represented by rounded boxes and the events as hexagons. An example of an EPC-scheme is given in Fig. 4 (Section 3.5.1.).

Furthermore, our method is based on the lessons learnt from the ESPRIT CODE-Project (Hars et al, 1992, Hars, 1993), which identified the value of the quantitative measurement as a efficient way of ensuring the quality of reference models. We also follow the AMI tradition and Basili's framework (Basili et al, 1988) to goal-driven derivation of model quality metrics, whose emphasis on goals leads to emphasis on what actions can be taken. Thus, the approach we present emphasizes two key points. First, the current objective must be defined, as it is stated in (Gilb, 1987): "Actions without clear goals will not achieve their goals clearly". Second, the goal

formulated is analyzed according to the objects (and measurable objects' attributes) managed and controlled by enterprise modellers.

Given a set of metrics derived from goals, we use the concept of benchmarking as a controlling tool (Heib et al, 1996-b). Generally, „benchmarking is a continual assessment of business objects against the best-in-class ones or a standard, based on measurable characteristics“ (Heib & Daneva, 1995). Hence, within the PROBE-project the benchmarking method means more than a single assessment of model quality; its emphasis is on comparison of business models against best-in-class ones. To ensure the systematic application of the benchmarking method, we follow the holistic approach for software benchmark design and use proposed in (Daneva, 1995) whose focus is on the steps of the software benchmarking process and how they are to be performed.

In the context of the above references, we introduce a documented and disciplined quality assurance procedure based on benchmarking. It comprises five basic steps:

1. *Define measures of model quality.*
2. *Select a best-in-class model. This can be a reference model or a representation, that describes the level of model quality that should be achieved.*
3. *Evaluate the studied models on the quality measures.*
4. *Compare assessed models against the best-in-class one.*
5. *Formulate corrective actions about the enhancement of the studied models.*

In this paper, we look at the step 1. We present the first version of our benchmark suite and illustrate how it can be used.

Before developing a sound benchmarking approach, it is worth clearing two basic points:

- what is a high quality business process model, and
- how model quality features can be approached systematically.

Once these concepts are established, two operational questions come into focus:

- what quality issues can be subjected to benchmarking, and
- how these can be quantified.

Our work addresses subsequently the above four questions. So, the next sections present our definition of model quality, the design of the quality measure system, a brief overview of some model quality issues to be considered, and the specifications of single measures.

3.2 The Definition of Quality in the PROBE-Project

Following Hars (1993) and Kesh (1995), any enterprise model should be adequate to some organization goal. Specifically, in the PROBE project, we considered three general benefits of process models for an organization (Heib et al, 1996-a):

- the models are automatable and transportable knowledge which helps the BPR-team to systematize and plan process improvement actions. It also provides a sound and consistent documentation of the represented processes.
- the models ensure a higher degree of transparency within the organization and thus, provide a basis for discussion on business processes themselves.
- the models provide a holistic description which integrates all relevant aspects (data, function, organization and process) of the business processes. The documentation should be exact enough to serve as a starting point for an EDP implementation.

Having these statements in mind, we propose the following definition of quality:

A business process model is said to be of a good quality if it provides the above given benefits for an organization unit.

3.3. Designing a Benchmark System

The ISO 9126 document (ISO, 1991) as well as the STEP standard (Tyler, 1992) suggest to develop an hierarchical quality measure system flexible enough to be customized to the needs of any benchmarker. We refine this idea by developing three level schema for evaluation of process models (Fig. 3).

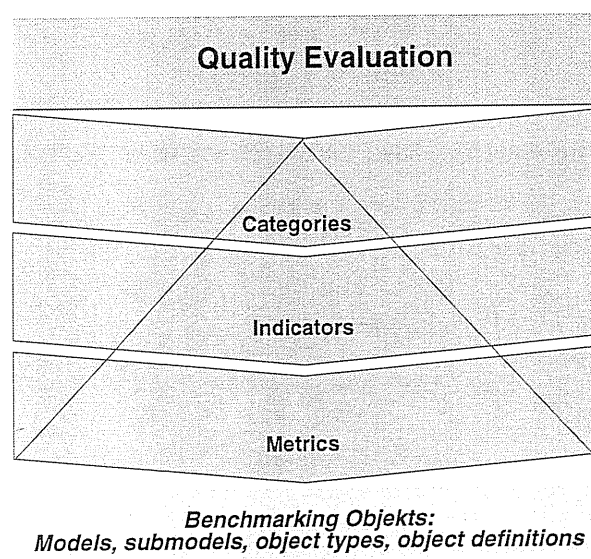


Fig.3. The PROBE benchmark system.

It consists of quality *categories*, *indicators* and *metrics*. The *categories* provide a high level view to the types of benchmark measurements performed on an EPC-schemes. Any category refers to the model as a whole, and not to submodels, object types and object definitions. The *indicators* are measurable aspects of the quality for any given category. The indicators are quantified by evaluating one or more facts about an EPC-model. Each fact to be evaluated is called a *metric*. Thus, both metrics and indicators are quantifiable, i.e. for each model there exists a number that is the value of certain model attribute expressed by an indicator or by a metric. Both metrics and indicators are quantified by measurement methods, which are step-by-step instructions on how measurements have to be carried out. The methods can be also expressed by means of a formula for calculating indicator's value or as a rule how to evaluate a metric (Heib et al, 1996-a).

The benchmarker can investigate business models at any of these levels.

3.4. Quality Issues Suitable for Benchmarking

Since the business process models are representations of the enterprise, it was obvious that we should consider quality issues addressed by metrics referring to enterprise data models, for example Entity-Relationship models. Since the process models can be seen as IS specifications at the requirements definition level (Scheer, 1995), it was helpful to get insight on the metrics dealing with the quality of software work products. Furthermore, since our definition of quality does not refer to the pure modelling process only, the metrics had to address the issues of the use and the customization of business process models. For these reasons, to establish the set of PROBE-categories we have conducted a careful analysis of the software features referenced by 27 publications and submitted for consideration by:

- ❑ software standard bodies: ISO (ISO, 1991), SPICE (Dorling, 1993), STEP (Tyler, 1992), BOOTSTRAP (Kuvaja et al, 1994).
- ❑ data modelling research community: ESPRIT-CODE-Project consortium (Hars et al, 1993, Hars, 1993), and other data engineering project teams (Batini, 1992, Heilandt & Kruck, 1993, Kesh, 1995).
- ❑ world-wide known software benchmarking consulting companies: Software Productivity Research (Jones 1995-a, Jones 1995-b, Jones 1996), Software Productivity Centre (Strigel, 1995), and Software Productivity Solutions (Card & Layman, 1996)
- ❑ computer industry: IBM (Albrecht, 1979, Brooks, 1995), Hewlett Packard (Grady, 1992), AT&T (Jones, 1995-a).

□ software engineering research community: McCabe (1976), Halstead (1977), Dreger (1989), Fenton (1991), Chidamber & Kemerer (1994), Churcher & Shepperd (1995), Kan (1995), Paulk et al (1995), Peralta et al (1995), Sharble & Cohen (1995).

□ government agencies: the U.S. Department of Defense (Brown, 1995), the U.S. Army (Dyson et al, 1993), Canadian government institutions (Rubin, 1995).

Based on a brain storming sessions and discussions together with IDS consultants, we have selected seven categories of model quality for further research: granularity, documentation level, modularity, semantic correctness, complexity, standardization level, and understandability. More of these have previously been discussed by A. Hars in his PhD Dissertation (1993). However, he limits his research to Entity-Relationship representation language, and does not discuss the ARIS process view and process modelling at all. Thus, we have appropriately refined Hars' categories. The PROBE categories are defined as follows:

- **Granularity:** It accounts for the details of the universe of discourse included in the representation.
- **Documentation level:** It refers to the extent of completeness to which the requirements to the model are met.
- **Modularity:** It is the ability of the model to build clusters.
- **Semantic correctness:** It provides information whether the modelling constructs represent correctly the universe of discourse.
- **Complexity:** It is the ability of model elements to build nested control flows and decision structures.
- **Standardization level:** It is the extent to which a model conforms with preliminary established standards within the modelling project.
- **Understandability:** It refers to how easy the user understands the modelling constructs without any further help.

3.5. The PROBE Benchmark Suite

To identify relevant indicators for each category, we investigated the quantifiable software metrics proposed in the above reference (see section 3.4.). First, we checked out which metrics are deemed relevant for studying business process models. Then, we tried to re-formulate the candidate metrics to the peculiarities of the EPC-modelling technique and we classified these metrics to the seven categories. After a round table discussion with the IDS experts it was decided to concentrate our research efforts on a single quality category, and further, to operationalize it through a set of 3-5 indicators. We decided to begin with the **complexity**. Thus, a set of appropriate complexity

indicators has been defined. These are synthetic measures derived from six visible attributes of the EPC-models: input objects, output objects, logical connectors, levels of nesting, cohesion factors, and simultaneously acting objects (so called co-objects).

The indicators were identified by analyzing benchmarking goals, i.e. the goals referring to what information the benchmarking study on business process models has to reveal. Each benchmarking goal was further broken down into a set of primitive measurement goals which are easily expressed by the attributes of the EPC-schemes. For any indicator we explained how it helps us improve the process models and how it would be more difficult to make improvements without the indicator. Finally, the indicators were specified by following AMI guidelines for metrics description (Pulford et al, 1995). We proposed to specify the indicators by using the following template:

<i>Name:</i>	Name of the benchmark
<i>Reference</i>	What model unit the benchmark is applied to?
<i>Definition:</i>	What does the benchmark mean?
<i>Goal:</i>	Why do we need the benchmark?
<i>Quantification Rules:</i>	What should be done to obtain benchmark numbers?

In the next section, we give an overview on the complexity indicators. Before giving their specifications, it is important to emphasize that our objective with this is precisely not to claim that our benchmarks are the final and complete word on EPC-relevant complexity indicators; we simply submit these as a candidate suite for further refinement and extensions.

We suggest to present the benchmark definitions together with an illustrative example.

3.5.1. An illustrative example

To show how benchmark data are collected we use the EPC in Fig. 4 which depicts the high level definition of the process „Check received transaction“ (SAP, 1994). This EPC is a reference model of the system SAP R/3 Analyzer 2.2.

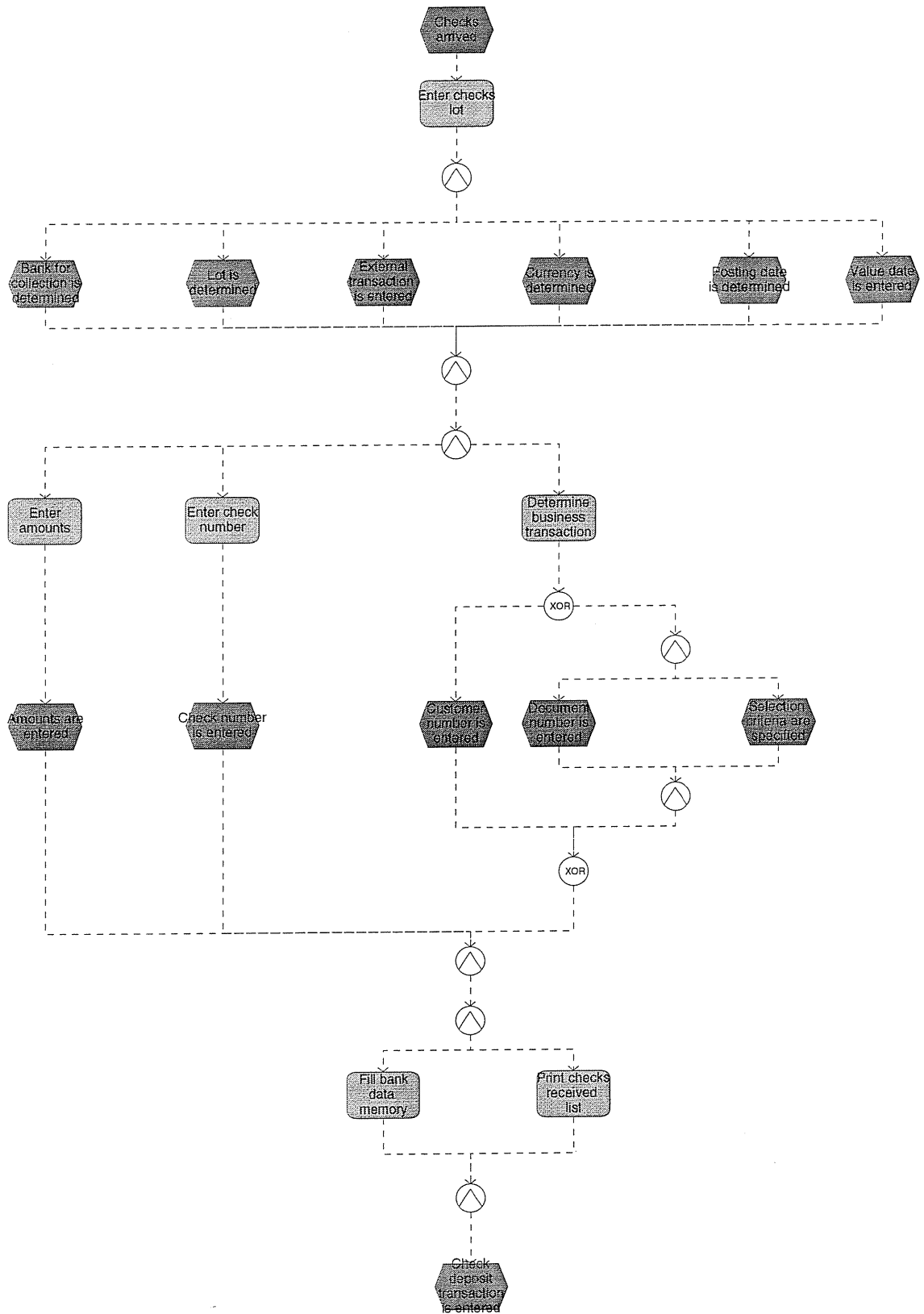


Fig.4. The reference model *Check received transaction*, SAP R/3 Analyzer 2.2.

3.5.2. Function Cohesion (FC)

In the software metrics literature, there are several established measurement models, that deal with the complexity of functions in software systems (Bauman & Richter, 1992). All of them consider the complexity in relation with some quantifiable features of input and output data items handled by a function. We follow this tradition together with the idea of nesting levels (Fenton, 1991), and McCabe's (1976) and Rechenberg's (1986) metrics based on program control structure.

In this work, we consider the fact that the events of an EPC can be nested due to their relations with logical connectors. We define the nesting level of an event with respect to a given function to be the number of connectors that build the path from the function to the event. A function may be started by a set of events allocated at different levels of nesting. For example, the functions *Fill bank data memory* and *Print checks received list* (Fig.4) are started by five events which are allocated at three different nesting levels. These are built by the connectors AND, XOR and AND. Note that we do not take account of the connector AND which precedes both the functions, since it links these functions, and hence, it does not refer to the starting events. The events *Document number is entered* and *Selection criteria are specified* build the level 3. Next, the event *Customer number is entered* finds itself at the nesting level 2. The rest two events *Amounts are entered* and *Check number is entered* are logically associated to the nesting level 1.

To consider the complexity of the connectors through which the events are nested, we use Rechenberg's measurement approach to complexity of logical expressions (Rechenberg, 1986). We assigned a weight to each connector (AND, OR and XOR) by modifying slightly Rechenberg's weighting scheme (Table 1.). When a function is started/ended by one event only, the weight is assumed to be equal to 1.

Logical connectors	Weight
AND	3
XOR	3
OR	2

Tab. 1. Weighting factors for logical connectors.

The definition of the benchmark Function Cohesion is given bellow:

Name:	Function Cohesion
--------------	-------------------

Reference:	Function Type
-------------------	---------------

Definition:	The Function Cohesion measures the intensiveness of the control flow handled by a given function. The indicator quantifies the structural complexity of the controls which define the start and the end of the function.
--------------------	--

Goal:	<p>How complex can be the query required to check the start and the end of a function?</p> <p>How costly is the access to the data handled by a function?</p> <p>What model's changes can be caused when an operation (deletion, replacement) is applied to a function?</p>
--------------	---

Quantification Rules:	$FC = (FC_{inp} + FC_{out})/2$ <p>where FC_{inp} and FC_{out} are the input and output function cohesion measures. These are defined as follows:</p> $FC_{inp}(n) = k_n N_n$ $FC_{inp}(i) = k_i (FC_{inp}(i+1) + N_i) \quad i=1, \dots, n-1$ $FC_{out}(l) = k_l M_l$ $FC_{out}(j) = k_j (FC_{out}(j+1) + M_j) \quad j=1, \dots, l-1$ <p>where i and j are nesting levels, k_i and k_j are weights, N_i is the number of the input events existing at the nesting level i, and M_j is the number of the output events at the nesting level j.</p>
------------------------------	--

Examples:

$$FC (\text{Enter checklist}) = (1+6 \times 3)/2 = 9.5$$

$$FC (\text{Determine business transaction}) = 6 \times 3 + (2 \times 3 + 1) \times 3 / 2 = (18 + 21) / 2 = 19.5$$

$$FC (\text{Fill bank data memory}) = ((2 \times 3 + 1) \times 3 + 2) \times 3 + 1 / 2 = 70 / 2 = 35$$

3.5.3. Event Cohesion (EC)

Unlike the indicator described in the previous section, the focus of the Event Cohesion is on how frequently and in what manner an event is used. The form of equation standing for a quantification rule is fast analogous to this of the Function Cohesion. The difference is in the use of the notion of event cohesion factor, that specifies how an event connects itself with functions. It characterizes the capability of an event to participate in complex control structures. We establish the following rules for defining the cohesion factor for an event:

1. The cohesion factor for an event equals to 1, if the event is produced by one function only and is an input to one function only. The cohesion factor 1 refers also to any start event that is an input to one function only, as well as, to any end event produced by one function only.
2. The cohesion factor for an event equals to 2, if the event is produced by more than one functions, but is an input to one function only. The cohesion factor 2 refers also to any end event produced by more than one functions.
3. The cohesion factor for an event equals to 3, if the event is produced by one function only and is an input to more than one functions. This cohesion factor refers also to any start event that is an input to more than one functions.
4. The cohesion factor for an event equals to 4, if the event is produced by more than one functions and is an input to more than one functions.

The specification of the Event Cohesion measure follows:

Name:	Event Cohesion
Reference:	Event
Definition:	The Event Cohesion measures the contribution of an event to build complex control structures in the model.
Goal:	How intensive the data objects referenced by an event are queried? How complex is an interface between different control flows? How frequent an event is used in building nested control structures?
Quantification Rules:	$EC = EC_{inp} + EC_{out} + c.NF$

where EC_{inp} and EC_{out} are the input and output event cohesion measures, c is an event cohesion factor, and NF is the number of functions directly affected by the event under consideration. The values of EC_{inp} and EC_{out} are defined as follows:

$$EC_{inp}(n) = k_n NI_n$$

$$EC_{inp}(i) = k_i \cdot (EC_{inp}(i+1) + NI_i) \quad i=1, \dots, n-1$$

$$EC_{out}(l) = k_l NO_l$$

$$EC_{out}(j) = k_j \cdot (EC_{out}(j+1) + NO_j) \quad j=1, \dots, l-1$$

where i and j are levels of nesting, k_i and k_j have the above meaning, NI_i is the number the of co-events at the level i , with which the event acts together as an input to a function, and NO_j is the number the of co-events at the level j , with which the event acts together as an output from a function.

Examples:

$$EC \text{ (Bank for collection is determined)} = 5 \times 3 + 5 \times 3 + 3 \times 4 = 42$$

$$EC \text{ (Amount are entered)} = 4 \times 3 + 1 + 3 \times 3 = 22$$

$$EC \text{ (Customer number is entered)} = 2 \times 3 + (2 \times 3 + 2) \times 3 + 3 \times 3 = 6 + 24 + 9 = 39$$

3.5.4. Cohesion of a Logical Connector (CLC)

The indicator Cohesion of a Logical Connector quantifies the complexity of a given control structure built by one or more logical connectors (AND, OR and XOR). The reason behind this indicator lies on the hypothesis that a random connecting of functions or events will result in difficulties to maintain an EPC-model; it will impede implementors and can result in inaccuracies when the information system is used, and can create performance problems. Thus, the CLC indicator is composed by two components: the weighted number of objects (NO) directly related with the connector, and the number of arrows (NA) between these objects and the studied connector. To develop a equation for obtaining a single value from the above numbers, we follow the suggestions of Shari Pfleeger from Systems/Software Inc. (1995) about how to derive a scalar measure from vectors. According to her, to obtain a valid equation, we can refer to Euclidean geometry and use the relationship between position and distance. We consider the two parameters as coordinate axes in a Cartesian plane (Fig. 5).

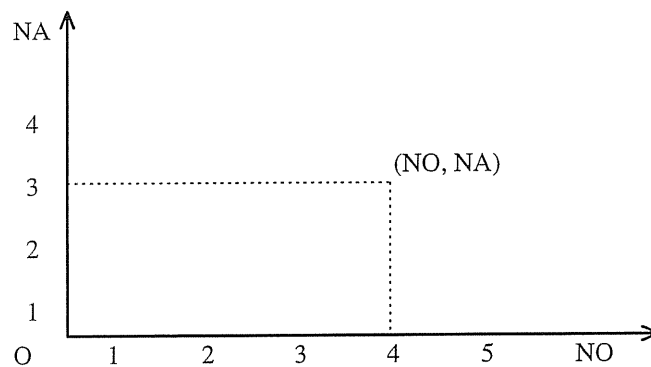


Fig. 5. The indicator's components mapped on the two dimensional space.

Let NO correspond to the X-axis, and NA - to the Y-axis. The geometry says that the distance of the point (x,y) from the origin O can be calculated from the formula

$$d = \sqrt{x^2 + y^2}.$$

We would use this formula as a quantification rule.

The description of the indicator is presented as follows:

Name:	Cohesion of a Logical Connector
Reference:	Logical connector
Definition:	The Cohesion of a Logical Connector quantifies the ability of a logical connector to maintain nested control flows.
Goal:	How to distinguish control intense models from data intense ones? Does a model tend to build compound decision structures? How intense is the control flow passing through a logical connector? What complexity is produced due to chaining logical connectors?
Quantification Rules:	$CLC = \sqrt{\frac{2}{NO} + \frac{2}{NA}}$

where NO is the weighted number of objects (events, functions or connectors) directly related with the connector under consideration, and NA is the number of arrows between these objects and the studied connector. The parameter NO is expressed by:

$$NO = NF + NE + 2 \cdot NC$$

where NF is the number of functions, NE - the number of events, and NC - the number of connectors directly related to the studied connector. To differentiate between the related connectors and the other related objects, weights have been assigned: the number of connectors is weighted by 2, and these of events and functions - by 1.

Examples:

$$CLC (\text{AND after the function } \textit{Enter checks lot}) = \sqrt{\frac{2}{(1+6)} + \frac{2}{7}} = 7\sqrt{2} = 9.9$$

$$CLC (\text{AND after the events } \textit{Amounts are entered} \text{ and } \textit{Check number is entered}) = \\ = \sqrt{\frac{2}{(1+1+2)} + \frac{2}{4}} = \sqrt{52} = 7.21$$

4. Validation Issues

To investigate the expressive power and the adequacy of the developed benchmarks, we followed Fenton's guidelines (Fenton, 1991) for software metrics validation. The operationalized benchmarks were applied to some models of the SAP R/3 Analyzer 2.2, as well to models developed by IWi and IDS Prof Scheer Inc. (Table 2). The quantified results were compared with evaluation opinions of IDS experts referring to the same models. We refined benchmark quantification rules when our method did not come to the same conclusion as the experts. It is worth mentioning that the benchmarking information of our first studies was collected manually. It included the evaluation of 11 business models with size between 14 and 214 objects, what took an effort of six man/weeks. This made clear the need for a computer-aided benchmarking of business process models.

Developer	Model	Number of objects	Number of functions	Number of events	Number of connectors
SAP AG R/3 Analyzer 2.2	Down payment processing (FI)	79	11	40	28
	Customer down payment request (FI)	14	4	6	4
	Vendor master data branch copy (FI)	57	12	23	22
	Lagermaterialentwicklung (MM)	214	51	101	62
	Chargenabwicklung (MM)	201	46	96	59
IWi and SAP AG SINCOM-2 Project	Payment handling	26	4	14	8
	Payment request	74	22	34	18
	Bank-reconciliation	48	15	19	14
	Recovery order	79	25	36	18
IDS Prof Scheer	Human resource administration As-Is	38	10	17	11
	Human resource administration As-To	16	7	9	0

Table 2. The evaluated EPC-models.

Furthermore, the discussion based on the model evaluation reports has lead us to the following conclusions:

- the indicators we have submitted quantify the presents or the absence of complex control structures in an EPC-model.
- the main benefit of the proposed benchmark suit is that the measures help identifying error-prone model fragments.

- a further study on the statistical independence among the benchmark values is needed to point out that the indicators quantify different features of EPC-models.
- the current version of benchmarks could not serve as a reliable basis to directly evaluate the efficiency and the effectiveness of the business processes represented in the models.

5. The Preliminary Application

To clarify the PROBE benchmarking concept we present a case study on two EPC models for the process *human resource administration*. Both the models have been developed by an international consulting team (with the participation of the IDS Prof Scheer Inc.) within a BPR-project in Eastern Europe.

The Model 1 (see Appendix 1) represents the current (As-Is) process at the enterprise X, an East-European electric company. The model is composed from 10 functions, 17 events and 11 logical connectors.

The Model 2 (see Appendix 2) is a reference model for the process *human resource administration* and represents an optimized process to be designed and implemented at the enterprise X. Therefore, for the purpose of our study, we selected this model as a best-in-class one. It contains 7 functions and 9 events, and is characterized by a lineal control flow. Logical connectors are not used.

Both the models have been assessed and benchmarked on the basis of the three indicators described in the previous sections. For each model we have evaluated the objects (functions, events, and logical connectors) with respect to the benchmarks and then we have calculated the average indicator values. Our study has resulted in generating two types of benchmarking figures: *model profiles* made up from the average values, and *comparisons* of the models on object-by-object basis. These finer comparisons can be made only when the benchmarked models have identical objects. Such figures help modellers identifying critical objects for which measured benchmark values differs from required ones.

The model profiles from our study (Fig. 6) show that the average evaluations of the model of the company X exceed those of the best-in-class model. Note that the average logical cohesion of the best-in-class model equals to 0 due to the lack of logical connectors in the EPC-scheme.

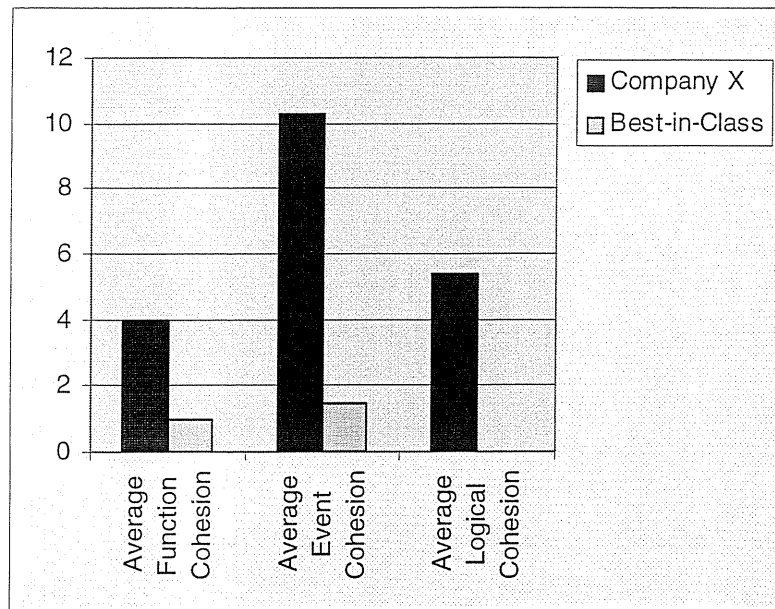


Fig. 6. Benchmarking profiles: average numbers.

Moreover, we identified 6 functions and 4 events that the studied models have in common. These are compared on the basis of their evaluations on the benchmarks Function Cohesion and Event Cohesion, respectively (Fig. 7 and Fig 8.).

Next, we would show how the produced benchmarking figures can be used and how they can deliver a roadmap for model improvement initiatives. First, the benchmarking profiles from Fig. 6. identify what object type (a function, an event or a logical connector) is deemed be extremely complex. Given this quantitative information, to define certain corrective actions, the modellers need a roadmap and guidelines where to begin from. Therefore, the further investigation on model complexity need to be focused on each particular benchmark. To identify which objects contribute the most to the high average benchmark values we compared the models on an function-by-function and event-by-event basis, respectively. Let us analyze the benchmark Function Cohesion (Fig. 7). As it is evident from the data, the high average Function Cohesion is due to the functions *Confirm internal job change*, *Confirm final employment*, *Prepare internal job change*. These can be submitted to the modellers for further enhancement.

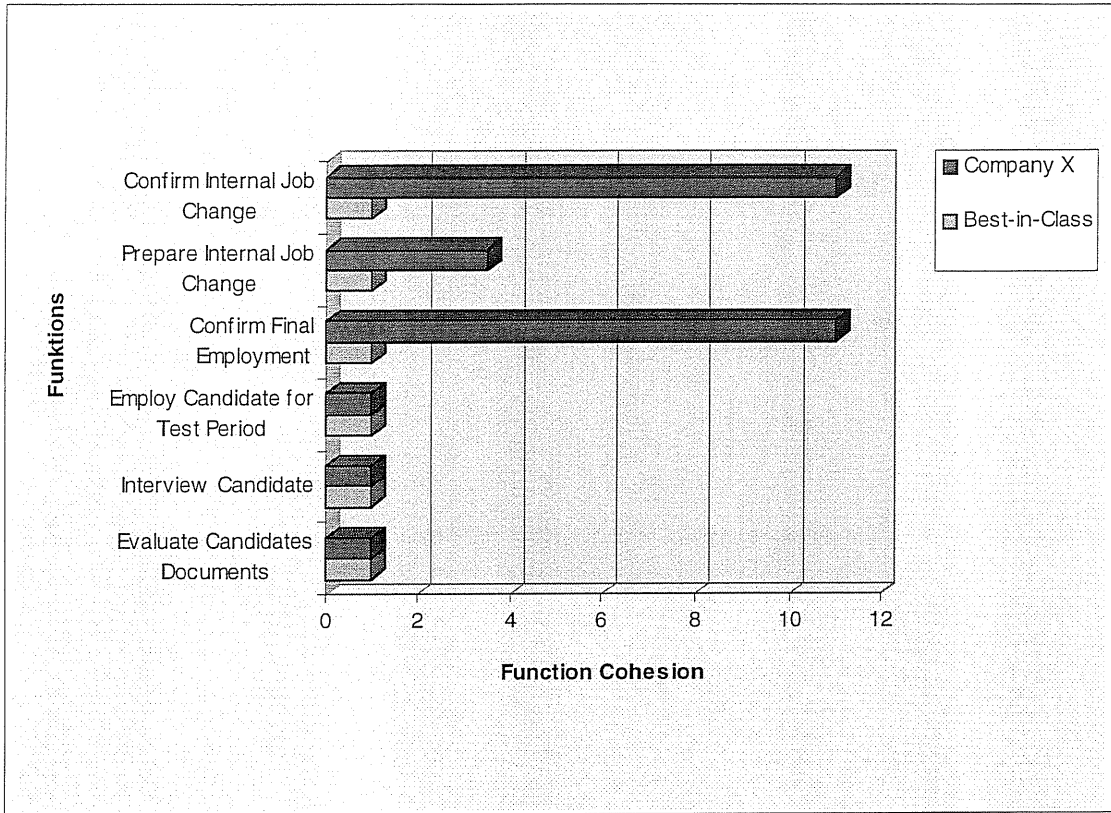


Fig. 7. Comparison on function-by-function basis.

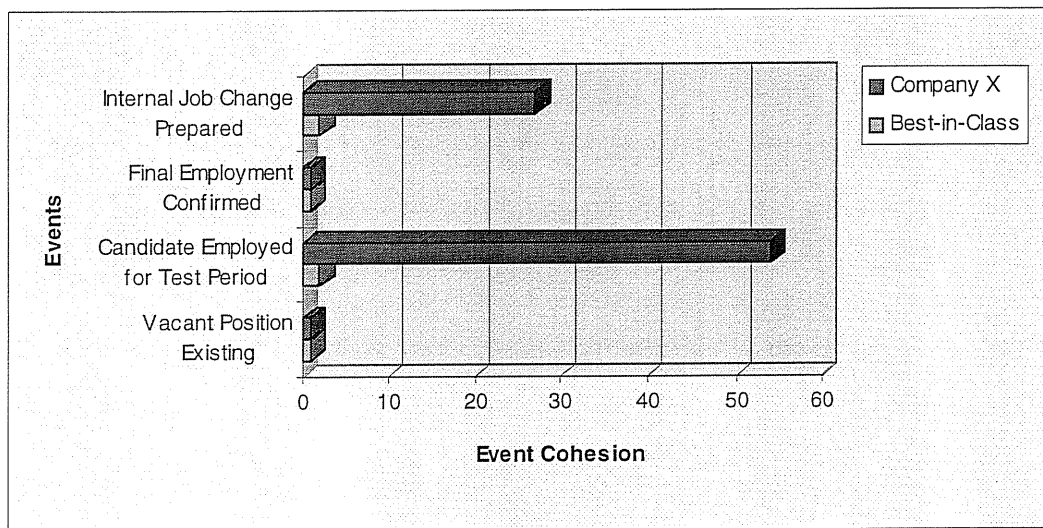


Fig. 8. Comparison on event-by-event basis.

6. Conclusions

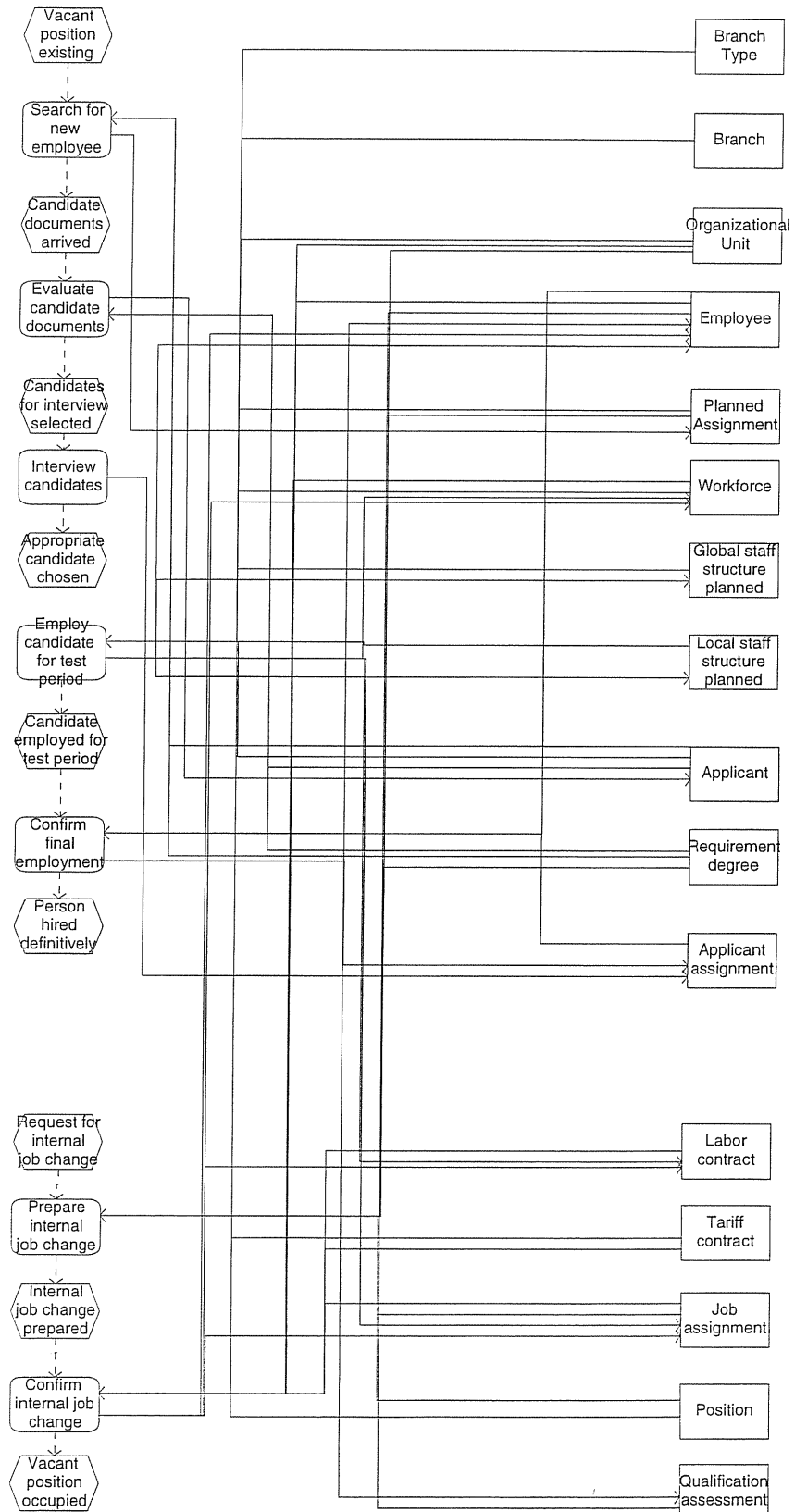
Benchmarking expertise is a set of knowledge and skills which support modellers in controlling quality of business process models. The current results provided by the PROBE-project have shown how the quality issues in enterprise modelling can be mastered according to a documented procedure that makes the model quality quantitatively known. We have proposed an enhanced defect detection method based on benchmarking. Our solution methodology contributes to:

1. introducing quality metrics technology into business process engineering practices,
2. applying a disciplined approach to validate business process models, and
3. ensuring that most faults are captured before the model is converted in a system design, or submitted to the BPR-team.

As the results, it was found that quality measures can be used to detect errors with business process models, as well as to grasp the progress of corrective actions quickly and accurately.

In order for the benchmarking to be feasible and effective, an appropriate tool must be developed which extracts benchmark data and exploits valuable assets that exist in process engineering organizations in the form of reference models and best-in-class practices. Next, to conduct reliable benchmarking, analytical and empirical validation studies on our indicators are needed. The work within the PROBE-project takes some steps toward addressing these issues. Currently we are working on the implementation of a prototype system to gather data on benchmarks from process models developed at IWi and IDS Prof Scheer Inc. As a continuation of our validation research, we intend to define a set of desirable properties that the model complexity benchmarks should possess, and to determine whether or not the submitted indicators (Function Cohesion, Event Cohesion, and Cohesion of Logical Connectors) exhibit those properties.

Appendix 2
Human Resources Administration: As-To Model



REFERENCE

1. Albrecht, A., Measuring Application Development Productivity; SHARE/GUIDE IBM Application Development Symposium, Monterey, California, Oct., 1979, pp. 83-92.
2. Batini, C., S. Ceri, S. Navathe, Conceptual Database Design: an Entity Relationship Approach, Redwood City et al, 1992.
3. Basili, V., H. Rombach, The TAME Project: Towards Improvement-Oriented Software Environment, IEEE Transactions on Software Engineering, Vol.14, No 6, June, 1988, pp. 758-773.
4. Bauman, P., L. Richter, Wie groß ist die Aussagekraft heutiger Software-Metriken?, Wirtschaftsinformatik, 34/6, Dezember, 1992, S.624-631.
5. Brooks, F., The Mythical Man Month, Addison-Wesley, Reading, MA, 1995.
6. Brown, J. (Ed.) Reengineering the Enterprise, Chapman and Hall, London, 1995.
7. Brown, N. (Ed.), The Program Manager's Guide to Software Acquisition Best Practices, Version 1.0. US Department of Defense, Washington, CD, July, 1995.
8. Card D., B. Layman, Applying Function Point Analysis to Project Management, Measure Up, Software Productivity Solutions, Vol. 2, No 1, January 1996, pp. 1-4.
9. Chidamber S.R., Kemerer C. F., A Metrics Suite for Object Oriented Design, IEEE Transaction On Software Engineering, June 1994, pp. 476 - 493.
10. Churcher N. I., Shepperd M. J., Towards a Conceptual Framework for Object Oriented Software Metrics, ACM SIGSOFT Software Eng. Notes, April 1995, pp. 69-76.
11. Daneva, M., Benchmarking Practice for Software Quality Achievement, Proc. of 5th. International Conference on Software Quality, Austin, Texas, Oct. 22-26, 1995, pp. 443-457.
12. Daneva, M., Software Benchmark Design and Use, In: Reengineering the Enterprise, Ed.: J.Brown, Chapman and Hall, London, 1995, pp. .
13. Dyson P.B., K.A. Dyson, and J.N.McGhan, Streamlined Integrated Software Metrics Approach Guidebook, Software Productivity Solutions, 1993.
14. Dorling, A. Software Process Improvement and Capability dEtermination (SPICE), Software Quality Journal, Vol. 2, No 4, December 1993, pp. 209-224.
15. Dreger, B., Function Point Analysis, Prentice Hall, Englenwood Cliffs, NJ, 1989.
16. Fenton, N., Software Metrics: A Rigorous Approach, Chapman & Hall, 1991.

17. Garlan, D., D.E. Perry, Introduction to the Special Issue on Software Architectures, IEEE Transactions on software Engineering, 21 (4), April 1995, pp. 269-279.
18. Gilb, T., Principles of Software Engineering Management, Addison Wesley, 1987.
19. Grady, R., Practical Software Metrics for Project Management and Process Improvement, Prentice Hall, Englenwood Cliffs, NJ, 1992.
20. Halstead, M., Elements of Software Science, Elsevier, 1977.
21. Hars, A. Referenzmodelle, Gabler, Wiesbaden, 1993.
22. Hars, A., C. Kruse, R. Heib, A.-W. Scheer, Ways of Utilizing Reference Models for Data Engineering, Proceedings on CIM Conference FMEA'92.
23. Heib, R., M. Daneva, Benchmarking - eine Definitionsklärung, Management & Computer, 3.Jg, H.4, 1995.
24. Heib, R., M. Daneva, A.-W. Scheer, ARIS-based Reference Model for Benchmarking, Veröffentlichungen des Instituts für Wirtschaftsinformatik, Heft 128, Saarbrücken, April 1996-a.
25. 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. Accepted.
26. Heilandt, T., P. Kruck, Ein algorithmisches Verfahren zur Bewertung und Verdichtung von Entity-Relationship-Modellen, Informatik-Forschung und Entwicklung, Vol. 8, 1993, S. 197-206.
27. Henry, S., D. Kafura, Software Structure Metrics Based on Information Flow, IEEE Transactions on Software Engineering, SE-7(2), Sept. 1981, 510-518.
28. ISO/IEC 9126, Information Technology - Software Product Evaluation - Quality Characteristics and Guidelines for Their Use, ISO, December, 1991.
29. Jones, C., Patterns of Software System Failure and Success, International Thomson Computer Press, 1995-a.
30. Jones, C., Software Benchmarking, SPR Knowledge Base, Software Productivity Research, Burlington, MA, Jan. 1995-b.
31. Jones, C., Applied Software Measurement, McGraw Hill, 1996.
32. Kan, S., Metrics and Models in Software Quality Engineering, Addison Wesley, Reading, MA, 1995.
33. Kesh, Quality Evaluation of entity-relationship Models, IEEE Transactions on Software Engineering, Vol.21, No 11, Nov, 1995, pp. 848-858.

34. Kitchenham, B., S. L. Pfleeger, N. Fenton, Towards a Framework for Software Measurement Validation, IEEE Transactions on Software Engineering, Vol.21, No 12, December, 1995, pp. 929-943.
35. 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, 1994.
36. McCabe, Th., A Complexity Measure, IEEE Transactions on Software Engineering, SE-2(4), Dec. 1976, pp.308-320.
37. Paulk, M., B. Curtis, et al: The Capability Maturity Model, Addison Wesley, Reading, MA, 1995.
38. Peralta A., Serras J., Slavkova O., Design Quality Metrics for Object-Oriented Software Development, Computer Languages and Systems Department of The Polytechnic University of Catalonia, Report LSI-95-4-R, 1995.
39. Pulford, K., A. Kuntzmann-Combelles, S. Shirlaw, A Quantitative Approach to Software Management, Addison-Wesley, 1996.
40. Rechenberg, P., Ein neues Maß für die softwaretechnische Komplexität von Programmen, Informatik Forschung und Entwicklung, 1986, pp. 26-37.
41. SAP AG, SAP R/3 Analyzer with R/3 Reference Models, Release 2.2., Walldorf, Decembre, 1994
42. SCOPE ESPRIT Project 2151: „SCOPE - Technology for Evaluation and Certification of Software Product Quality“, Project Brochure, November, 1992.
43. Scheer, A.-W., Wirtschaftsinformatik, Referenzmodelle für industrielle Geschäftsprozesse, 6. Auflage, Springer-Verlag, Berlin et al, 1995.
44. Sharble R. C., S. S. Cohen S. S., The Object-Oriented Brewery: A Comparison of Two Object-oriented Development Methods , ACM SIGSOFT Software Eng. Notes, pp.69-73, April 1993.
45. Strigel, W., Assessment in Small Software Companies, Technical Report of Software Productivity Centre, Vancouver, 1995.
46. Tyler, J., Models, Managing Models, Quality Model: an Example of Quality Management, U.S. Department of Commerce, National Institute of Standards and Technology, NISTIR 4738, December, 1992.

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

- Heft 136:** M. Daneva, R. Heib, A.-W. Scheer: Benchmarking Business Process Models, Oktober 1996
- Heft 135:** M. Remme, J. Galler, M. Göbl, F. Habermann, A.-W. Scheer: IuK-Systeme für Planungsinselfn, Oktober 1996
- 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 Planungsinselfn, 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 Planungsinself: 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: URMEEL, 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 68:** W. Kraemer: Einsatzmöglichkeiten von Expertensystemen in betriebswirtschaftlichen Anwendungsgebieten, März 1990
- Heft 67:** A.-W. Scheer: Modellierung betriebswirtschaftlicher Informationssysteme (Teil 1: Logisches Informationsmodell), März 1990
- Heft 66:** W. Jost, G. Keller, A.-W. Scheer: CIMAN - Konzeption eines DV-Tools zur Gestaltung einer CIM-orientierten Unternehmensarchitektur, März 1990
- Heft 65:** A. Hars, A.-W. Scheer: Entwicklungsstand von Leitständen^[1], Dezember 1989
- Heft 64:** C. Berkau, W. Kraemer, A.-W. Scheer: Strategische CIM-Konzeption durch Eigenentwicklung von CIM-Modulen und Einsatz von Standardsoftware, Dezember 1989
- Heft 63:** A.-W. Scheer: Unternehmens-Datenbanken - Der Weg zu bereichsübergreifenden Datenstrukturen, September 1989
- Heft 62:** M. Zell, A.-W. Scheer: Simulation als Entscheidungsunterstützungsinstrument in CIM, September 1989
- Heft 61:** A.-W. Scheer, G. Keller, R. Bartels: Organisatorische Konsequenzen des Einsatzes von Computer Aided Design (CAD) im Rahmen von CIM, Januar 1989
- Heft 60:** A.-W. Scheer, W. Kraemer: Konzeption und Realisierung eines Expertenunterstützungssystems im Controlling, Januar 1989
- Heft 59:** R. Herterich, M. Zell: Interaktive Fertigungssteuerung teilautonomer Bereiche, November 1988
- Heft 58:** A.-W. Scheer: CIM in den USA - Stand der Forschung, Entwicklung und Anwendung, November 1988
- Heft 57:** A.-W. Scheer: Present Trends of the CIM Implementation (A qualitative Survey) Juli 1988
- Heft 56:** A.-W. Scheer: Enterprise wide Data Model (EDM) as a Basis for Integrated Information Systems, Juli 1988
- Heft 55:** D. Steinmann: Expertensysteme (ES) in der Produktionsplanung und -steuerung (PPS) unter CIM-Aspekten, November 1987, Vortrag anlässlich der Fachtagung "Expertensysteme in der Produktion" am 16. und 17.11.1987 in München
- Heft 54:** U. Leismann, E. Sick: Konzeption eines Bildschirmtext-gestützten Warenwirtschaftssystems zur Kommunikation in verzweigten Handelsunternehmungen, August 1986
- Heft 53:** A.-W. Scheer: Neue Architektur für EDV-Systeme zur Produktionsplanung und -steuerung, Juli 1986
- Heft 52:** P. Loos, T. Ruffing: Verteilte Produktionsplanung und -steuerung unter Einsatz von Mikrocomputern, Juni 1986
- Heft 51:** A.-W. Scheer: Strategie zur Entwicklung eines CIM-Konzeptes - Organisatorische Entscheidungen bei der CIM-Implementierung, Mai 1986
- Heft 50:** A.-W. Scheer: Konstruktionsbegleitende Kalkulation in CIM-Systemen, August 1985
- Heft 49:** A.-W. Scheer: Wirtschaftlichkeitsfaktoren EDV-orientierter betriebswirtschaftlicher Problemlösungen, Juni 1985
- Heft 48:** A.-W. Scheer: Kriterien für die Aufgabenverteilung in Mikro-Mainframe Anwendungssystemen, April 1985
- Heft 47:** A.-W. Scheer: Integration des Personal Computers in EDV-Systeme zur Kostenrechnung, August 1984
- Heft 46:** H. Krcmar: Die Gestaltung von Computer am-Arbeitsplatz-Systemen - ablauforientierte Planung durch Simulation, August 1984
- Heft 45:** J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS-D, Ein Werkzeug zur Messung der Qualität von Software-Systemen, August 1984
- Heft 44:** A.-W. Scheer: Schnittstellen zwischen betriebswirtschaftlicher und technische Datenverarbeitung in der Fabrik der Zukunft, Juli 1984
- Heft 43:** A.-W. Scheer: Einführungsstrategie für ein betriebliches Personal-Computer-Konzept, März 1984
- Heft 42:** A.-W. Scheer: Factory of the Future, Vorträge im Fachausschuß "Informatik in Produktion und Materialwirtschaft" der Gesellschaft für Informatik e. V., Dezember 1983
- Heft 41:** H. Krcmar: Schnittstellenprobleme EDV-gestützter Systeme des Rechnungswesens, August 1983, Vortrag anlässlich der 4. Saarbrücker Arbeitstagung "Rechnungswesen und EDV" in Saarbrücken vom 26. - 28.09.1983
- Heft 40:** A.-W. Scheer: Strategische Entscheidungen bei der Gestaltung EDV-gestützter Systeme des Rechnungswesens, August 1983, Vortrag anlässlich der 4. Saarbrücker Arbeitstagung "Rechnungswesen und EDV" in Saarbrücken vom 26. - 28.09.1983
- Heft 39:** A.-W. Scheer: Personal Computing - EDV-Einsatz in Fachabteilungen, Juni 1983
- Heft 38:** A.-W. Scheer: Interaktive Methodenbanken: Benutzerfreundliche Datenanalyse in der Marktforschung, Mai 1983
- Heft 37:** A.-W. Scheer: DV-gestützte Planungs- und Informationssysteme im Produktionsbereich, September 1982
- Heft 36:** A.-W. Scheer: Rationalisierungserfolge durch Einsatz der EDV - Ziel und Wirklichkeit, August 1982, Vortrag anlässlich der 3. Saarbrücker Arbeitstagung "Rationalisierung" in Saarbrücken vom 04. - 06. 10.1982

- Heft 35:** J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS-D, Konzept einer computergestützten Prüfungsumgebung, Juli 1982
- 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.