System modeling of the R&D domain through the object–process methodology: a practical tool to help R&D satisfy its customers' needs

Doron Meyersdorf[1] and Dov Dori[2]

[1] TEFEN USA, 1065 E. Hillsdale Blvd, Foster City, CA 94404, USA
[2] Faculty of Industrial Engineering and Management, Technion, Israel Institute of Technology, Haifa 32000, Israel

Research and Development is an intricate process involving a host of challenging technological frontiers. This paper presents a practical tool to help R&D personnel satisfy the needs of customers by a systematic evaluation of R&D performance. To establish a sound methodology for R&D performance evaluation, we employ the object-process analysis (OPA) part of the object–process methodology. Using OPA, we gradually expose the constituents of the R&D system, its environment and the interaction between them. Through this systematic top-down refinement, we provide a comprehensive generic view of the R&D domain. The framework presented in the paper allows the user to describe, order, and interrelate the issues that R&D managers should consider when managing, evaluating and planning R&D. As the approach addresses structure and behaviour in a unified, integrated manner, the analysis provides insights into both the static and dynamic aspects of the R&D domain and establishes a solid basis for its enterprise modeling.

1. R&D quality and productivity evaluation — an introduction

The analysis and evaluation of R&D quality and productivity is the basis for maintaining a process of continuous improvement (Dar-El and Meyersdorf, 1991; Brown and Svenson, 1988). Conventional systems for R&D productivity measurement are usually based on financial measures, while more sophisticated ones employ performance indices that account for both tangible and non-tangible factors of each R&D sub-system. R&D quality factors are rarely taken into consideration so that usually only the final product is evaluated (Sardana and Vrat, 1989; Mandakovic and Souder, 1987; Cordero, 1990; Oral et al., 1991). Normally, if any process indicators are accounted for, they are limited to issues of budget and/or schedule.

There is a consensus among R&D managers that continuous improvement is critical in the R&D process. This, in turn, is a key to customer satisfaction and delight — the ultimate goal of the entire spectrum of the product life cycle activities (Meyersdorf and Dar-El, 1993).

Current management literature has begun to incorporate total quality management (TQM) concepts into the process of product innovation (Gobeli and Brown, 1993; Eidt, 1992; Francis, 1992; Montana, 1992). In this context, TQM emphasizes the following three principles.

(1) Customer satisfaction is the most important target for a competitive organization.
(2) Continuous improvement must take place in all areas, especially during the early design stages, in which relatively small efforts can potentially reduce time-to-market and life cycle cost.

(3) Employees, including R&D staff, must be strongly involved in the improvement process.

Since most current R&D performance evaluation models lack reference to the R&D process characteristics, the first step in developing a framework for the implementation of these three product innovation TQM principles is the analysis of the R&D *process* (Rothwell, 1992; Griffin, 1992; Smith and Reinertsen, 1992).

This issue has been the main subject of a series of workshops for executives of Israeli high-tech R&D firms, held at the Technion, Israel Institute of Technology. These workshops brought together senior R&D personnel from various companies in order to address the problem of R&D process analysis and evaluation for the purpose of achieving continuous improvement (Meyersdorf, 1995).

The goal of the first workshop was to create a common basis for the research through identifying and defining the factors that influence R&D quality and productivity (Dar-El and Meyersdorf, 1991). The two main factors identified in the first workshop as crucial for R&D success were:

(1) precise characterization of the product and its verification in the market, and
(2) high quality management of the R&D process.

The characterizing factors of the R&D process, from idea generation through manufacturing to support, were identified in the second workshop.

The analysis of these findings was followed by an intensive field work, which led to the design and construction of a measurement and evaluation system, called Tree of Measures (TOM)[1] (Meyersdorf and Dar-El, 1993). The system consists of a set of readily available surrogate measures, that are expected to provide a comprehensive overview of R&D performance. One such surrogate measure for example is the Level of Project Innovation (LPI), which expresses the relative age of the technology which was implemented in the project and is calculated as

$$LPI = 1 - \frac{1}{V}\sum_i \left(V_i \frac{A_i}{P_i}\right)$$

where A_i is the age of technology i that was implemented in the project,

P_i is the age if the previous generation of the same technology i,

V_i is the investment in acquiring technology i, and

$$V = \sum_i V_i$$

The LPI index is used to benchmark the level of innovation of the project, which, in turn, provides for comparing the innovation level within the R&D department of an enterprise or among firms in the domain.

The TOM system was designed to enable a company management to identify weak elements in the R&D process, such as schedule or budget slips at one or more particular milestones of a project, unsatisfied customers, poor product performance, etc. However, some pilot TOM implementations and their analysis at the third workshop, led us to the conclusion that the cycle time for improving the R&D process is usually considerably long. It is in the order of magnitude of the 'time-to-market' of the evaluated project. Hence, the surrogate measures comprising TOM, although effective in the long and intermediate runs, are usually not available at a time that enables the project management to respond quickly enough to improvement requirement in the R&D process of the project currently under way.

This relatively long cycle time has motivated us to search for 'on-line' feedback, which refers to a particular project under examination, as opposed to a long term feedback, the purpose of which is to improve the R&D process. The short term, on-line feedback should be capable of providing adequate response and warning signs as closely as possible to 'real time'. Such prompt feedback would enable the application of corrective measures to the R&D process of the ongoing project while it is still feasible and effective. To this end, we describe briefly the object-process paradigm

[1] A glossary of specialized terms is provided at the end of the paper.

and apply it to analyse the R&D universe of interest.

This analysis has yielded the identification of a number of feedback loops with variable duration and domains. In this paper we focus on the short-term feedback cycle, as it is the one which is most likely to provide the needed timely information. A comprehensive description of the work appears in (Meyersdorf, 1995).

2. Object-based approaches to management and feedback loops

Forrester (1968) has established a quantitative theory of systems dynamics. His assertion is that within the system boundary, the basic building block is the feedback loop. The feedback loop is a path coupling decision, action level (or condition) of the system, and information, with the path returning to the decision point. Every decision is made within a feedback loop. The decision controls actions which, in turn, influence the factors triggering the decision. For example, if a budget slippage is detected, a decision is triggered to tighten the financial control over the subsequent phases of the project.

Roos et al. (1980) examine the use of influence diagrams to help understand political processes within organizations. The primary motivation for devising influence diagrams was to help consultants get a picture of the power games going on within the organization and elaborate on theoretical models of organizational functioning. The basic elements of an influence diagram are concept variables. These include the policy variables that can be manipulated, the goals or performance criteria that are to be aimed for and the intervening causes and effects. These concepts are represented as point on the plane and the causal assertions linking these concepts are represented by arrows between these points. A positive sign denotes that an increase in the concept variable at the end of the tail will lead to an increase in the variable at the head, and that a decrease leads to a decrease. A negative sign indicates the opposite movement of the head variable from that of the tail one.

Hall (1984) implemented a process-theoretic approach to postulate a descriptive theory of the natural logic of organizational policy making. The model is used to explain how the policies of a sample firm became adapted and how, together with critical events, this caused the firm to evolve in particular directions rather than in others. One tool used as part of their method is a set of the organization's Cause Maps. The maps are hard to read and interpret, and it is difficult to tell cause and effect or feedback causality relations from their examination.

Object-based approaches to the study of management issues have become an avenue of research since the object paradigm has gained popularity as a means to carry out sound analysis and design rather than being merely a programming method. The shift toward objects as the centre of analysis followed years of process oriented development of methods, such as the data flow diagram (DFD)-based ADISSA (Shoval, 1988), and methods to describe systems dynamics by an elaboration of finite state machines, notably Statecharts (Harel, 1987). Works related to our subject can be classified into two major groups: those dealing with methods for systems analysis and those that apply performance evaluation methods to the R&D domain.

Jacobson et al. (1992) has proposed the 'use case' approach to modeling behaviour of a system and Jacobson et al. (1994) shows how this approach can be applied to business situations. The use case approach defines typical scenarios and examines them in view of the system's requirements and specifications. It has become popular as a complementary means to 'pure' object-based analysis, which can potentially overcome problems in addressing system dynamics directly by many object-oriented analysis methods.

Bailetti and Callahan (1993) have applied an object-based approach to the analysis of the coordination structure of international collaborative technology arrangements and modeling of inter-company relationships. Bretschneider et al. (1991) have presented an approach that uses Predicate-Transition nets — a variant of Petri nets — and a set of rules to model decision processes. As the authors correctly point out in the summary, a major problem with their method is the synthesis of object structure and process flows. This very problem is directly addressed in the object-process methodology applied in this paper. In spite of the prevailing claim that there must be tradeoff between process modeling and structure (object) modeling, we show below that such tradeoff is not necessarily mandatory. In the object–process methodology (OPM), used in this paper, the two aspects are addressed concurrently. Benefiting from synergy, rather than invoking a tradeoff problem, structure and behaviour complement each other to provide a clear and coherent understanding of these two major system's aspects alike.

3. Object process analysis

Object-oriented analysis (OOA) provides satisfactory tools for expressing the static, structural aspects of the system under consideration. In particular, aggregation and generalization relations are explicitly expressed (Coad and Yourdon, 1991; Embley et al.,1992; Jacobson et al., 1992; Nerson, 1993; Rumbaugh et al., 1991; Shlaer and Mellor, 1992). However, most OOA methodologies are not ideal for representing the dynamic behaviour of systems. For this purpose, OOA resorts to earlier process-based approaches, usually some variant of data-flow diagrams (De Marco, 1978). These are not an integral part of the primary object model, which is static in nature. Thus, for example, object modeling technique (OMT) applies Statecharts (Harel, 1987) for modeling the dynamics aspects of the system and DFD for its functional modeling. This multiplicity of models to describe various systems aspects carries an inherent problem of consistency and integration across the various models (Dori and Goodman, 1996).

To obtain a balanced and unified representation of the R&D domain, we employ the object–process methodology (OPM) (Dori, 1995, 1996a, b). OPM treats objects and processes as two distinct types of classes of things and integrates the structural and procedural aspects of a system into a unified, coherent representation framework. The methodology also enables comprehensive complexity management through seamless, recursive and selective scaling. OPM is especially suitable for analyzing the R&D process due to its consistent, unified and balanced representation of both the static-structural and dynamic-behavioural aspects of R&D.

OPA is part of OPM which is used to analyse the system under consideration as the first phase of its development. This work makes extensive use of the scaling capability to present the R&D system in a manner that enables one to selectively focus on particular points of interest within the system.

An object process diagram (OPD) is the graphical expression of OPA. It is a visual formalism that combines ideas applied in object diagrams and data flow diagrams (DFD). As the legend of Figure 1 shows, objects and processes are represented in OPDs as rectangles and ellipses, respectively. They are linked by procedural links — effect, agent and instrument links — denoted by lines ending with an arrowhead, a black circle and a blank circle, respectively.

An effect link can point from an object, affected by a process, to that affecting process, or from the process to an affected object. An agent and an instrument of a process are enablers ! objects that enable that process, but are not affected by it. Agents are intelligent ! usually a human or a group of humans ! while instruments can be machines, tools, data, recipes, algorithms, etc. A fundamental rule in constructing an OPD is that a procedural (effect or instrument) link can only link an object to a process or vice versa. It cannot link an object to an object or a process to a process.

Structural links in an OPD express the long-term relations among objects, as well as among processes. The two most common structural links are aggregation (whole-part) and generalization (which induces inheritance), denoted by a solid and a blank triangle, respectively. A third fundamental structural relation is characterization, which is the relation between an object and its features (attributes and operations). Its graphical symbol appears in Figure 1. Scaling is an important tool for managing the complexity of real-life systems. OPD employs scaling by controlling the visibility and level of detail of the system's components.

4. Illustration: electronic Chip R&D System

To demonstrate the principles of object–process analysis we show how a set of object–process diagrams is applied to describe a Chip R&D System. All the OPD figures in this paper were drawn using OPCAT — Object–Process CASE Tool — a CASE tool that supports OPM. OPCAT has been developed by the second author and is being used in a number of large-scale analysis projects in high-tech industries.

Figure 1 shows a top view of the Chip R&D System as an object which, through the process of Interaction, communicates with the object Environment.

Figure 2 demonstrates an important feature of the object–process methodology, which is its recursive scaling capability. Each one of the three things (objects or processes) of Figure 1 is blown up such that its lower-level constituents are exposed within the blow-up frame. Thus, one can see that the Chip R&D System consists of the objects R&D Engineering Staff, Tapeout Model with its Model Performance attribute, and Chip (the actual product) with its Chip Performance attribute. The latter attribute is transferred from the Environment to the System through the Reporting process, which is part of Interaction.

Within the Chip R&D system there are two processes: (1) Chip Development and (2) Evaluation & Modification, in which Chip Performance, reported by the beta site Customer, is evaluated, and Tapeout Model is modified as needed.

Chip Development, whose agent (marked by the agent link) is the object R&D Engineering Staff, yields the Tapeout Model. The attribute Model Performance of the object Tapeout Model, along with the actual Chip Performance data are inputs to the Evaluation and Modification process.

The environment is also blown up. We see that it consists of the Fab Cleanroom, the Chip with its Chip Performance attribute, and the (beta site) Customer. The Customer is the agent for the Usage & Testing process, which yields the Chip Performance. Chip Performance, depicted as an attribute of the manufactured Chip within the Environment object, is data regarding the actual performance of the Chip. Chip Performance is obtained in field tests by the beta site Customer through the Usage & Testing process.

Finally, the Interaction process of Figure 2 is also blown up to expose its two constituent processes — Fabrication and Reporting. Fabrication uses as input the Tapeout Model and produces the Chip. Reporting takes the Chip Performance, resulting from the Usage

Figure 1. The Chip System-environment OPD.

Figure 2. The electronic Chip System of Figure 2 scaled up.

& Testing, and moves it to the Chip R&D System. Here it is used for the Evaluation & Modification process, which, in turn, affects the Tapeout Model.

Examining Figure 2, one can clearly identify the feedback loop whose components are (in this order) Tapeout Model, Fabrication, Chip, Usage & Testing, Chip Performance, Reporting, Chip Performance, Evaluation & Modification, and back to Tapeout Model, which closes the loop. The feedback loop improves the Tapeout Model through response obtained from its field testing. An interesting fact to note in the loop, as in any thread consisting of linked things, is that objects and processes alternate along the loop. This is not a coincidence. Rather, it is a fundamental outcome of the definition of processes as things that affect objects. Procedural links always link an object to a process or vice versa, but never an object to an object or a process to a process. Links of the latter type are structural and include aggregation-particulation (whole-part), characterization and generalization-specialization relations. This concludes our example of the application of the object–process methodology to a particular R&D domain, and we now turn to a discussion on a generic, domain-independent R&D world.

5. Interaction between the R&D system and its environment

Figure 3 is an instance of the universal system-environment OPD (Dori, 1995). Interaction is the process that mediates between the object System and the object Environment. This generic scheme, which is applicable to any conceivable domain, is adapted (instantiated) in the OPD of Figure 1 to the R&D domain, pertinent to a particular business (chip manufacturing).

As Figure 3 shows, the R&D world consists of the R&D system and its environment. In general, the System and the Environment exchange material and/or energy and/or information through the process of Interaction.

In the OPD of Figure 4 the Interaction process is blown up to show its two constituent processes: Requirement Generation and Realization. As the OPD shows, the instrument that enables the Realization is the R&D System. More specifically, the instrument is the object Product Lifecycle Model, shown in Figure 5 as part of the R&D System.

Figure 3. The R&D system-environment Interaction OPD.

Figure 4. The Interaction process of Figure 1 blown up.

Figure 5. R&D system blown up.

Figure 6. Realization blown up.

Figure 5 also shows the object R&D Prerequisites as part of the R&D System which is an input to the R&D Execution & Management process.

The Realization process comprises all the activities done within the Interaction by the R&D Environment to transform the Product Lifecycle Model, which is a tangible R&D output, to a sellable, operational and serviceable product. Hence, Realization consists of the processes Manufacturing, Usage and Support, as shown in Figure 6.

6. R&D Resources and Prerequisites

R&D Resources is the primary object within the R&D Prerequisites. The object R&D Resources is unfolded in Figure 7, where the fractalization relation, symbolized by a black and white triangle (see Legend of Figure 1), is a combination of the aggregation (black triangle) and generalization (white triangle) relations. The semantics of a fractal relation is that the things (objects or processes) linked to the basis of the black and white triangle are both parts and specializations of the thing linked to the top of the triangle. Thus, the objects Organizational Resources, Human Resources, Technological Resources and Managerial Resources are all fractals (parts and specializations) of the object R&D Resources.

R&D Prerequisites are blown up in Figure 8 to expose three constituent object and one process. The objects are Requirements, R&D Resources and Operational Targets. Requirements and R&D Resources are inputs to the process Requirement Engineering, which yields the object Operational Targets. This object, in turn, is the instrument enabling the R&D Execution & Management process, carried out by R&D Resources. Requirement Engineering, performed within the R&D Prerequisites, is the disciplined interpretation and translation of Requirements into Operational Targets — the instrument that enables the R&D Execution and Management process.

As shown in Figure 8, R&D Resources and Operational Targets enable the R&D Execution & Management process within the R&D

Figure 7. The fractals of the object R&D Resources unfolded.

Figure 8. The R&D feedback loop.

Prerequisites, while Product Lifecycle Model is the object resulting from this process.

A feedback loop can be easily tracked through the following thread. Starting with the Requirement Generation process, Requirements are generated. These, along with R&D Resources, are inputs to the Requirement Engineering process, which outputs Operational Targets. R&D Resources and Operational Targets enable the R&D Execution & Management process, which, in turn, outputs the Product Lifecycle Model. The latter is used for the Realization which is part of the Interaction between the R&D System and the R&D Environment.

The resulting Product, which is part of the R&D Environment, is used by R&D Customers. R&D Customers include the manufacturing, engineering maintenance, and beta site users. They all take part in the Requirement Generation process, which is enabled also by the Technology & Competition object, which acts as an instrument for the Requirements Generation process. At this point the feedback loop is closed, as new Requirements are created and gathered for the next generation of Product.

7. Discussion

The object–process methodology framework for monitoring and controlling R&D activity presented in figures 3–8 can be applied at different levels of abstraction (by a systematic scaling of the factors considered). This would provide a vehicle for applying a consistent methodology across the layers of R&D management, from those responsible for one project to those responsible for a programme area, to those responsible for the overall R&D strategy within the enterprise.

The use of this uniform approach across an organization would alleviate often-encountered problems, such as conflicts in the allocation of resources and skills to competing projects, priority settings, and problems of interpretation that can arise when moving from one level of abstraction to another (e.g., from the project level to the programme management level).

A second attractive feature of this approach is that the method considers the dynamics of the R&D process. In particular, feedback loops at different time scales are identified. It can therefore be used for evaluating the progress of a single project on the short time scale and of the adequacy of a particular R&D resource (e.g., the availability of specific skills or infrastructure) on a longer time scale. By explicitly identifying the factors in the R&D system and the interactions among them, the R&D manager is able to better control the R&D process and facilitate effective communication among the R&D team members.

An R&D evaluation system inspired by our approach has been applied as part of a research (Meyersdorf, 1995) in five Israeli high-tech firms in the electronics and communication industry. The feedback loops identified here were used to distinguish among the various project phases and the interpretation of the evaluation results. For example, a problem was identified in the structure of a particular project in one of the firms participating in the research. To solve the problem, the short term feedback loop initiated corrective measures to improve the structure of that particular project, while the long term feedback loop was effective in establishing standards and procedures within the firm for generating projects with improved structure.

To suit its specific needs, an organization would need to develop a tool adapted to its circumstances, using the generic Object–Process CAse Tool (OPCAT) presented in this work, as a foundation. While it is not a very easy task, and would require spending intellectual efforts of senior R&D personnel, the benefits of adopting the method and using it systematically would certainly outweigh this investment. This is primarily due to the fact that the costs that can be avoided in future development stages are known to increase exponentially as we proceed from conceptual prototyping to detailed design to prototype manufacturing. A side benefit would be the detailed R&D enterprise modeling resulting from this activity, which provides insights into details of the R&D operation that had never been considered before.

8. Summary

A comprehensive analysis of the R&D domain has been introduced using OPCAT — the Object–Process CASE Tool, which supports the implementation of the object–process methodology (OPM).

The concurrent gradual and selective exposure of the structural and behavioural aspects of the R&D domain was enabled by the various scaling tools offered within OPCAT. Scaling makes the complexity management of this system a tractable task and provides insights into the dynamics of the system and its feedback loop. The results of this analysis have been instrumental in devising an evaluation a methodology for R&D quality and productivity. The approach has been successfully applied within several Israeli High-Tech firms to generate an overall measurement and evaluation system for R&D quality and productivity.

This work demonstrates the applicability of OPM for specifying structural and behavioural semantics of complex systems at the enterprise level within a unifying frame of reference. The framework proposed in this work, if properly implemented and adapted to the particular circumstances of an organization, could improve the quality of R&D management by ensuring that all relevant factors receive due consideration at the proper time. OPM can serve as a sound basis for enterprise modeling and an effective tool for business process reengineering of R&D units as well as other functions within the enterprise.

Acknowledgments

The authors wish to thank Professor Ezey Dar-El of the Faculty of Industrial Engineering and Management, Technion, Israel Institute of Technology, for proposing the research topic of this paper and guiding the first author throughout his D.Sc. dissertation. Thanks also go to the anonymous referees for their valuable comments. The work of the second author was supported in part by the Fund for Advancement of Research at the Technion.

Glossary of abbreviated terms

CASE — Computer Aided Software Engineering
DFD — Data Flow Diagram
LPI — Level of Project Innovation
OMT — Object Modeling Technique
OOA — Object-Oriented Analysis
OPA — Object–Process Analysis
OPCAT® — Object–Process CASE Tool
OPD — Object–Process Diagram
OPM — Object–Process Methodology
TOM — Tree of Measures
TQM — Total Quality Management

References

Bailetti, A.J. and Callahan, J.R. (1993) The coordination structure of international collaborative technology arrangements. *R&D Management*, **23**, 2 April, 129–146.

Brown, M.G. and Svenson, R.A. (1988) Measuring R&D productivity. *Research Technology Management*, **31**, 4, 11–15.

Bretschneider, F., Kopf, C. and Zolg, M. (1991) Process control for large complex systems. In Nelson, B.L. , Kenton, W.D. and Clark, G.M. (eds.), *Proceedings of the 1991 Winter Simulation Conference*.

Coad, P. and Yourdon, E. (1991) *Object Oriented Analysis*. Prentice Hall, Englewood Cliffs NJ: (2nd Ed.)

Cordero, R. (1990) The measurement of innovation performance in the firm: an overview. *Research Policy*, **19**, 3, 185–192.

Dar-El, E.M., and Meyersdorf, D. (1991) Raising R&D productivity - one more look! In Sumanth et al. (eds.)*Productivity Management Frontiers-II*, Vol. 1. Amsterdam; Elsevier Science pp. 103–109.

De Marco, T. (1978) *Structured Analysis and System Specification*. New York: Yourdon

Dori, D. (1995) Object–Process Analysis: maintaining the balance between system structure and behaviour. *Journal of Logic and Computation*, **5**, 2, 227–249.

Dori, D. (1996a) Unifying system structure and behaviour through Object–Process Analysis. *Journal of Object-Oriented Analysis*, July-August, 66–73.

Dori, D. (1996b) Object–Process Analysis of computer integrated manufacturing documentation and inspection. *International Journal of Computer Integrated Manufacturing*, **9**, 5, 339–353.

Dori, D. and Goodman, M. (1996a) On bridging the analysis-design and structure-behavior Grand Canyons with object paradigms. *Report on Object Analysis and Design*. **2**, 5, 25–35.

Dori, D. and Goodman, M. (1996b) From Object–Process Analysis to Object–Process Design. *Annals of Software Engineering,* Vol. 2. Object-Oriented Software Engineering: Foundations and Techniques, pp. 25-20.

Eidt, C.M. Jr. (1992) Applying quality to R&D means 'learn-as-you-go'. *Research Technology Management*, **7**, 8, 24–31.

Embley, D.W., Kurtz, B.D. and Woodfield, S.N. (1992) *Object Oriented Systems Analysis*. Englewood Cliffs, NJ: Prentice Hall.

Forrester, J.W. (1968) *Principles of Systems*. Cambridge, MA: Wright Allen.

Francis, P.H. (1992) Putting quality into the R&D process. *Research Technology Management*, **7**, 8, 16–23.

Gobeli, D.H. and Brown, D.J. (1993) Improving the process of product innovation. *Research Technology Management*, **3**, 4, 38–44.

Griffin, A. (1992) Evaluating QFD's use in US firms as a process for developing products. *J. Prod. Innov. Manag*. **9**, 171–187.

Hall, R.I.(1984) The natural logic of natural policy making: its implications for the survival of an organization. *Management Science*, **30**, 8, 905–927.

Harel, D. (1987) State charts: a visual formalism for complex systems. *Sci. Comput. Program*, **8**, 231–274.

Jacobson, I., Christersen, M., Johnson, P. and Overgaard, G. (1992) *Object Oriented Software Engineering - A Use Case Driven Approach*: Reading, MA: Addison Wesley.

Jacobson, I., Ericsson, M., and Jacobson, A. (1994) *The Object Advantage: Business Process Reengineering with Object Technology.* : Reading, MA: Addison Wesley.

Mandakovic, T. and Souder, W.E. (1987) A model for measuring R&D productivity. In Sumanth, D. et al. (eds.), *Productivity Management Frontiers-I*, **Vol. 1**. Amsterdam: Elsevier Science, pp. 139–146.

Meyersdorf, D. (1995) Research and development quality and productivity: measurement and improvement process. D.Sc. Thesis, Faculty of Industrial Engineering and Management, Technion, Israel Institute of Technology (in Hebrew).

Meyersdorf, D. and Dar-El, E.M. (1993) A new approach for R&D productivity measurement. In Sumanth, D. et al. (eds.),*Productivity and Quality Management Frontiers-IV*, Vol. 1. Industrial Engineering and Management Press, 43–52.

Montana, J. A. (1992) If it isn't perfect, make it better. *Research Technology Management*, **6**, 7, 38–41.

Nerson, J. M. (1993) Applying Object Oriented Analysis and design. *Communications of the ACM*, **35**, 9, 63–74.

Oral, M., Kettani, O. and Lang, P. (1991) A methodology for collective evaluation and selection of industrial R&D projects. *Management Science*, **37**, 7, 871–885.

Roos, L.L. Jr. and Hall, R.I. (1980) Influence diagrams and organizational power. *Administrative Science Quarterly*, **25**, 57-71.

Rothwell, R. (1992) Successful industrial innovation: critical factors for the 1990s. *R&D Management*, **22**, 3, 221–239.

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. and Lorensen, W. (1991) *Object-Oriented Modeling and Design*. Englewood Cliffs, NJ: Prentice Hall

Sardana, G. D. and Vrat, P. (1989) Productivity measurement of applied industrial research in an organization: a conceptual framework. Sumanth, D. et al. (eds.)*Productivity Management Frontiers-II*. Amsterdam, Elsevier Science, pp. 47–54.

Shoval, P. (1988) ADISSA: Architectural design of information systems based on structured analysis. *Information Systems* **13**, 193–210.

Shlaer, S. and Mellor, S.J. (1992) *Object Lifecycles, Modeling the World in States*. Englewood Cliffs, NJ: Yourdon, PTR Prentice Hall.

Smith, P.G. and Reinertsen, D.G. (1992) Shortening the product development cycle. *Research Technology Management*, **5**, 6, 44–49.

