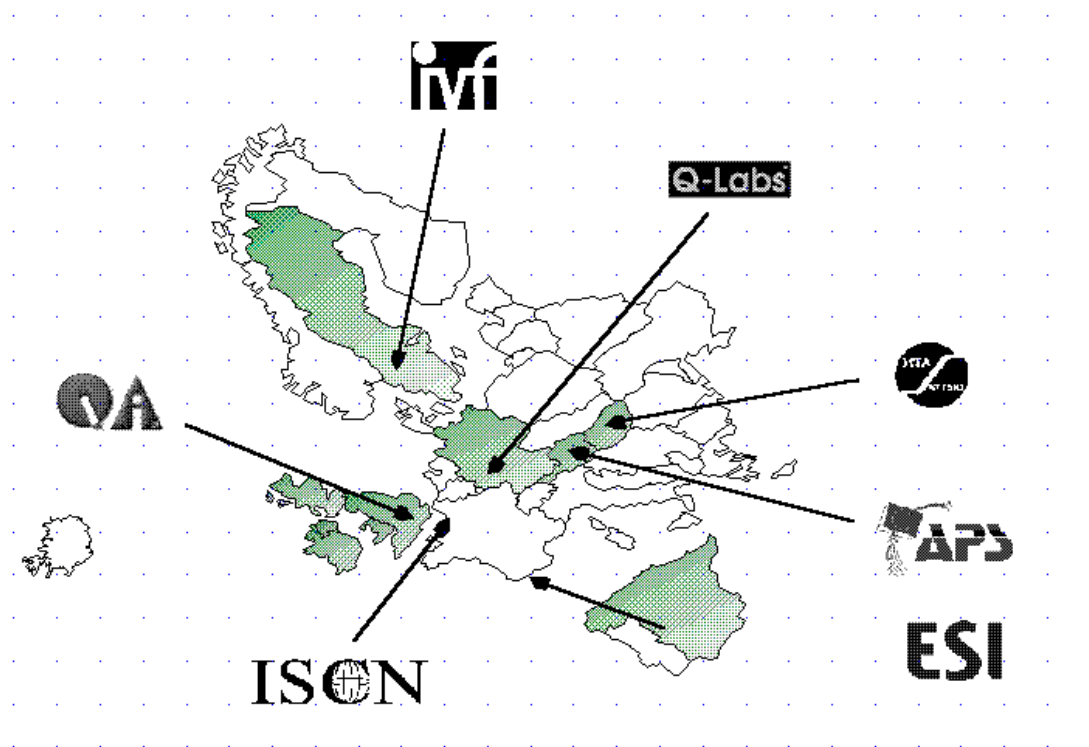


# Practical Improvement of Software Processes and Products

1997 Theme - Cooperation and Business  
Opportunities in the IT Field



*Strategic Alliances and Business Opportunities*

*Learning Strategies*

*Information Systems for Effective Process Improvement Collaboration*

*Exchange of User Experience in Process Improvement*

Preface .....	v
Conference Programme Committee .....	viii
Conference Sponsors .....	viii
Local Organising Committee .....	viii
Conference Session Chairs .....	ix

## Conference Programme

0800 *Registration of Participants - Music in the Conference Hall*

0900 *Official Opening and Wellcome*

### Session 1 (Day 1): Initial Key Talks

0915 *Business and Cooperation Opportunities in Central and Eastern Europe* 1-1  
Klaus Woelken, CEC DG3, Brussels, Belgium

0945 *The State of Software Best Practice in Central and Eastern Europe* 1-2  
M. Biró, MTA SZTAKI, Hungary (moderator);  
J. Gorski, Technical University of Gdansk, Poland;  
Yu. G Stoyan, A.F.Loyko, M.V.Novozhilova, National Academy of Sciences of Ukraine;  
I.Socol, SIVCO, Romania (INSPIRE INCO-Copernicus Project);  
R.Vajde Horvat, I.Rozman, J.Györkös, University of Maribor, Slovenia

1030 *Coffee Break in Exhibition Area*

### Session 2 (Day 1): Strategic Alliances and Business Opportunities

1045 *ISCN - A Networking Concept for Transnational Co-operation in Improvement and Learning Strategies* 2-1  
Richard Messnarz, ISCN Ltd., Ireland  
Micheal Mac an Airchinnigh, Trinity College, Dublin, Ireland  
Miklos Biro, Sztaki, Budapest, Hungary

1115 *Learning to Improve with BOOTCHECK* 2-15  
Amor Dominguez, Izaskun Doiz, ESI, Spain

1145 *Success Factors in Technology Transfer: Quick State-of-the-art Mapping* 2-17  
Christophe Debou, Anders Gustavsson, Q-Labs GmbH, Kaiserslautern

1215 *A Concept for Managing East - West Outsourcing Business* 2-25  
Miklos Biro, Tibor Remszö, Sztaki, Hungary  
Christophe Debou, Q-Labs GmbH, Kaiserslautern, Germany  
Alec Dorling, IVF, Sweden  
Richard Messnarz, ISCN, Austria and Ireland

1245 *Business Lunch*

---

**Session 3 (Day 1): Learning Strategies**

- 1400 *Learning to Improve - the Essential Ingredients* 3-1  
Håkan Wickberg, Alec Dorling, IVF, Gothenburg
- 1430 *Best Regional Technology Transfer - A Learning Concept for Innovation Management Organisations* 3-13  
Bernhard Posch, Christine Stöckler, APS, Graz, Austria  
Richard Messnarz, ISCN, Dublin, Ireland
- 1500 *PICO - A Training Concept for Creating a Learning Culture and Initiative in IT Organisations* 3-23  
Richard Messnarz, editor of joint article, ISCN Ltd., Dublin, Ireland  
The PICO Team <http://www.iscn.ie/projects/pico/>
- 1530 *PICO's Multi-Method Framework Tool-Set for Self-Assessment* 3-42  
Susanne Lanzerstorfer, Hans Scherzer, APAC, Vienna
- 1600 *Coffee Break in Exhibition Area*

**Session 4 (Day 1): Open Forum**

- 1615 *EXCUSES* 4-1  
*An Experiment for Use Cases in Capturing User Expectations in Software Development Projects*  
Alessia Billi, Franco Correrini, Sodalìa SpA, Trento, Italy
- 1645 *Experience With Teleworking and Outsourcing Management* 4-10  
Martin Hollinetz, TPI : Technologies - Projects - Integration, Graz, Austria
- 1715 *Process Improvement Through AMI (PITA) (a Process Improvement Experiment under ESSI)* 4-11  
Vassilis Kopanas, INTRACOM S.A., Peania, Attika, Greece
- 1745 *Analysis of the Performance of Software Businesses with Artificial Neural Networks* 4-22  
V. H. Haase, University of Technology Graz, Austria
- 2030 *Social Reception with the General Secretary of the Hungarian Academy of Sciences*

---

**Session 5 (Day 2): Information Systems for Effective Process Improvement Collaboration**

- 0900 *Distance Training and Co-operative Work through Internet Video-Conferencing* 4-1  
 G. Bazzana, E. Fagnoni, G. Rumi, ONION Communications-Technologies-Consulting, Brescia, Italy  
 J. Boegh, DELTA Danish Electronics, Light & Acoustics, Hørsholm, Denmark  
 E. Van Veenendaal, KEMA International, ET Arnhem and Eindhoven University, The Netherlands  
 S. Geyres, SMC International-Division PSTI Evaluation, Toulouse Labège, France
- 0930 *A European Software Good Practice Repository* 4-10  
 Giuseppe Satriani, European Software Institute, Bilbao-Spain
- 1000 *NQA - Network based Quality Assurance system* 4-11  
 Richard Messnarz, ISCN Ltd., Dublin, Ireland  
 Robert Stubenrauch, JOANNEUM RESEARCH, Graz, Austria

1030 *Coffee Break in Exhibition Area*

**Session 6 (Day 2): Methods and Experience**

- 1045 *Contract Management with Euromethod* 6-1  
 Alfred E. Helmerich, FAST, München, Germany
- 1115 *Process Improvement Experiment at MemoLuX* 6-18  
 Miklós Biró, Éva Feuer, MTA SZTAKI, Budapest, Hungary  
 Janos Ivanios, Memolux, Budapest, Hungary
- 1145 *Configuration Management for safe Delivery of Software Systems* 6-30  
 Martin Brett, Robert Bosch GmbH, Germany, FV/PLI3
- 1215 *SPI: an experience report from GSM development* 6-45  
 S. Di Muro, S. Humml, A. Lora, S. Scotto di Vettimo, Italtel SpA, Cassina de Pecchi (Milano)-Italy  
 G. Bazzana, G. Rumi, ONION Communication-Technologies-Consulting, Brescia-Italy
- 1245 *Business Lunch*
- 1400 *MODAL: the CEGELEC Software Process Memory* 6-65  
 Yves Benoit, CEGELEC, Clamart
- 1430 *Evaluating Software Product Quality* 6-73  
 Jørgen Bøegh, DELTA Software Engineering, Hørsholm, Denmark
- 1500 *The Reliability of ISO/IEC PDTR 15504 Assessments* 6-86  
 J.M. Simon, AQT, France  
 K. El Emam, Fraunhofer IESE, Germany  
 S. Rousseau, E. Jaquet, Sanofi, France  
 F. Babey, AFNOR, France

- 1530 *Process Improvement for Better Software Quality* 6-101  
Katalin Balla, Tamás Langer, IQSOFT Ltd., Budapest, Hungary
- 1600 *Building Resource and Quality Management Models for Software Inspections* 6-115  
Lionel C. Briand, Oliver Laitenberger, Isabella Wiczorek, Fraunhofer IESE,  
Kaiserslautern, Germany
- 1630 ***Coffee Break in Exhibition Area***
- 1645 *Improvements through Configuration & Change Management and Test of Software for a* 6-132  
*Test Path Generator*  
Marcus Begemann, Harry Debler, DTK GmbH, Palmaille 82, D-22767 Hamburg

**Session 7 (Day 2): Panel Discussion**

- 1715 Panel Chair: Dr Miklos Biro 7-1  
Experts Panel : Christophe Debou, Amor Dominguez, Alec Dorling, Richard Messnarz,  
Bernhard Posch, Hakan Wickberg
- Titel: Key Success Factors in SPI and Collaboration Opportunities with Central and  
Eastern Europe*
- 1830 Closing of the Conference

# Preface

Software Process Improvement is much more than just using standard processes and checklists to evaluate the existence of practices. The success largely relies on

- how the goals of process improvement can be aligned with the organisation's strategic business goals
- how the people in the organisation are motivated to work on their own effectiveness within efficient organisational processes
- how improvement experience can be leveraged and multiplied
- how a critical mass is reached after a process improvement champion started to convince top management
- how the technical manager can learn the business manager language to be able to convince top management for creating funds and support for software process improvement
- how efficiently processes can be executed by effective cooperation and team work based on synergies and win-win
- etc.

There are even social aspects. In the field of innovation management (not just software innovation) one of the key issues is "the skills of people" which largely decides the innovation of organisations. Processes and standards are then a standard framework for these innovative people to express their new ways of work in a style that can be understood and executed by other people in the organisation.

The ESI & ISCN 1997 Conference thus concentrates on issues like

- Business and Co-operation Strategies
- Learning Strategies
- Information Systems for Co-operation and Learning
- Industrial Software Process Improvement Experience

and offers a set of tutorials addressing a holistic view on process improvement, taking into account business, people, social, and technical factors.

It further is launching the first results of PICO, a training initiative under the EU Life Long Learning Programme. <http://www.iscn.ie/projects/pico>

This year's ESI & ISCN conference raised the topic of East-West cooperation addressing issues like

- assessing and enhancing the maturity of central and East European software firms
- enabling mutually beneficial cooperation between West and East firms in the IT field
- identifying investment opportunities

- creating an outsourcing bridge to Eastern Europe

The largest cost factor in development is the effort of people, while the technology and communication cost are decreasing. Why should Europe look into far away countries, if Eastern Europe could be a low cost resource to start competing with aggressive price policies from other continents.

Of course, the opening of Central and Eastern Europe is also a social and political issue and must be done very carefully taking into account the different arguments, but it seems necessary to ensure Europe's position in this business by early enough establishment of a network of qualified partners in Central and Eastern Europe.

*For East European co-operation please contact*

Dr Miklos Biro  
MTA Sztaki  
miklos.biro@sztaki.hu

*For information concerning the ESI support for this initiative please contact*

Mrs Amor Dominguez  
ESI  
amor.dominguez@esi.es

*For information about the PICO training initiative, training, and consulting please contact*

Dr Richard Messnarz  
ISCN  
rmess@iscn.ie

*For training and learning strategies please contact*

Mr Alec Dorling  
QAI  
dorling@qai.u-net.com

*For cooperation in PI initiatives with Sweden and Scandinavia please contact*

Mr Hakan Wickberg  
IVF  
hw@ivf.se

*For cooperation in technology transfer, innovation management, and learning initiatives with Austria please contact*

Mr Bernhard Posch  
APS  
Posch@aps.tu-graz.ac.at

*For cooperation in process improvement research and consulting please contact*

Mr Christophe Debou  
Q-Labs  
cd@q-labs.de

For information about east-West EU funding potentials please contact

Dr Klaus Woelken  
CEC DG 3  
natascha.genschow@dg3.cec.be

We invite you to join the process improvement discussions and presentations at this conference to actively help establishing a culture in which organisations work together, exchange know how, and share effort, knowledge and risk to work on process improvement problems.

Dr Richard Messnarz  
Director, ISCN Ltd.  
Co-ordinator of the Conference Board



## **Conference Programme Committee**

Miklos Biro, Sztaki, Hungary  
Christophe Debou, Q-Labs, Germany  
Amor Dominguez, ESI, Spain  
Alec Dorling, QAI, UK  
Khaaed El Emam, IESE, Germany  
Richard Messnarz, ISCN Ltd., Ireland  
Bernhard Posch, APS, Austria  
Hakan Wickberg, IVF, Sweden

## **Conference Sponsors**

- International Software Consulting (Collaborative) Network, ISCN Ltd., Dublin, Ireland
- Quality Assurance Institute - Europe, QAI, UK
- Swedish Institute for Industrial Research and Cooperation, IVF, Gothenburg, Sweden

## **Local Organising Committee**

- Gustav Hencsey, Scope Ltd., Budapest Hungary
- Miklos Biro, Sztaki, Budapest Hungary

Email: [hencsey@sztaki.hu](mailto:hencsey@sztaki.hu)

# Conference Session Chairs

## Initial Key Talks

Richard Messnarz, Technical Director, ISCN

## Strategic Alliances and Business Opportunities

Hakan Wickberg, IVF, Sweden

## Learning Strategies

Amor Dominguez, ESI, Spain

## Open Forum

Micheal Mac an Airchinnigh, President, ISCN

## Information Systems for Effective Process Improvement Collaboration

Christophe Debou, Q-Labs, Germany

## Industrial Experience

Alec Dorling, SPI lead expert, IVF and Director, QAI - Europe

Tibor Remszö, IT Division Head, Sztaki

## Panel about Cooperation Opportunities

Coordinator: Dr Miklos Biro, Sztaki, Hungary

# **Business And Cooperation Opportunities in Central and Eastern Europe**

Dr Klaus Woelken  
European Commission  
Directorate General III/F-2  
Brussels, Belgium

Dr Woelken's talk will address EU funding options supporting the establishment of collaboration between West and East companies in partner consortia jointly working on research, industrial applications, and process improvement.

To obtain the slides of this talk and to receive further information material and support please contact:

Mrs Natascha Genschow  
European Commission  
Directorate General III/F-2  
N105 05/90  
200, rue de la Loi  
B-1049 Brussels, Belgium  
Tel. +32-2-296.71.39  
Fax. +32-2-296.83.97  
E-mail: [Natascha.Genschow@dg3.cec.be](mailto:Natascha.Genschow@dg3.cec.be)

# The State of Software Best Practice in Central and Eastern Europe

M. Biró, MTA SZTAKI, Hungary (moderator);  
J. Gorski, Technical University of Gdansk, Poland;  
Yu. G Stoyan, A.F.Loyko, M.V.Novozhilova, National Academy of  
Sciences of Ukraine;  
I.Socol, SIVECO, Romania (INSPIRE INCO-Copernicus Project);  
R.Vajde Horvat, I.Rozman, J.Györkös, University of Maribor, Slovenia

## Introduction

Increasing the competitiveness of the industry in general and the software industry in particular is a cornerstone of the survival and growth of all economies including those of the European Union (EU) and those of Central and Eastern Europe (CEE). Is it possible to increase competitiveness in these two regions simultaneously by way of mutually fruitful cooperations?

The benefits for Central and Eastern European partners would come from the progress on the experience curve which has a definitely higher importance than the actual immediate financial benefit. This issue will be analyzed in more detail in a forthcoming talk.

The benefits for European Union partners are the following for example: highly educated workforce, cultural proximity, relatively low cost in CEE. Nevertheless, cooperation is difficult to initiate because of perceived threats arising from the former neglect of the development of a quality culture.

The objective of this key talk is to highlight the facts and the trends of software best practice application and of its supporting environment: the information technology market in Central and Eastern Europe.

The type of information available from different countries will appear to be more or less non-uniform which is natural at this stage. However, this issue has already been addressed in the following way at the Policy and Strategy Formulation Panel held in June 1997 in Budapest preparing the 3rd EU-CEEC Information Society Forum:

"Information based on reliable data is of utmost importance for the formulation of information society policy and its effective and efficient implementation. For the time being governments have to work with an insufficient set of data which are currently available mostly provided by private commercial organizations. However, the existing information gap has to be closed in order to put policy making on a solid basis. Therefore the governments are invited to increase their efforts in the collection of reliable data relevant to the Information society, and to support any activities in the harmonization of statistics on order to guarantee their international comparison."

## INSPIRE and European Union funded Software Process Improvement projects in CEE

There are several European Union funded Software Process Improvement projects running fully or only partly in CEE. The PASS ESSI PIE project for example, has a separate presentation at the ESI-ISCN'97 Conference. Other projects, like the ColorPIE ESSI ESBNET project, are in the proposal stage at the time of writing this paper.

INSPIRE (Initiative for Software Process Improvement in Regions Exterieures) is a project that is being funded by the European Commission through the Copernicus Programme of the European Commission. It has the objective of granting access to the experience and knowledge of software process improvement methods currently available in Western Europe, to small and medium sized software development organisations (SMSD) in Eastern and Central Europe.

INSPIRE is designed to help SMSDs overcome barriers to successful software process improvement actions. The four principal objectives are:

1. Raising awareness of Software Process Improvement benefits among decision makers and change agents in SMSDs
2. Educating SMSD managers and staff in practical software process improvement skills
3. Helping SMSDs to maintain momentum in carrying through their improvement plans
4. Bringing participating SMSDs together in regional Work Groups.

INSPIRE has participants from Poland, Estonia, Hungary, and Romania.

### Hungary

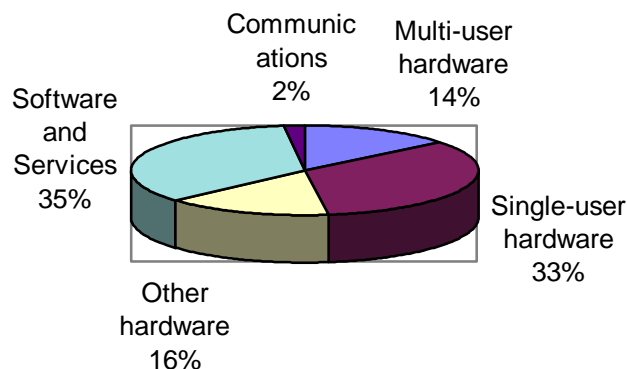
A more detailed analysis of the Hungarian information technology market already appeared in the proceedings of the ISCN'96 conference. A few of the figures are repeated below for convenience.

#### *Size of the Hungarian Information Technology Market (1993)*

**US\$ 610.4M**

#### *Relative Size of the Submarkets*

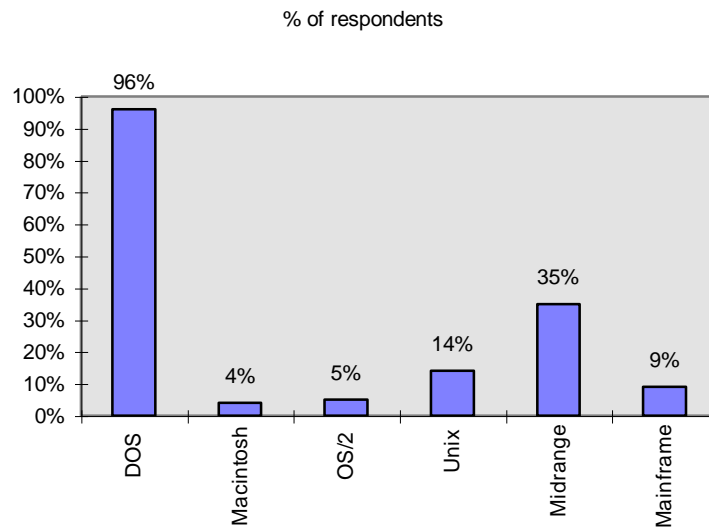
MBIRO. 1.



Data source: IDC Ltd.

**Hardware Installations (1994)**

MBIRO. 2.



Data source: Deloitte & Touche, IDOM

The dominance of DOS based PC's is primarily due to their relatively low price, even though their original market penetration was determined by the past CoCom restrictions on the transfer of high technology.

**Quality Awareness in Hungary**

The general Hungarian Quality Scene is best characterized by the increasing number of ISO 9000 which grew from very few at the beginning of the 90's to over 500 today.

Up to now however, there are few software development organizations which have achieved ISO 9000 certification including the Informatics and Systems and Control departments of MTA SZTAKI.

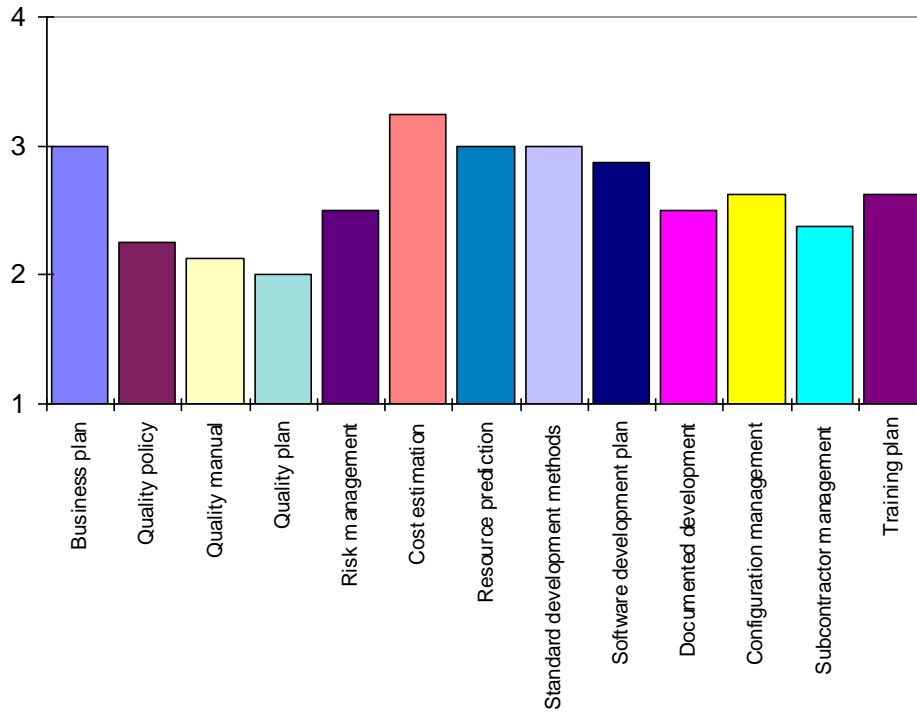
As far as the capability maturity of software development firms is concerned, we assessed some software companies with the help of the BOOTSTRAP software process assessment methodology. According to our assessments, the maturity levels of assessed software producing units were between 1.25 and 2.75.

In order to getting a broader picture of the quality awareness of the Hungarian software industry, we created a short questionnaire. Companies were asked to reply voluntarily and anonymously. 88 percent of respondents knew about ISO 9000 standards, 38 percent knew the BOOTSTRAP methodology. A few have heard about CMM, SPICE and TickIT methodologies and standards, other methodologies were not well known. The demand or requirement for formal certification has not become obvious yet. The majority of respondents (88 %) does not or rarely requires formal certification to ISO 9000 from their subcontractors. Usually they are not required to have formal certification as a subcontractor, either. At the same time, the majority of respondents feel the need for the formal certification of their quality management system. Some of them are planning a certification or are currently undergoing one. The initiations of quality management are present almost everywhere.

The second half of the questionnaire was directed towards the specific areas of quality management. Questions were asked about the level at which processes of a specific area are accomplished or the existence and level of detail of certain documents. Answers could be chosen from a range of four levels. Results were of course not precise enough to conclude at some general maturity level, but are satisfactory to make comparisons between awareness in the various quality areas. The following chart shows the results of this part of the questionnaire. Level 1 means, the process or task is not performed or the documentation does

not exist, level 4 means that the process is fully performed and the documentation is complete.

MBIRO. 3.

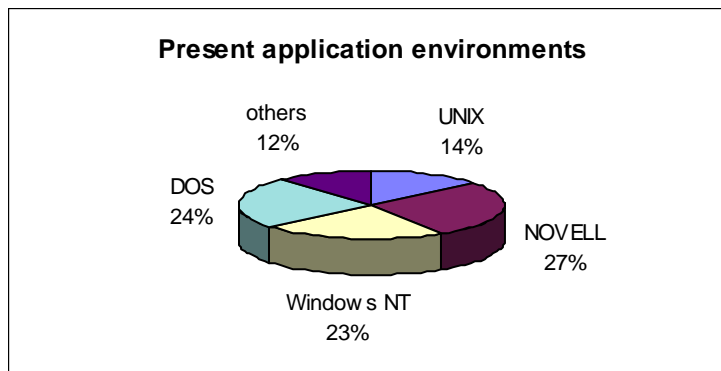


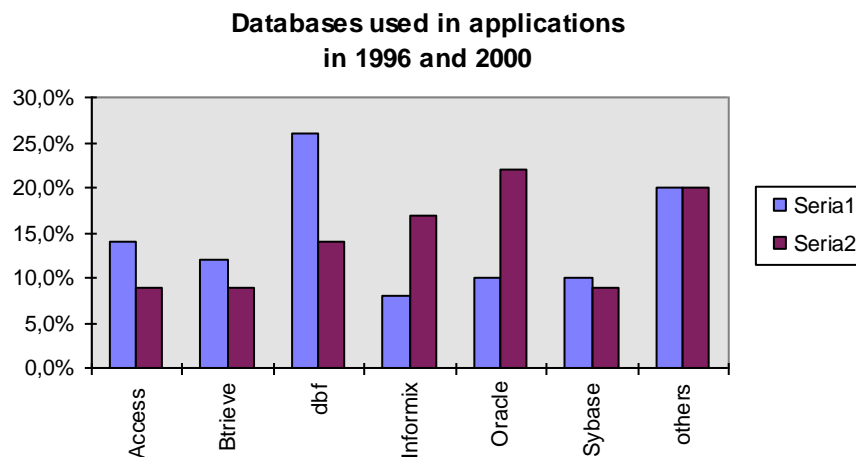
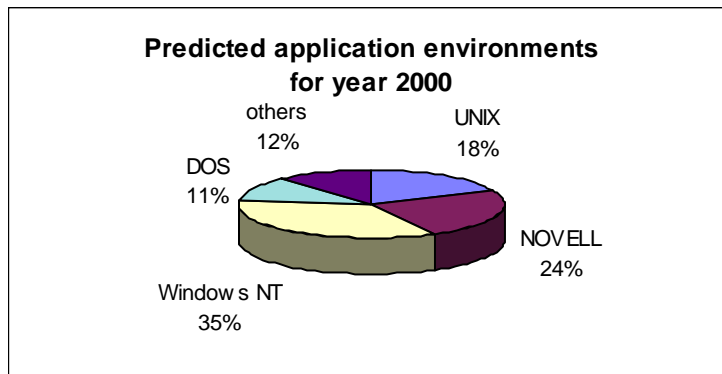
## Poland

### *Polish IT market in general:*

present value: 2 000 millions USD  
 estimated value for year 2000: 6 000 millions USD

### *Infrastructure for applications (now and predicted for year 2000):*





***Trends towards year 2000:***

- increased use of Windows NT (decreasing the share of DOS based applications)
- increased use of GUIs
- increased use of Oracle and Informix (for new applications, preserving the existing applications)

***Attempt to „sell” SPI in the Polish market:***

Within the period March-April 1997 there was a market study towards identification of needs for improvement of software processes in Polish institutions. We have selected 60 institutions to contact. We have concentrated on large institutions, assuming that small and middle enterprises do not have enough capital to invest in technology improvements (in Poland we do not have yet well defined schemes to support small and middle business in technology advances). The institutions have been divided into 3 categories:

1. suppliers of IT infrastructure (hardware + software) and system integrators for end users, e.g. HP, Oracle, Unisys, Computerland,...
2. software developers, e.g. CSBI, PROKOM, POLSOFT,...
3. software clients (sometimes with a large software department), e.g. banks, insurance companies, administration, ...

Those institutions have been offered a range of assessment and improvement activities aiming at their software processes.

The offer covered the following areas:

1. Requirements engineering



2. Project management
3. Quality Assurance
4. Object-oriented development
5. Information security
6. Safety of computer control systems

We have defined 15 themes for co-operation which have been included into our catalogue and offered 15 courses. Of course, we have also declared that we are open to discuss specific themes if the needs are outside of what has been offered in the catalogue.

The plan of the study was as follows:

1. sending the offer to an institution
2. checking (by telephone contact) if the institution is interested in further cooperation
3. if YES, sending a full catalogue describing possibilities of cooperation in detail, and
4. starting talks about possible projects (by telephone, fax, e-mail and visits)
5. if this is successful, the institution becomes an actual client

According to the above steps we can classify the institutions as:

- a) uninterested in co-operation
- b) interested in co-operation, but not yet able to identify a specific project
- c) interested in a project - talks are in progress
- d) actual clients

The campaign has been launched at the beginning of March '97.

The results of the marketing activities, after six weeks, were as follows:

	suppliers	developers	clients
uninterested	16	12	6
interested	2	4	13
in progress	0	2	1
actual clients	0	2	2
<b>TOTAL</b>	<b>18</b>	<b>20</b>	<b>22</b>

For the group of clients we have distinguished three subgroups:

	banks	administr.	others
uninterested	4	1	1
interested	10	2	1
in progress	0	0	1
actual clients	2	0	0
<b>TOTAL</b>	<b>16</b>	<b>3</b>	<b>3</b>

### ***Assessment of the results***

The results show that there is interest in software technology transfer and process improvement. An interesting observation is that the client organizations seem to be more aware of this need than e.g. software developing companies. From our contacts we see that there is a very visible feeling that something must be changed and that the things can not be continued as they used to. From other side, this feeling is not directly translated to a deeper understanding of what should be changed and where the investments should go. So the market

needs more awareness building actions, more success stories, more demonstration of positive examples.

Almost all institutions which responded positively declared interest in courses and training. Again this shows the need to continue the awareness building activities. It also gives a chance that such training seminars can lead to more concrete co-operation in the future.

It is estimated that the software market in Poland will increase three times till the end of this century. This means that the needs for technology improvement (which are already here) and will rapidly grow in the next years.

## **Romania**

More than 90% of the Romanian software companies are private ones; it still exist state owned companies ( no more than 10 in the whole country) as the Institute of Research in Informatics. In the state companies the number of persons are rather big - a number between 300-400 specialists. Comparing with this , the software private companies are usual with no more than 10-20 specialists , up to 50 ones;

- all the big computer companies ( specially hardware, but also Oracle, Microsoft, Progress, Informix, a.s.o) are present on Romanian market;
- communication system is in progress for modernisation;
- concerning human resources, in 1995 there were a number of 116000 programmers ( in 1995) in Romania;
- comparing with India , for a population which is 35 times smaller in Romania, the number of programmers is only 8 times smaller in Romania than in India;
- comparing with Russia, for a population which is 10 times smaller in Romania, the number of programmers for our country is only 2 times less than in Russia;
- in Romanian Universities are graduated 600 (six hundred) high software computer specialists each year. Romanian students are awarded every year at the international contests on software programming.

According to the National Commission for Informatics, in the next years, one of the trends will be to increase the volume of Romanian software market (the minimum annual rate proposed it will be 150%, starting from 1998). The structure of the Romanian software market will be also changed, a special importance will be given to the software applications and software services (and not to the software systems as it is now).

The research, educational and economical institutes will be linked to the international communication buses, to allow the Romanian participation to the worldwide informational flow.

It is provided to bring new technologies in software industry (actually, a big part of the market is owned by applications developed with FOXPRO under DOS environment).

To raise and grant the software quality, a big importance will be given to the ISO 9000 certification. In order to stimulate the ISO 9000 certification, the software companies ISO9000 certificated will be provided with a lot of facilities, as reducing or completely removing the taxes.

Software Technological Parks will be created / implemented and new standards will be developed for software engineering (mandatory and optional standards, recommendations, Romanian keyboard, Romanian IT terminology).

The public administration activities will be fully informatized, to improve the services and to simplify the procedures and administrative practices.

Training is another important problem. It is provided to change the structure of the specialized faculties and to add new domains, as project management, marketing and quality assurance for IT industry. Strong cooperation will be established between education and software industry, through achieving practical experience by working in the software industry field. The student's scholarships will be free of taxes.

A lot of unemployed people will be absorbed by a colateral industry, referring to the data production. A big number of software companies will be involved in this activity and it is estimated that about 20 million USD will be obtained from exporting data production.

## Slovenia

The PROCESSUS project (Assessment and introduction of a quality system) has been initiated in 1994 by the cooperation of the research group Laboratory for Informatics at the University of Maribor and by the Slovenian local industry (11 organizations). The financial support of the Ministry of Science and Technology of Slovenia gives to the project the national attribute. The partners from the local industry can be divided in three groups:

**Group A:** *Large information organizations* (extensive software development activities, consulting and training activities, interest for selling the PROCESSUS methodology),

**Group B:** *IT departments within large enterprises* (software support of main activities of organization like pharmacy, insurance, banking, etc.),

**Group C:** *Independent software companies* (software development activities, small organizations (up to 20 employees), interest for certification).

In the project two issues are intertwined:

- the **research issue** involving the development of methodology which can be applied for the wide range of potential organizations seeking for quality of the software process,
- the **implementation issue** involving the use of the methodology to introduce and maintain quality systems in participating organizations.

The greatest effort within the first phase of the project was done to set-up the PROCESSUS methodology. For this purpose the existing models and their usability were studied. The *Capability Maturity Model (CMM)* /Paulk1,93; Paulk2,93/ and *ISO 9001 standard* /ISO9001/ together with *ISO 9000-3 guidelines* /ISO9000-3/ were chosen as the most appropriate models. The CMM proved to be a great support and guidance towards software process improvement in many organizations worldwide, while the ISO 9001 standard is already almost obligatory for the European software organizations. Within the PROCESSUS methodology we tried to use advantages of both models, therefore a detailed comparison of both models was done and an integrated model based on the comparison results was developed.

In the second phase of the project the integrated model (together with all supporting documentation and software tool) was used as the basis for the quality system improvement in cooperating organizations. The use of the integrated model provided also metrics of the quality of the integrated model and improvement methodology.

### **The PROCESSUS Methodology**

The methodology consists of three major parts which are supported by a PROCESSUS Tool and are based on the integrated model /Rozman, 97; Vajde, 97/:

#### **Introduction of the organization,**

The first part is rather informal and is aimed to restore the contacts and knowledge about organization and methodology. Using the introductory questionnaire the information about

software organization is acquired. The methodology together with other important issues (like needed time, resources, finances, human aspects, etc.) is presented to the organization's management and personnel. This step is important, since many organizations at the beginning of the quality establishment do not completely realize the complexity of needed effort.

### Assessment

In the methodology implementation three types of assessments are anticipated:

- \* **preliminary assessment** (8-16 hours) - it is performed at the beginning of the improvement project and is aimed to gather the information of procedures in the organization. Results of the preliminary assessment give the consultants guidelines for their further work, and the organization the information about their present state.

- \* **intermediate assessment** (3-6 hours) - it is aimed to obtain reports of the improvement progress. Usually only some parts of the quality system are assessed in order to gain the results of established areas and their effectiveness. It is performed as a self-assessment.

- \* **global assessment** (15-25 hours) - it is performed to assess the whole quality system. It can be performed as a self-assessment, as a second- or third- party assessment.

For the purpose of the assessment the questionnaire based on the integrated model was defined. It is implemented within the PROCESSUS Tool, which supports the assessment at the organization (filling the questionnaire) and also offers the automation of analysis of filled questionnaire (generating reports, data exports to other applications and archiving).

### Improvement activities

Improvement activities are set up according to the concept of methodology and the experiences with cooperating organizations. Activities are divided to:

- \* **Seminars.** The purpose of seminars is to educate the personnel of the organization with aspects of the quality system improvement and also the software engineering activities improvement. Within the methodology ten seminars are provided that are performed each as a one day lecture and discussion of related problems. Seminars are divided in three thematically related groups: Introduction to quality system management, Quality system improvement and Up-to date technologies.

- \* **Workshops** which are aimed to personnel assigned to preparation of a quality manual and other types of quality system documentation. At the first workshop the attendance of the management representative is recommended, because the global issues (quality policy and organization structure) are discussed. There are four workshops provided within the improvement methodology (all together 32 hours). At workshops only guidelines for quality manual and other documents are given. For that purpose the *prototypes of the quality manual together with structure and content of other documents* are prepared.

- \* **Consultations.** The consultation activities are the most flexible and detailed part of improvement activities. Namely, all actual questions and problems occurring during the quality system improvement should be solved within the consultations. Therefore, the content of each consultation is completely related to these problems. In addition, the correctness and appropriateness of defined procedures and appertain documentation are assured. For each organization a group of trained and experienced consultants is assigned. Reports of the improvement of the quality system are prepared on a regular basis and all involved personnel are informed of achievements. Needed time for consultations is hard to predict in advance because of its considerable dependency on size of organization, complexity of procedures in organization and, nevertheless, skills of personnel involved in quality system improvement. Experiences showed that in small organizations (up to 15 employees) 40-60 hours of consultations are needed.

**Results of the PROCESSUS Project**

The results of the PROCESSUS project prove that the developed methodology is directed toward right goals. In the following table some statistics of achieved results within cooperating organizations are presented.

TYPE OF ORG.	NUMBER OF ORG.	ACHIEVEMENTS
<b>A:</b> <i>Large information organizations</i>	1	The primary goal of this organization was to use the methodology for consultation activities in other companies. Some projects are already being launched in different SW companies using the PROCESSUS methodology.
<b>B:</b> <i>IT departments within large enterprises</i>	3	One of these IT departments has already arranged all procedures in accordance with SW QS requirements. The enterprise has already achieved the ISO 9001 certification. In other two cases the SPI activities within IT departments initiated the quality improvement activities along other departments. The majority of procedures performed in IT departments are already established in accordance with SW QS requirements.
<b>C:</b> <i>Independent software companies</i>	7	Two independent companies have already achieved certification. Another two companies have cooperated in project already with the attention not to apply for ISO certification. Within these two organizations some of procedures were improved. The last three organizations have stopped their SPI projects on a half way. Reasons: lack of money and other resources, low motivation of management and personnel.

**Plans for further work on SPI area**

At the time when this paper is being prepared the intensive research of software market in Slovenia is in progress. The main issues addressed in the research are: the quality awareness of Slovenian companies and the knowledge of the existing models. The information gained will be used for further dissemination of PROCESSUS methodology and SPI knowledge in Slovenia.

Further, new research has already been launched at our laboratory - the core of the research is the automation support of the software process and its influence on the personnel in organizations.

**Ukraine**

By some estimations [3] in 1986 in Ukraine about 120 thousands PC was sold, in 1997 the growth of volume of sales up to 150 thousands pieces is expected.

For support of development of information technologies the Ukrainian National Agency on Information (UNAI) with the President of Ukraine have worked out the National program of informatization of Ukraine. For realization of the program in the state budget of Ukraine it is planned to allow the following finances:

- For 1998 - US \$ 15M
- For 1999 - US \$ 16,1M
- For 2000 - US \$ 15M.

The important problem is the introduction of systems of quality in IT field.

In former Soviet Union there was determined integral state system of the quality standards in IT field. For lack of market economics the state was binding all manufacturers to perform

certain standards of quality determined manufacture of hardware and software. Such approach allowed to include shortly all the participants of IT development process in state system of quality. However, in general, due to absence of their interest the introduction of systems of quality was carrying formal character.

In 1991 Soviet Union had been disintegrated. Since then Ukrainian economics had been underwent huge changes. On IT market the share of the state enterprises had been considerably reduced. The non-state enterprises appeared was not finding time for problems of quality because of weak development of the market, recession in economics, absence of a competition etc.

However for these years (1991- 1997) there were important changes. President of Ukraine created the Ukrainian National Agency on Information . Created are the State Committee of Ukraine on Protection of the Consumer Rights and Ukrainian Association of Quality (UAQ) as well. These organisations carry out large work on introduction of quality systems in IT field. In particular, UAQ, which is the institution engaged in certification, was created in the beginning of 90-th and now it is uniting more then 170 collective members (various enterprises and organisations of Ukraine) and about 1300 associated members. Under UAQ two subdivisions had been formed. There are Inter-branch centre of quality "Gain" guided basically by classification work, certification and training; and the centre "Gain - system" engaged in development of quality systems and rendering scientific - methodical support in this sphere. Moreover UAQ carries out annual forums "Days of quality in Kiev".

Due to activities of these organisations in Ukraine the state system of certification of production (so named UkrSEPRO) is working beginning with 1993. The Law of Ukraine " About protection of the rights of the consumers " and Decree of Cabinet of the Ministers of Ukraine " On standardisation and certification" play the important role in formation of the standards of quality in Ukraine.

Ukrainian state standards of quality (SSU) had been entered by 1996. These standards are immediate application of the ISO 9000 ones. There are: SSU ISO 9000-1-95 "Standards on quality management and maintenance of quality". Part 1. "The managing instructions(indications) at a choice and application"; SSU ISO 9001-1-95 "Systems of quality. Model of maintenance of quality during designing, development, manufacture, installation and service"; SSU ISO 9002-1-95 "Systems of quality. Model of maintenance of quality during manufacture, installation and service"; SSU ISO 9003-1-95 "Systems of quality. Model of maintenance of quality during the control of ready production and its tests"; SSU ISO 9004-1-95 "Quality Management and elements of system of quality". A Part 1. "The managing instructions".

For an explanation of the basic ideas of the ISO 9000 standard and features of its introduction in Ukraine series of events had been organised by UNAI and UAQ in 1996-1997. In particular, 21.02.96 within the framework of the Exhibition EnterEX-96 the seminar " Control Systems of quality and their certification for the manufacturers of computers and service centres " was carried out.

Within the framework of co-operation to European organisations, the Ukrainian experts have taken part in the project "Qualification of Ukrainian Software Specialists" (QUALUS) maintained by the program of the German government TRANSFORM in 1996. Within the framework of the project the international seminar " Quality of the software. The information market" had been organised where the report of Jurgen Heene, which is director of WIDIS GmbH "German experience in certification on ISO 9000" has been discussed. Within the framework of the project expert of WIDIS GmbH, Dr. Bernd Schildwach has carried out in Kiev a number of consultations on installation of systems of quality.

Last years in the Ukrainian computer literature (ComputerWorld/ Kiev, Computer review etc.) [4-9] articles, interviews to the representatives of UAQ, UNAI, explaining features of the standard ISO 9000 standard, advantages of its introduction to Ukrainian firms publish regularly. At first on the Ukrainian market representations of foreign firms (IBM, Siemens, monitors - Samsung, Funai, ViewSonic, Hewlett- Packard) have certificated own production. Since the end of 1996 the largest Ukrainian firms engaged in assembly and sale of computer engineering, have begun to certificate production. For example, there are firms "Kvazar-

Micro", "Formula" and some other. One of the main reasons inducing such firms to carry out certification and to introduce systems of quality at the enterprises is the requirement of Ukrainian government to the participants of state tenders on informatization of official institutions to certificate production. In particular, due to certification of production the firm "Kvazar-Micro" could win in tender on informatization of Cabinet of the Ministers of Ukraine in the beginning of 1997.

However, these positive changes practically do not touch firms engaged in creation of the software. First of all it is connected with very small scales of this business in Ukraine, that is caused not only recession in economics, but also huge scales of a piracy. The opportunity of "free-of-charge" use practically of any software has resulted that nowadays Ukrainian software firms are concentrated, mainly, on creation of the software for automation of accounting activities, help systems on the legislation, programs of language translation, check of spelling etc., that is for those areas, which have purely Ukrainian specific character and for which, as a rule, there are no foreign program systems. Besides there are multimedia-products (textbooks, games etc.) and CAD/CAM/CAE systems, however, in very small scales because of recession in economics.

The analysis of activity of Ukrainian firms engaged in IT field, shows, that introduction at them of systems of quality depends on the following major factors.

1. The requirements of the customer.

Now there are essential differences between branches of IT market oriented on Ukraine and outside. Lack of systems of quality and information of their features prevents progress of production of Ukrainian firms on the external markets. Moreover, many Ukrainian software-firms and programmers who are carrying out some projects for the foreign customers, have met with that fact that the absence of necessary technology on maintenance of quality of production have as a result breaking off the contracts and financial losses. On inside Ukrainian market only government shows the requirements to certificate production and to introduce systems of quality (as a rule to the tender participants). Sometimes those enterprises, which need IT to use them in manufacture of export production, foreign customer of which require its certification. So for example, the authors of this presentation have met with that fact at Kharkov turbine factory, when they was selling own CAE system. It was explained by desire to certificate the factory production, which was directed to the foreign customer.

2. Scales of manufacture.

It is obvious, that under manufacture by the small enterprises of simple and unlabour-consuming production the problem of introduction of systems of quality and certification of production, as a rule, is not realised and is not put. With increase of scales of manufacture the problem of effective its organisation taking into account measures on maintenance of demanded quality becomes more urgent.

3. Qualification of the employees of firm.

The ways of the solution of the problem of effective organisation of manufacture and observance of demanded quality depend on knowledge and qualification both managers, and employers of firm. Now in Ukrainian software-firms under realisation of the large projects the emphasis is made on use of various technologies of organisation of group work: drawing up of the technical projects, organisation of working meetings, independent testing etc. Thus the questions of quality are not allocated in a separate problem, but are considered as one of necessary results.

It seems to the authors, that the realisation of the following measures is necessary for successful introduction of systems of quality in Ukraine.

1. The realisation of training seminars, schools etc., dissemination of advantages of introduction of systems of quality (including financial).

2. Development of methodology of smooth transition of various firms to complete introduction of systems of quality. Such transition should pass through some (five - six) levels of systems of quality and correspond to various conditions both firms and its production.

3. State stimulation of firms introducing systems of quality and certification own production.

## Conclusion

We claim that it is possible to increase competitiveness in the European Union and in Central and Eastern Europe simultaneously by way of mutually fruitful cooperations whose precondition is the assessment and improvement of the capability of the CEE software industry. This joint interest manifests itself by several initiatives including the ColorPIE ESBNET ESSI proposal, the PASS ESSI PIE project, the INSPIRE INCO-Copernicus project, etc... which are all represented at the ESI-ISCN'97 Conference.

However, we would like to draw the attention to a fact which has to be taken into account if we want to achieve real results. In CEE countries IT business is mainly done by SMEs which are too small to invest into SPI. They usually do not have enough capital and are very much opportunity driven. It is apparent that there is a need to further develop SPI models which would be applicable to SMEs not only in CEE but in the EU as well. Most of the present models are targeting large companies and are too „heavy" to be applicable to SMEs. The main difference should be that the feedback loop from investment to the actual benefit should be much shorter and the investment should be split into small slices. Without having such a model it is rather unlikely that SMEs will be able to enter the „improvement path" in a planned and systematic way.

## References

- [1] Biró,M.; Feuer,É.; Remzső,T. The Hungarian Quality Scene - Potential for Co-operation. In: Proceedings of the ISCN'96 Conference on Practical Improvement of Software Processes and Products. (ed. by R.Messnarz). (International Software Collaborative Network, Brighton, 1996) pp.56-63.
- [2] Biró,M. IT Market and Software Industry in Hungary. Documentation of the ESI 1997 Members' Forum.
- [3] J. Gorski, Assessment of the market needs concerning improvements of software technologies, Technical Report, Centre of Software Engineering/ITTI, Poznan, May, 1997 (in Polish)
- [4] International Organization for Standardization, *ISO 9001, Quality Systems - Model for quality assurance in design/development, production, installation, and servicing*, ISO 9001 : 1994 (E), International Organization for Standardization, Geneva, Switzerland, 1994
- [5] International Organization for Standardization, *ISO 9000-3, Guidelines for the application of ISO 9001 to the development, supply and maintenance of software*, ISO 9000-3 : 1991 (E), International Organization for Standardization, Geneva, Switzerland, 1991
- [6] M.C.Paulk, C.V. Weber, S. Garcia, M.B.Chrissis, and M. Bush, *Key Practices of the Capability Maturity Mode, Version 1.1*, Software Engineering Institute, CMU/SEI-93-TR-25, February 1993
- [7] M.C.Paulk, B. Curtis, M.B.Chrissis, and C.V. Weber, *Capability Maturity Model for Software, Version 1.1*, Software Engineering Institute, CMU/SEI-93-TR-24, February 1993
- [8] I. Rozman, R. Vajde Horvat, J. Györkös, M. Hericko, *PROCESSUS - Integration of SEI CMM and ISO Quality Models*, Software Quality Journal, March 1997



- [9] R. Vajde Horvat, I. Rozman, *Challenges and Solutions for SPI in a Small Company*, Proceedings of European Software Engineering Process Group '97 Conference, Amsterdam, 16-19th June, paper C307c.
- [10] Sobolev O.. The program of informatization - is ordered to survive. - ComputerWorld/ Kiev, N30-31 (149) from 20.08.1997, P.4.
- [11] Sobolev O.. How to confirm quality of manufacture. - ComputerWorld/ Kiev, Û 10 (82) from 13.03.1996, P.6.
- [12] Sobolev O.. Voluntary certification - not whim, and necessity for the civilised market. - ComputerWorld/Kiev, N 24 (96) from 19.06.1996, Pp.10-11.
- [13] Firm system of quality: a way to long-term success on computer the market. - ComputerWorld/ Kiev, N 27 (99) from 10.07.1996, Pp.8-9.
- [14] Firm system of quality: a beginning of a way. - ComputerWorld/ Kiev, N 4 (124) from 5.02.1997, Pp.6-7.
- [15] Ribalchenko V. ISO 9000 and quality ON. - Computer review, N 7 (80) from 26.02.1997, Pp.29-30.
- [16] Ribalchenko V., Ryabopolov S. Certification. - Computer review, N 17 (90) from 23.05.1997, Pp.26-29.

# ISCN - A Networking Concept for Transnational Co-operation in Improvement and Learning Strategies

Richard Messnarz  
*ISCN Ltd., Ireland*

Micheal Mac an Airchinnigh  
*Trinity College, Dublin, Ireland*

Miklos Biro  
*Sztaki, Budapest, Hungary*

*I.S.C.N. Ltd.*  
*Florence House, 1 Florence Villas*  
*Bray, Co. Wicklow, Ireland*  
*Tel. +353 1 286 1583, Fax. +353 1 286 5078*  
*email: office @iscn.ie, URL: <http://www.iscn.ie/>*

## **Introduction**

ISCN is a company established to co-ordinate the joint consulting and development activities of some of Europe's top experts. It focuses on collaborative cost sharing projects of different partners, all of them focusing on the development of process improvement products, training and services.

The ISCN office, established as Ltd. organisation in Dublin, is led by 3 international experts, 1 WWW administrator, and 1 co-ordinating office and conference manager. This small office co-ordinates the activities of about 40 associated experts who work in cost sharing projects in which ISCN acts as co-ordinator and make provision for dissemination of results.

ISCN (URL: <http://www.iscn.ie>) started in 1994 with a first conference in May 1994 in Dublin dealing with "Practical Improvement of Software Processes and Products". This conference series continued with ESI-ISCN 1995 in Vienna, ISCN'96/SP'96 in Brighton/London. The latest in the 'Co-operation

and Business Opportunities for Eastern/Western Countries in the IT field. Practical Improvement of Software Processes and Products' will take place in Budapest in November 1997 and is a co-operative venture.

*ISCN is partner in :*

EPIC is an ESSI dissemination action in which best practice know how (from PIEs) is discussed in up-to-date video workshop environments connecting distributed European workshops. ISCN plans to re-use the EPIC technology experience to connect the different partners of the network for a long lasting efficient communication and collaboration.

PICO (Process Improvement Combined approach) is an EU Leonardo project (started in 1995 under the EU life long learning programme). It developed a configurable set of training courses, plus a book, and a framework tool. PICO takes into account most recent improvement methodologies and is like an introduction to the different best practices covering process improvement from analysis to success. The book was written by 25 authors from 10 EU countries with contributions from Europe's leading industry. Please find further information at <http://www.iscn.ie/projects/pico/>

ISCN co-ordinates the development of NQA (Network based Quality Assurance environment) which is an Intranet based quality assurance system providing quality documentation guidelines, an on-line quality manual, computer supported project administration, templates with industry examples, and role plays for software development. The major advantage is that it runs on any WWW server and can be used in transnational co-operation, like it is used in ISCN network development co-operations. At the moment the development is co-financed by ISCN and three additional partners, including Hyperwave Ltd. Hyperwave is a hypermedia database system which can store any kind of music, videos, text, etc. and NQA can be used together with Hyperwave as the basic underlying database for Internet based archiving and access control. Please find further information at <http://www.iscn.ie/projects/nqa/>

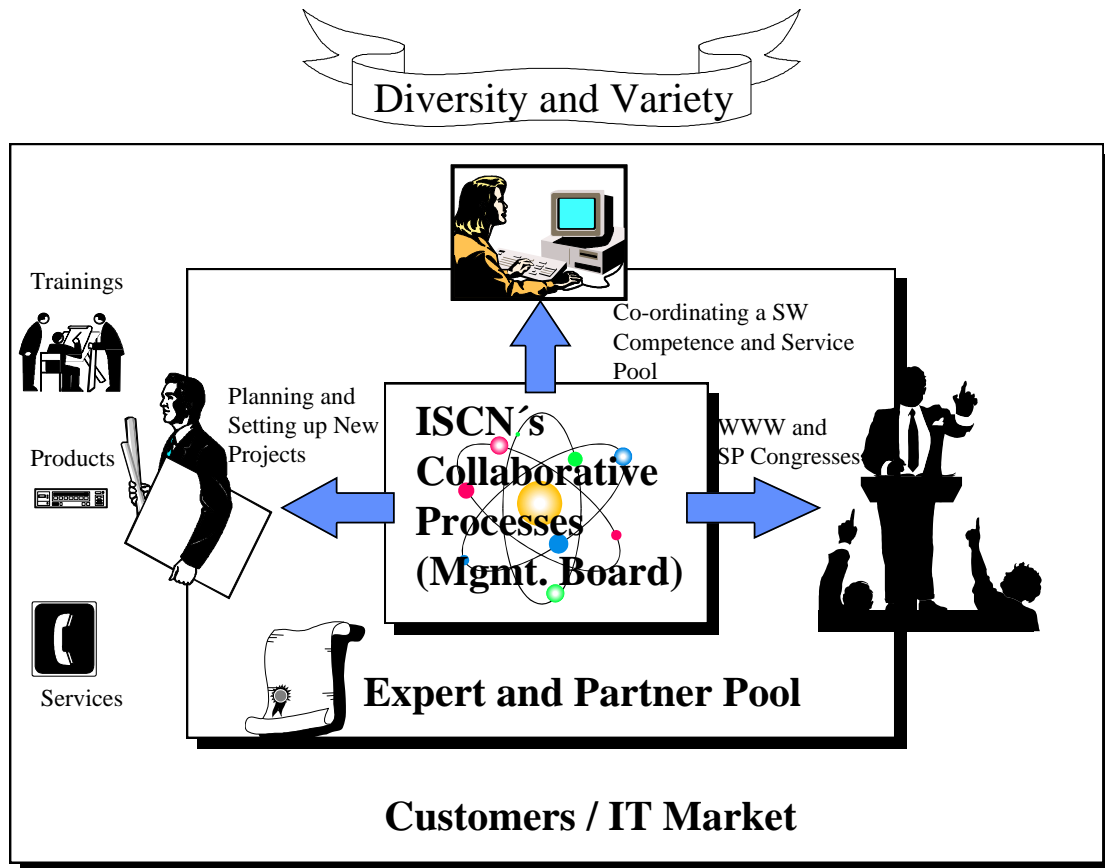
ISCN partners developed ESD (Expert and company Skill Database) which is a configurable database storing company service and expert skill profiles and providing an expert system functionality to select proper experts and companies based on skill and service data and on restrictions (such as salary, languages spoken,...). Please find further information at <http://www.iscn.ie/projects/esd/>

BESTREGIT is an EU Leonardo project (started in 1996 under the EU life long learning programme). It mainly involves non-IT partners from regional innovation transfer offices and re-uses an approach from IT programmes called PIEs (Process Improvement Experiments) in which innovative ways of work are experimented to find best practices. The project starts with a 100% industry based approach, but will continuously be refined because firstly not all principles from industry are applicable to innovation transfer organisations, and secondly there might be success principles in innovation transfer not used so far in industry standards.

PASS is a joint project with Hungarian partners (an ESSI PIE) and its purpose is to develop a modular, platform independent, integrated networked system satisfying functional requirements of EU standards in public accountancy and applicable for the Hungarian as well as to the international market.

## **Mission, Strategy and Goals**

The ISCN banner is variety and diversity. The mission of ISCN is to satisfy the needs of its partner firms for highly qualified expert support of their software process assessment and improvement initiatives. There are several methods for software process assessment, measurement, and improvement. ISCN encourages the combined use of such approaches and methodologies by using effective teamwork and collaboration based on win-win situations for all, the customers, the experts, and ISCN.



**Figure 1 : ISCN's General Architecture**

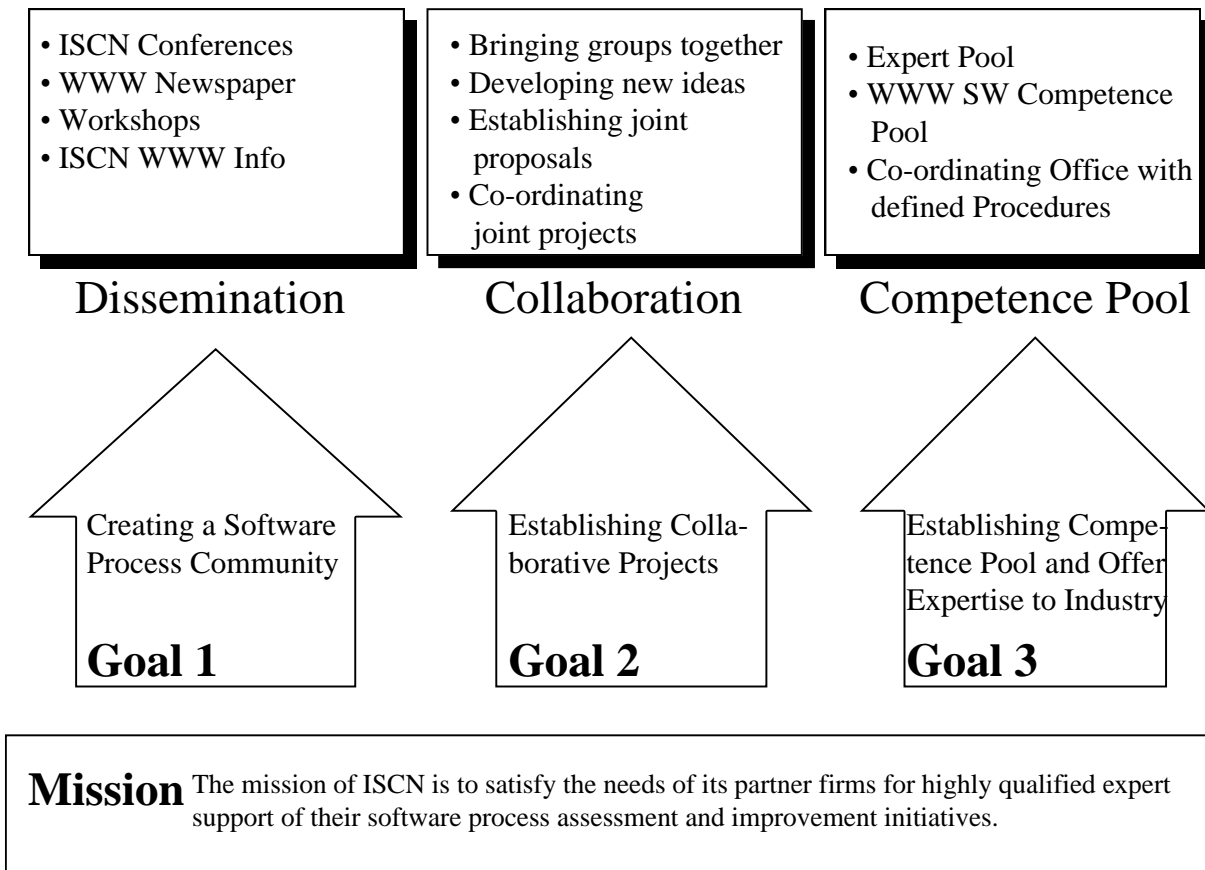
A key asset of ISCN is its pool of experts who represent a wide range of approaches and methodologies allowing a synergetic combination of the skills most suitable to the specific requirements of the customer. ISCN has been and will be committed to the highest professional traditions by applying quality assurance and continuous improvement to its own consulting processes, while it is also ready to re-engineer these processes if there is an opportunity for better satisfying the needs of business partners.

In this spirit, ISCN exploits the capabilities offered by most recent information and communication technologies to enable the most efficient organisation of its own activities, to bring about radical changes in the ways customers are served. These technologies coupled by its strong commitment allow ISCN to become a virtual part of its business partner's organisation. ISCN is by consequence an extended enterprise which stretches the traditional boundaries of professional consulting.

ISCN pursues three strategic lines of action: Dissemination, Collaboration, and the establishment of a Competence Pool. For each of these goals ISCN has been developing products and services for infrastructure support from 1994 up to now (see section 3).

Activities supporting the goal of "Creating a Software Process Community" are

- an annual ISCN conference about "Practical Improvement of Software Processes and Products"
- a process improvement newspaper on WWW which containing top articles about most recent initiatives and process improvement experiences
- a set of distributed local workshops (in EPIC - connected with video technology) dealing with process improvement



**Figure 2: ISCN's Strategic Goals**

Activities supporting the goal of "Establishing Collaborative Projects" are

- using the annual conference as a point for identifying new ideas and forming groups of partners with shared interests
- installing email and discussion groups and supporting the process of formulating new ideas in project proposals
- Co-ordinating and supporting the process of planning, estimating, and controlling the projects.
- Providing the partners with facilities for up-to-date communication and quality assurance procedures

Activities supporting the goal of "Establishing a Competence Pool of Experts" are

- designing and distributing an ISCN leaflet which contains a service portfolio
- using defined procedures (ISCN process model) for expert selection, team establishment, and project control.
- establishing collaborative agreements with SPI experienced companies based on win-win situations
- establishing a WWW pool of SPI experienced organisations

### **ISCN's Implementation Strategy and Infrastructure Initiative**

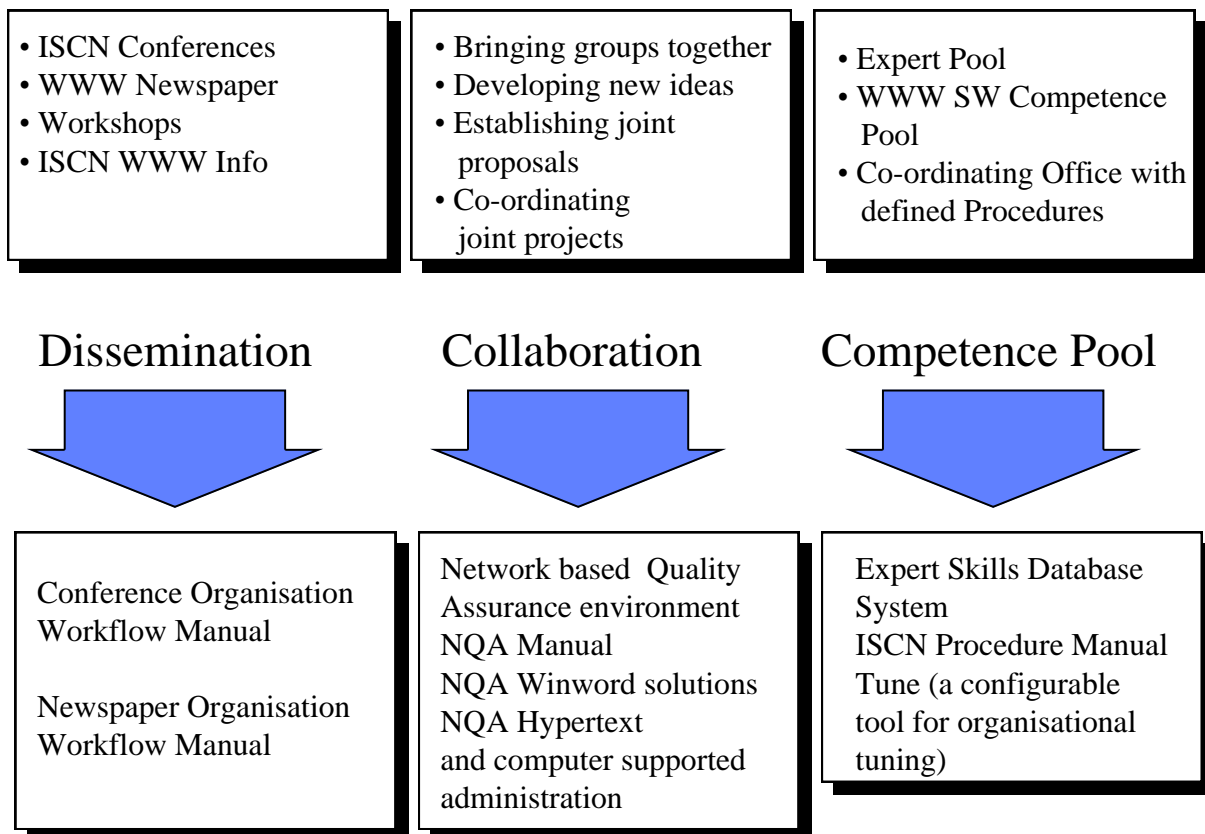
To ensure that the goals are achieved and that the activities related to the goals are efficiently carried out a number of development projects have been performed between 1994 and now. These projects aimed at the development of process models [11] and software to automate, support, and standardise best practise work processes for professionally organising and co-ordinating collaborative projects, dissemination activities, and international services (as outlined in section 2). These infrastructure

support technologies and models are available to partner organisations and also to the market based on the license agreements published on the ISCN WWW site.

ISCN recently decided to run a policy in which all technologies and models developed in-side ISCN are offered to outside organisations for

- establishing ISCN partner organisations (representatives) who (based on a franchising like business agreement) offer ISCN products, training, services, and workshops to their region
- supporting other (large organisations, networks, etc.) organisations in enhancing their infrastructure support programme and service portfolio for better organising dissemination actions, consulting, training, etc.

The following technologies and models form part of the ISCN infrastructure support programme:



**Figure 3 : ISCN Technologies and Models (Products) Supporting Collaborative Processes**

The *ISCN Procedure Manual* describes procedures such as the establishment of expert skill profiles, the evaluation of expert applications, the integration of new members into an expert pool, procedures for team selection, project establishment and co-ordination, and procedures for expert pool maintenance.

The ISCN *Conference Organisation Workflow Manual (COM)* describes a business and marketing driven approach for organising conferences. This approach is different from organising academic conferences because it mainly focuses on principles such as aggressive marketing, selecting top people and establishing an industry driven workshop-style event, and professionally designing and planning (including cost estimation) events. The manual was used for ESI-ISCN'95 in Vienna and is currently being employed for organising ISCN'96-SP'96 in London/Brighton.

The *Newspaper Organisation Workflow Manual* describes a process model to market, design,

and edit a WWW newspaper. The procedures of NOM are being used to design a WWW newspaper for the annual conference.

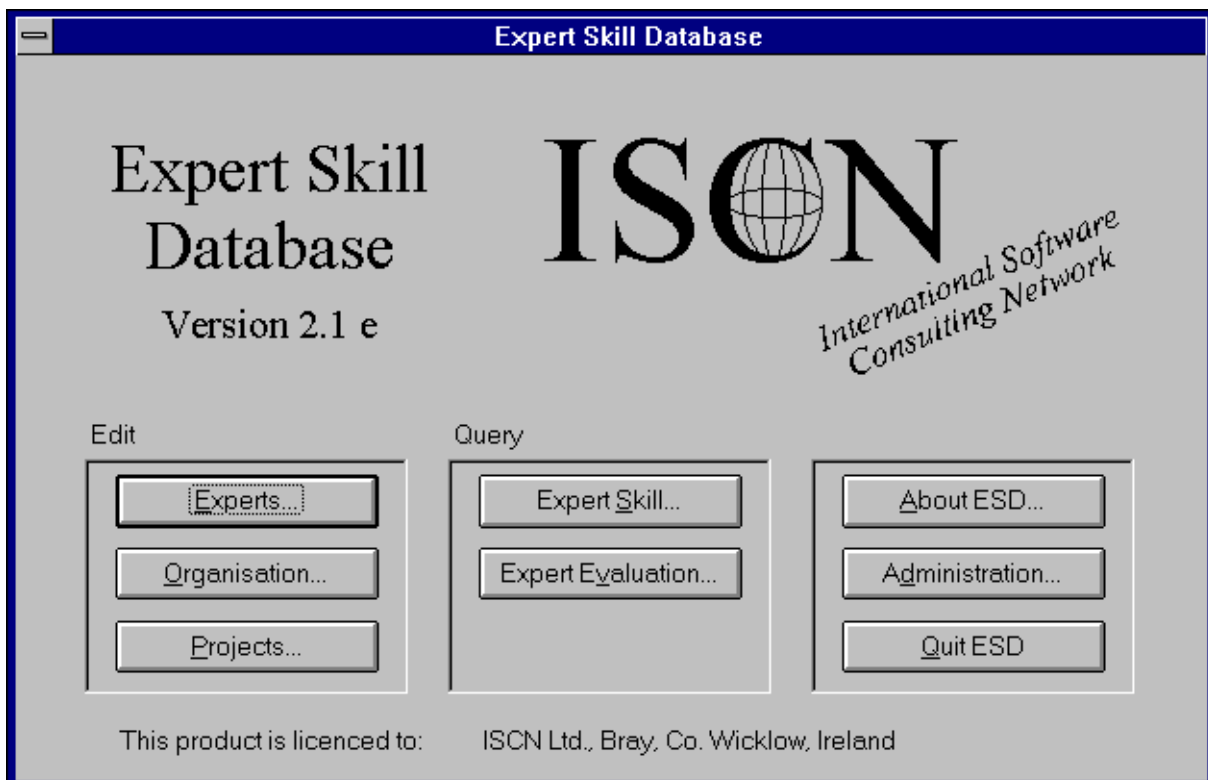
### ***The Expert Skill Database (ESD)***

ESD mainly focuses on the development of computer supported decision support systems for the establishment of regional, Europe wide, East European, or international corporate or expert networks. The ESD represents a configurable database system for expert pool administration, decision support for expert and team selection based on skill and capability requirements, including expertise re-evaluation and project administration processes.

The database can be used either for expert or for corporate networks, the skill profiles can be adapted as needed by a computer supported configuration function. In case of corporate networks the database will be configured with company service and capability profiles and the expert contact details are interpreted as the contact person for this company.

The database specifically can be used to

- evaluate experts skills and experience and to establish a computer supported expert pool
- receive project/customer requirements and to support the selection of an appropriate expert team
- establish projects and to assign teams to the projects
- evaluate the customer satisfaction leading to expert performance values taken into account in the computer supported team selection processes



**Figure 4 : The ESD Main Window - <http://www.iscn.ie/projects/esd/>**

### **Network based Quality Assurance environment (NQA)**

NQA stands for Network based Quality Assurance environment and defines a phase specific development model with planning, development, acceptance and delivery, and maintenance scenarios. For each phase (scenario) NQA provides

- an activity flow chart view
- a role based team model
- a document overview

- to be able to understand the necessary processes and worksteps from 3 viewpoints:
- steps to be performed
- roles and responsibilities
- documents to be produced

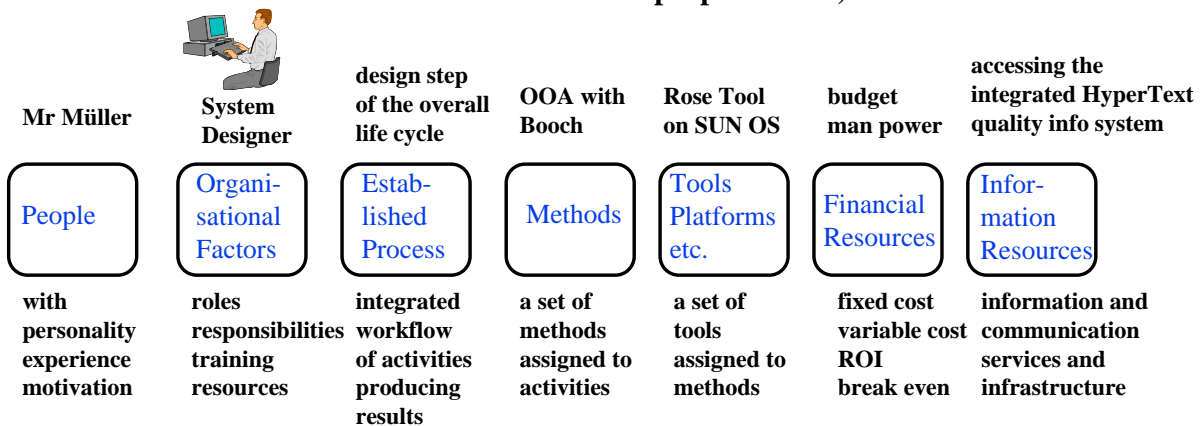
For all the documents NQA also provides content specific information, checklists, and templates. In the HTML component there is a *Road Map for Readers*.

In addition all work instructions are mapped onto ISO 9001, thus providing guidelines about how to satisfy an ISO 9001 audit.

According to NQA an *organisational system* consists of clusters of different work scenarios which can be described with process models comprising activities, workflows, results, roles, and resources. People are assigned to roles, roles are assigned to activities, activities are part of a workflow, activities produce results, and roles use resources to perform the activities. Resources can also be software systems used by a role to produce the required result. This way software systems are integrated into organisational systems, the users of a software system represent roles in the organisational system.

**Every Process is a complex integration of**

- **People** (playing roles)
- **Roles** (performing activities)
- **Activities** (supported by methods & tools)
- **Resources** (financial, information)
- **Workflows** (established process)
- **and non-Technical Aspects** (cultural and people factors)



**Figure 5 : Software Management Processes**

NQA describes a team management approach for four major work scenarios: planning and analysis and contract management, detailed design and implementation, acceptance test and delivery, and maintenance. In addition it provides management guidelines and work instructions taking into account the ESA PSS 05 Software Engineering Guidelines and the ISO 9001 international standard.

Two types of NQA are made available:

Intranet Solution for SMEs		
Basic Function	Description	
NQA Winword Macros	set of templates (including examples from industry) an index menu for selection of templates in an Intranet a set of macros automating project creation, project administration, and archiving of different types of documents	
NQA Hypertext Quality Manual	A system of Hypertext documents offering <ul style="list-style-type: none"> <li>• a quality manual index</li> <li>• a set of templates on HTML with industry examples</li> </ul>	



<b>Intranet Solution for SMEs</b>	
<b>Basic Function</b>	<b>Description</b>
	<ul style="list-style-type: none"> <li>• a road map for readers</li> <li>• an on-line description of role plays, activity flows, and document flows for planning, developing, acceptance testing, and maintenance.</li> </ul>
NQA Paper Manual	<ul style="list-style-type: none"> <li>• a printed manual (with references to ISO 9001)</li> </ul>
Complementary Software	<ul style="list-style-type: none"> <li>• after creating a project archive supported with the NQA Winword tool it is useful to employ MS SourceSafe for archiving all files with a configuration management system</li> <li>• because only standard components and software are used, MS Exchange can be used to create an effective document flow between team members</li> </ul>
<b>Internet Solution for Distributed Teams</b>	
<b>Basic Function</b>	<b>Description</b>
NQA Hypertext Quality Manual	NQA Intranet solution PLUS
NQA Hypertext Administration Functions	<p>The Hypertext on-line manual includes CGI scripts supporting</p> <ul style="list-style-type: none"> <li>• computer supported project administration</li> <li>• editing support for use of templates and archiving them under projects</li> <li>• automatic index generation for finding the description of certain terms</li> <li>• automatic contents generation of HTML project documents</li> </ul>
Hyperwave Database System as Archiv Platform	<p>A powerful object oriented WWW database system which in 1997 won the Ce-Bit award allowing</p> <ul style="list-style-type: none"> <li>• computer supporting archiving off all types of multimedia material</li> <li>• version control</li> <li>• search functions</li> <li>• etc.</li> </ul>
Complementary Software	NQA CGI HyperText can work on any WWW platform and supports any browsers, Netscape and Explorer are preferred.

**Table 1 : Overview of ISCN`s NQA Technology**

***TUNE (Configurable Electronic Questionnaire)***

TUNE is a product which will provide a framework to carry the maturity level idea into many branches, especially into non-traditional IT organisations, and non-IT organisations.

The tool can be used on small pocket computers like palmtops without any restrictions, it can be simply installed on every computer using MS DOS, and a user manual describes all functions of TUNE.

The tool

- is an electronic questionnaire which helps to gather maturity data.
- can be configured with any questionnaire using a SPICE like architecture of a process dimension and a capability dimension.
- can calculate maturity profiles with maturity levels ranging from -10 to 20.
- can store one main organisation and up to ten projects for each assessment.
- has an autosave and version management function. After each answered question,
- automatically saves the most current version of the questionnaire. If the user chooses
- to save the file, elder versions won't be lost.

- can store the completed questionnaire as an MS-EXCEL readable file and to use this interface for other MS-EXCEL based evaluation tools.
- The tool can be used on a MS-DOS platform and does not require any expanded or extended memory. Thus this tool can be used on computers like palmtops.

ISCN is the making business agreements with consultants who contribute their questionnaire, and ISCN sells the questionnaire combined with the tool. At the moment ISCN works on a project for developing a personal character development questionnaire which will test the influence of people effectiveness on the organisation's efficiency. Also TUNE will be used as an assessment tool for innovation transfer organisations, configured with a process and capability dimension specific to technology transfer in general (not only specific to software development).

## Concepts for Transnational Co-operation

The work of ISCN follows the paradigm of a "Collaborative Process Improvement Enterprise" in which a group of firms exchange know how, their service and training portfolio, and offer a large set of methodologies and services together to the market.

Here ISCN is not planning to play a dominant leader role, but the role of a supporting co-ordinator and infrastructure support partner, who also provides co-ordination of shared collaboration and development activities (share the budget and effort and work together to achieve results at less effort for your own organisation).

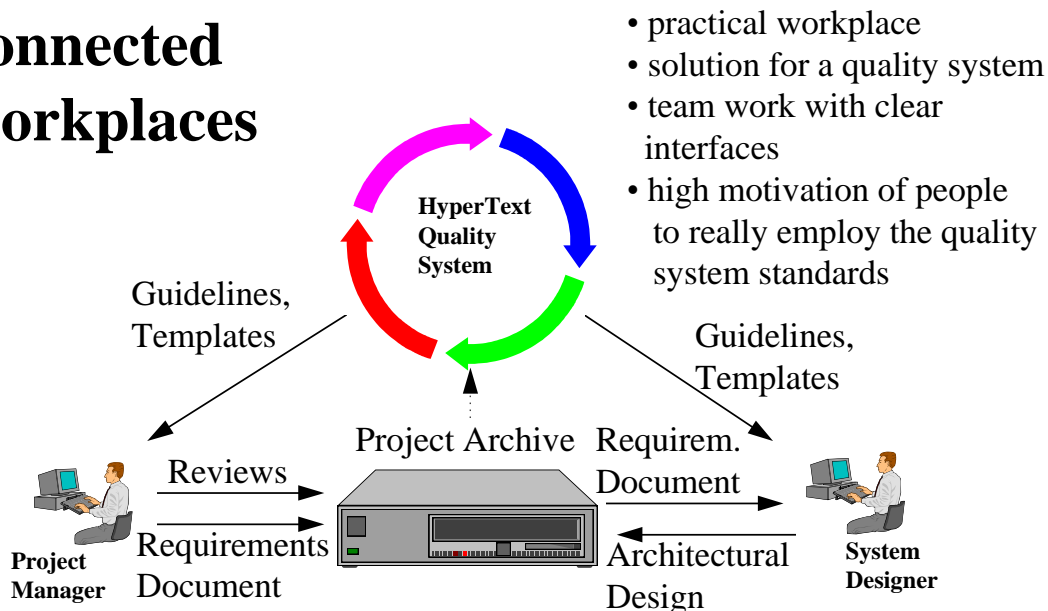
The main advantage of Internet is the significant reduction of communication cost, the possibility to establish project archives and discussion forums for teams, as well as to link different workshops together with video technology.

### Co-ordination of Distributed Projects

NQA is used in distributed EU projects in which products are developed and where scenarios such as project planning, product development, acceptance testing, and maintenance are applicable.

The NQA on-line Quality Manual (and administration system) based on HyperWave is installed and people can access templates, on-line help information concerning development procedures and document contents, and can publish results into the archive with WPW (Web Publishing Wizards).

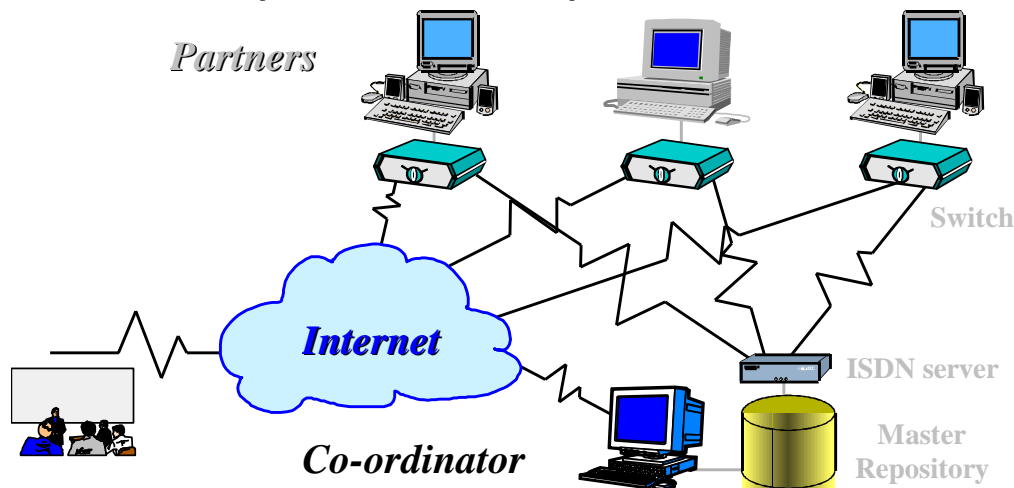
## Connected Workplaces



measured by ISCN: motivation increase from 17% to 57%, ISO Certification in 5 months

Figure 6 : Using a HyperText Quality System over Internet

### **Distributed Process Improvement Workshops**



**Figure 7: Video-Audio- Real Time Communication Systems**

Sometimes a co-operation via a joint NQA WWW Quality server still requires personal contact of focused teams. For this there is a group of video-audio-chat systems on the market (CU See-Me, Webboard, Netmeeting, ...) who are quite cheap (less than 100 US dollars) and which can work with guaranteed quality over ISDN and with sufficient quality over Internet (depending on the bandwidth). These Web-based meetings can be done very cost-effective by, for instance, transferring and freezing the picture, and doing the communication with audio and with chat (if audio is too slow). Even if you use ISDN (then you have guaranteed quality) the connection cost is much cheaper than a project meeting cost.

### **WWW Search Engine for Expertise**

ISCN's long term plan is to run ESD not only as a local system for large consulting firms, regional technology transfer offices, or networks, but to make it available through WWW forming a Search Engine on the Web where experts can register with their skill profiles (paying a registration fee) and companies can access and select expert teams according to their needs (paying a small consulting fee).

### **International (Across-Countries) Experience Exchange**

Of course it is important to create awareness for process improvement in the regions of Europe and to initiative process improvement related actions in the regions. However, it is as well important to establish a platform (like the annual ISCN conference) to allow exchange of the experience made in the regions across the different nationalities.

A small joke: If there would not have been business with the Indians in the middle age we would miss a large number of spices which make our food tasting good. And the same philosophy ISCN applies for the conference series. Every year we

- form a partnership of organisations from different countries
- announce the event across nations and countries
- motivate transnational experience exchange
- and focus all topics around the issue of "Practical Improvement of Software Processes and Products", discussing potentials of co-operation, set up of win-win partnerships, and presentation of new ideas and approaches.

### **ISCN's Partnership and Co-operation Model**

The current situation in IT industry shows that most management failures are due to inefficient capacity planning [8] [10], and in many cases the cost are underestimated, the market is more competitive than expected, and the identification and establishment of new business which should

bring return on investment (ROI) becomes a game theory problem [17].

ISCN pursues an approach which could be described as "market driven cost sharing model". The below phase model describes a kind of living philosophy of ISCN.

### ***Phase 1: Idea Exchange and Focused Teams***

Usually at conferences, workshops, and within the ISCN team a number of brand new ideas are discussed. The ISCN co-ordination office then tries to prepare executive summaries of the ideas and to identify groups of partners who have shared interest in a certain approach, product, etc.

### ***Phase 2: Group Set Up and Team Identification***

After a joint vision and product (or service) idea has been created ISCN follows the approach of radical marketing, by presenting the idea to others in the conference, and in the workshops, and by trying to find additional contributing partners strengthening the team (even sometimes offering it already as a future product getting feedback on the market demand). Usually in this phase the effort is distributed among a group of partners, and with ISCN as a co-ordinator of the team. This results in an identified team with a clear mission, a plan, and commitment to put further effort into it.

### ***Phase 3: Initiative Kick Off***

In this phase either an EU proposal is written, or a development agreement of a group of partners is established. ISCN in this phase is leaving its dominant position and another partner of the team takes the prime position, whereas ISCN becomes an infrastructure supporter and expertise provider.

### ***The Sustainable Business Cycle***

Every year the conference forms a platform for discussing the ideas needed for starting phase 1 of the above model.

## **Future**

ISCN's future goals are to work on collaborative concepts, to identify and exploit further infrastructure support and team co-ordination potentials, and to adhere to the model of networking across different countries and nations.

In future ISCN plans to grow into a set of local partners per region of Europe which offers the ISCN products and training courses on behalf of ISCN.

A collaborative bridge between countries, regions, companies, universities, and non-IT organisations will be a key challenge in the future and ISCN believes that such an approach based on multi-nations, multi-methods, and multi-industry will fit the European market very well. It is like finding a concept of speaking to each other, while still everyone can speak his own native language (language in this context as an acronym for process improvement experience)

## **References**

- [1] Basili V., An Experience Factory, in: *Proceedings of the International Conference on Lean Software Development*, Stuttgart, Germany, Oct. 1992
- [2] Biro M., Kugler H-J., Messnarz R., et. al., BOOTSTRAP and ISCN - A Current Look at a European Software Quality Network, in: *The Challenge of Networking, Proceedings of the CON'93 Conference*, Oldenbourg, 1993
- [3] Boehm B.W., A Spiral Model of Software Development and Enhancement, in: (eds.) Nahouraii E.,

- Butler J.T. Kavi K., Petry F.E., Richter C., Tham K., *Software Engineering Project Management*, pp. 128-142, Comput. Soc. Press of the IEEE, Washington, DC, USA 1988
- [4] Chroust G., Computer Integrated Work Management, in (ed.) Mittermeir R., *Shifting Paradigms in Software Engineering*, pp. 4 -13, Springer Verlag, Wien, New York, Sept. 1992
- [5] Debou C., ami - A New Paradigm for Software Process Improvement, in: *Proceedings of the ISCN'94 Conference on Practical Improvement of Software Processes and Products*, ISCN Ltd., Dublin, Ireland, 1994
- [6] Debou C., Fuchs N., Haux M., ami - A Taylorable Framework for Software Process Improvement, in: *Proceedings of the ESI-ISCN'95 Conference on Practical Improvement of Software Processes and Products in September 1995 in Vienna*, ISCN Ltd., Dublin, Ireland, 1995
- [7] Dorling A., SPICE - Software Process Improvement and Capability dEtermination, *Software Quality Journal*, Volume 2, 1993, pp. 209-224.
- [8] Genuchten M., Why is Software Late ? - An Empirical Study for Delay in Software Development, *IEEE Trans. Software Engineering*, pp. 582 - 590, June 1991
- [9] German Interior Ministry, German V-Model, Bonn, August 1992
- [10] Glass R.L., The Software Crisis - Is it a Matter of Guts-Management, in: (ed.) Reifer D.J., *Software Management*, Fourth Edition, IEEE Computer Society Press, Los Alamitos, 1993
- [11] Kellner M. I., A Method for Designing, Defining, and Evolving Software Processes, in: *Proceedings of the European Software Engineering Process Group Conference 1996*, Software Engineering Institute, Carnegie Mellon University, May 1996
- [12] Messnarz R., Mac an Airchinnigh M., Biro M., Tully C., ISCN - The International Software Collaborative Network, in: *Proceedings of the ISCN'96 Conference on Practical Improvement of Software Processes and Products in December 1996 in Brighton/London*, ISCN Ltd. Dublin, Ireland, 1996
- [13] Messnarz R., Kugler H.J., BOOTSTRAP and ISO 9000: From the Software Process to Software Quality, in: *Proceedings of the APSEC'94 Conference*, Comput. Soc. Press of the IEEE, Tokyo, Japan 1994
- [14] Rohen M., ESSI - The European Systems and Software Initiative, in: *Proceedings of the ESI-ISCN'95 Conference on Practical Improvement of Software Processes and Products in September 1995 in Vienna*, ISCN Ltd. Dublin, Ireland, 1995
- [15] Rutschek G., FESTO - Experience with the ESSI Application Experiment ODB, in: *Proceedings of the ESI-ISCN'95 Conference on Practical Improvement of Software Processes and Products in September 1995 in Vienna*, ISCN Ltd. Dublin, Ireland, 1995
- [16] Stewart J., Leonardo - The Networking Programme, in: *Proceedings of the ESI-ISCN'95 Conference on Practical Improvement of Software Processes and Products in September 1995 in Vienna*, ISCN Ltd. Dublin, Ireland, 1995
- [17] von Neumann, Morgenstern, *Theory of Games and Economic Behaviour*, Princeton University Press, Princeton, New York, 1944

## Author CVs

### ***Dr Richard Messnarz***

Dr Richard Messnarz graduated from the University of Technology Graz, Austria in 1990. From 1990 to 1995 he has been involved in a number of different European research projects and industry initiatives. In the ESPRIT project BOOTSTRAP he developed the algorithm for calculating maturity profiles of software processes. He performed 12 BOOTSTRAP assessments at companies from 4 different countries (his leading customer is Robert BOSCH in Germany). In the ESSI project 10788 he worked together with the German company FESTO and developed a lean life cycle and team model for software development. In the COMETT project ISCN (International Software Consulting Network) he established a conference series and a business firm for co-ordinating the work of a network of process improvement experts and for establishing collaborative projects of ISCN members and partners for problem solving and training purposes.

In 1996 he became the technical director of ISCN Ltd. which at the moment is the technical co-ordinator of a training initiative called PICO (Process Improvement Combined Approach) which is funded by the EU Leonardo program, which organises annual SP congresses in partnership with leading process improvement conferences, which is the EU co-ordinator of a PIE (Process Improvement Experiment) in Hungary, and which is partner in various industry projects dealing with business processes and new software engineering approaches.

His fields of expertise are: process analysis, process modelling, object oriented analysis, approaches such as „component software and outsourcing models“, „virtual enterprises“, and „learning organisations“.

Since 1994 he is a conference chair of the ISCN conference series, with ISCN 94 in Dublin, ESI & ISCN 1995 in Vienna, ISCN 96 in Brighton near London, and ESI & ISCN 1997 in Budapest.

### ***Dr. Mac an Airchinnigh***

Dr. Micheal Mac an Airchinnigh, President of ISCN Ltd since its foundation in 1994, is a Senior Lecturer in Computer Science (1993) at the University of Dublin, Trinity College.

Dr. Mac an Airchinnigh graduated from the University of London with a degree in Pure Mathematics (BSc 1978). Having completed a Master's degree in Computer Science at the University of Dublin (MSc 1981) he joined the faculty as Junior Lecturer in Computer Science. He obtained his doctorate in Computer Science in 1991 (PhD. University of Dublin).

In addition to his academic work, Dr. Mac an Airchinnigh has been a founder member of both VDM Europe and Formal Methods Europe. He is also a past chairman of Ada Europe. Dr. Mac an Airchinnigh has been a consultant with Motorola Cellular, Arlington Heights, IL, USA.

### ***Dr. Miklós Biró***

Dr. Miklós Biró graduated from the Loránd Eötvös University of Budapest as a mathematician in 1977. He is research associate of the Computer and Automation Institute of the Hungarian Academy of Sciences (MTA SZTAKI) since that time. He is currently the Head of Quality Management of MTA SZTAKI which acquired TÜV CERT certification according to EN ISO 9001. He defended his doctoral thesis in 1983 and was awarded the newly established Hungarian Ph.D. degree in 1995. Still in 1995, he obtained the Executive MBA (Master of Business Administration) degree from the International Management Center at École Supérieure de Commerce de Rouen, France in association with the Krannert Graduate School of Management at Purdue University, USA. He speaks fluently both English and French.

He taught operations research and integer programming since 1978 at the Loránd Eötvös University of Budapest. He worked as a professor of mathematics, computer science, and operations research in the USA (City University of New York, St. John's University) from 1983 to 1987. Currently, in addition to his project management and head of quality management activities at MTA SZTAKI, he delivers lectures and Ph.D. courses on software quality management at Hungarian universities.

He has numerous publications in international scientific journals and conference proceedings in the fields of combinatorial optimization, human-computer interaction, decision and negotiation support,

and software quality management.

He initiated and is managing the membership of MTA SZTAKI in the Bootstrap Institute, the European Software Institute (ESI), and the Hungarian Quality Consultants' Association. He is an approved Bootstrap assessor. He initiated and is managing the Hungarian participation in the European Union supported ESSI/VASIE (Value Added Software Information for Europe), ESPRIT/Web4Groups, ESSI/PIE (Process Improvement Experiment), and INCO/COPERNICUS projects. He manages the National Point of Contact of the ESA-IRS (European Space Agency Information Retrieval Service) activities partially performed by MTA SZTAKI.

He is member of the editorial board of the journal on Software Process Improvement and Practice published by Wiley, president of the professional division for Software Quality Management of the John von Neumann Computer Society, member of the joint healthcare software accreditation body of the above society and the Hungarian Healthcare Informatics Society. He is managing director of IT Information Technology Consulting Ltd., a senior consultant member of ISCN (International Software Consulting Network), and invited consultant to various international and Hungarian organizations. He is member of the Hungarian Quality Society, the Hungarian Operations Research Society, and the decision support systems working groups of IFIP and EURO.

# Learning to Improve with BOOTCHECK

Amor Dominguez  
*ESI, Spain*

Izaskun Doiz  
*ESI, Spain*

*European Software Institute  
Bilbao, Spain  
amor.dominguez@esi.es*

BootCheck is a cost-effective, easy to use self-assessment tool initially conceived to create awareness among industry users about the benefits of software process assessment and improvement methods. Recent industry experiences confirm that the tool can be used in very different environments independent of the level of software involvement of the organisation.

BootCheck self-assessment tool is part of the EXPRESS project, an initiative promoted by the European Software Institute to create awareness about the benefits of software process assessment and improvement methods. The aim of BootCheck is to enable individual Software Producing Units (SPUs) in an organization to make quantified assessments of their software capabilities, and to use this information as a key element in continued process improvement programmes. Final self-assessment results are presented graphically according to three widely accepted best practice models: ISO 9000, SPICE and BOOTSTRAP, which serve as a useful reference framework throughout the assessment process.

As part of the activities planned in the development of BootCheck, ESI carried out a beta test in order to collect some feedback on its overall usefulness and its applicability in the industry. A total of sixty-one organizations in Europe and India took part in the beta test. Evaluators were asked to rate BootCheck's general and technical features. Based on the feedback collected, 'Overall usefulness' stands out as being the most valued feature, with 88% of the participants sharing this appreciation. 'User friendliness' (75%) and 'General guidance' (74%) also ranked top among other features evaluated by BootCheck users. In addition, an outstanding majority (83% of the sample) stated to have been self-sufficient when using the tool. These figures suggest that the level of technical expertise required to use BootCheck is low, although some background on SPI methods is advisable. Finally, the interest shown for future versions of BootCheck is overwhelming, with 93% of the beta test participants interested in receiving information about forthcoming versions.

At this point, it is noteworthy that beta test participants profile corresponded to a fairly homogeneous set, with 90% of the organizations stating to have been involved in some kind of software improvement initiative within their organizations. Obviously, a favourable attitude towards SPI initiatives can be expected from such a highly receptive sample, which could somehow call into question the objectiveness of the beta test results obtained. However, despite their shared consciousness on SPI benefits, organizations involved in the trials varied significantly in size and business activity, and in their software involvement level for that matter. This clearly emphasizes the



fact that BootCheck's wide scope of application has undoubtedly favoured the high acceptance level obtained in such markedly different business environments.

In addition to the advantages which can be directly associated with the self-assessment process itself, further benefits can be obtained when BootCheck is used in conjunction with an experienced assessor, i.e. when Assisted-BootCheck sessions are performed. To mention a few: on-site training and guidance are provided throughout the assessment session; increased objectiveness is also achieved through the assessor's mediation in the interpretation of the assessment results; and, finally, a comprehensive assessment report is elaborated where conclusions are presented according to the SPICE framework.

Assisted-BootCheck sessions were performed at eleven user sites in the Basque Country, Spain. Organizations assessed ranged from small (32 employees) to large (3,300 employees) and pertained to different industry sectors such as: white goods, aeronautics, machine-tool, and automotive industry. Independent of the application environment selected, general agreement was that BootCheck can be successfully used as a first step in the implementation of an improvement process. Specially valued by the users was the immediacy of the graphical analyses obtained, a distinctive attribute directly related to BootCheck's model-based approach.

To conclude, the extremely favourable reception of a model-based tool like BootCheck highlights the fact that simple but cost-effective SPI initiatives are certainly welcome by organizations willing to improve their software practices. The positive beta test results reveal that user expectations have been largely met with BootCheck's current level of sophistication. Moreover, the enthusiasm generated by the initial use of BootCheck has triggered the development of new BootCheck-related products and services which will further support industry users. Future enhancements to be incorporated to BootCheck include: an organization-wide profile which integrates multiple assessment sessions; a benchmarking report which enables the comparison of organizational performance against industry best practice; and, finally, a pre-assessment preparation session to define the scope of a full SPICE assessment.

BootCheck has recently been placed in the public domain and is being distributed widely to support the European software industry at large.

# Success Factors in Technology Transfer: Quick State-of-the-art Mapping

Christophe Debou

*Q-Labs GmbH, Kaiserslautern*

Anders Gustavsson

*Q-Labs GmbH, Kaiserslautern*

## **Introduction**

This article discusses Q-Labs strategy for transferring technology in the area of Software Engineering. The idea of our strategy is to maintain a position on the cutting edge of research and at the same time assisting industry in gaining leadership in software engineering (quick state-of-the-art mapping concept). Transferring technology normally implies major changes in the organisation and the Q-Labs way have proven to work when integrating goal-oriented changes while taking the human aspects into considerations (Change facilitation) Those concepts are illustrated with some lessons learnt from several technology transfer programs. Finally, the potential of central and eastern European (CEE) countries for technology transfer applying those concepts, is highlighted. Q-Labs is currently establishing co-operation in Central/Eastern Europe (CEE) as part of its strategic world-wide network of leading software engineering research and technology transfer institutions.

## **Introduction to Q-Labs**

Q-Labs is a services company in the Software Engineering field. Q-Labs works with customers requiring process improvement strategies to increase quality and productivity while at the same time shortening cycle time. The majority of Q-Labs clients are organizations developing and/or purchasing large and complex software intensive systems. Q-Labs represents one of the most prominent groups of experts within the field of Software Process Improvement, Software Quality Methods and Systems Analysis. The corporation employs more than 50 experts, many with over 15 years industrial experience. Q-Labs is located in Sweden (Lund and Linköping), Germany (Kaiserslautern) and in the USA (College Park, Maryland). All Q-Labs offices hold an ISO 9000 registration. One of Q-Labs strengths reside in its network of strategic alliances with major institutions as summarized in the figure below.

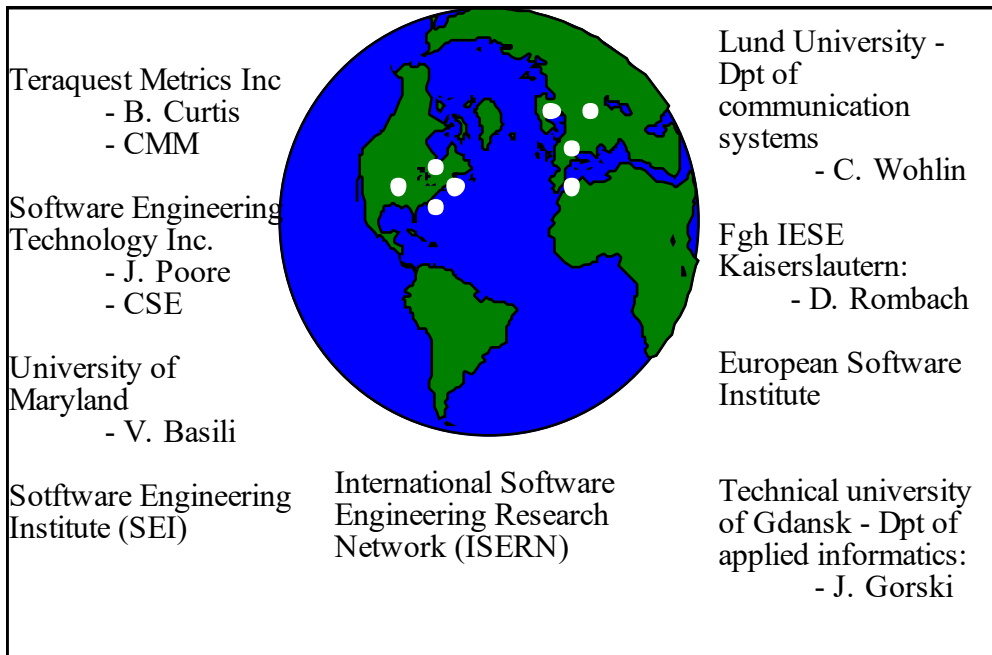


Fig. C.DEBOU.1 : The Technology Transfer life cycle

Q-Labs experience areas include software process improvement (CMM, Experience Factory, IDEAL model, G/Q/M, PSP), Cleanroom Software Engineering (incremental development, functional verification, statistical usage testing), human factors (people-CMM, team building, change management), architecture and reuse (domain specific architectures, reuse strategy, reuse management, reuse process, re-engineering) and software acquisition (procurement process improvement, supplier and project audits, product assessments, software certification, quality assurance).

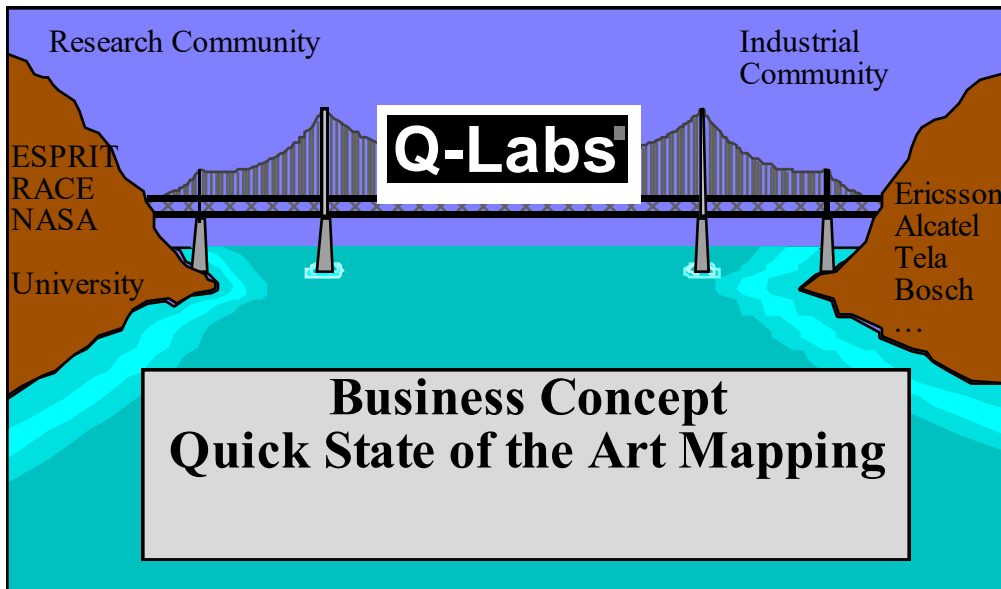


Fig. C.DEBOU.2 : Quick State of the Art Mapping

Q-Labs business mission is to transfer state-of-the-art methods for software engineering from the

research community to research reality in cooperation with customers and partners. The idea of “quick State-of-the-art Mapping” is to maintain a position on the cutting edge of research and at the same time assist industry in gaining leadership. This enables to forecast the methods and techniques that will be competitive in the future and to transfer the knowledge to customer. For instance, Q-Labs pioneered the establishment of Cleanroom ([Proceedings) Software Engineering in Europe. Cleanroom philosophy is based on the paradigm that it is perfectly feasible to develop defect-free software in actual practice.

### Quick State of the Art Mapping & Change facilitators: Success factors

Successful quick State-of-the-art Mapping in Software Engineering is based on the cooperation of different organizations with different focuses. The main idea is to based on the findings in the industrial setting to identify solutions that solves the specific problems of the organization. Looking into the life cycle of a new technology in figure 3. We can see that new ideas are developed and refined in basic research environments, these ideas are then applied in different contexts and environments where the applicability of the technology is evaluated in pilot projects and experiments. If the technology is found suitable, the technology need to be transferred to the whole organization and will finally if successful be regarded as a best practice for the organization with a wide spread usage. The role of q-labs is to work closely with applied research organization to capture, evaluate and to spread the technologies in the organizations. Finally the role of Q-Labs is to feed back experiences both to the applied and basic reassert organizations.

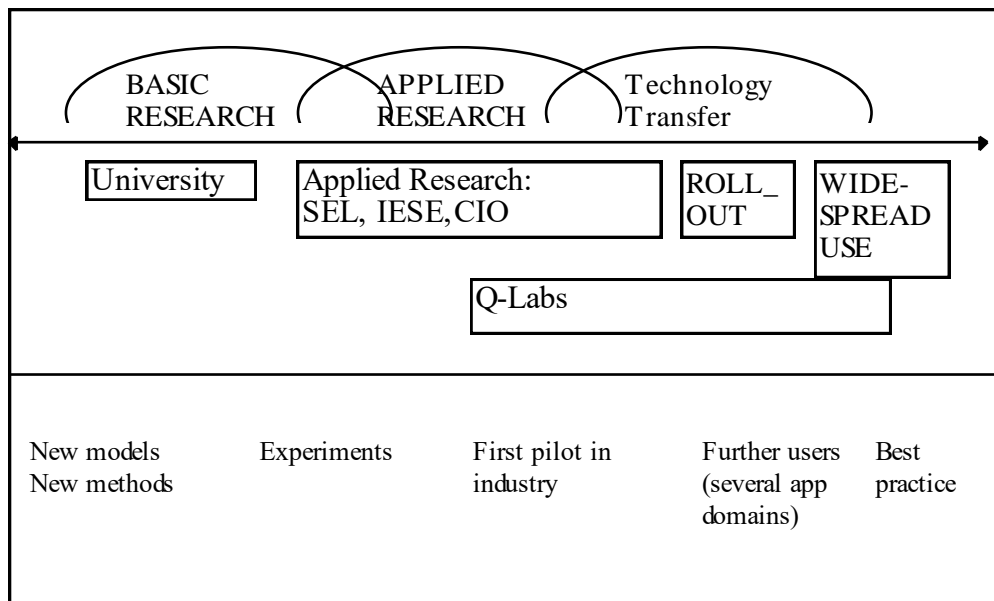


Fig. C.DEBOU.3 : The Technology Transfer life cycle

Successful technology transfer is based on the notion of systematic process improvement. Here the organization investigates how the changed technology will affect the organization/projects. A well defined scenario where the experiences from the previous use of the technology is taking into consideration for the technology transfer. For technologies where only “success” stories of other organizations or for research results it is recommended to make a first desktop evaluation in order to evaluate the proposed technology within the organization. When more in depth results are presented, a more thoroughly desktop evaluation can be performed. The desktop evaluations allows the organization to easily and with a limited cost to sort out technologies not appropriate for the organization. The next step to take is to perform some experiments out of the context of the development organization. Some organizations provide this kind of experiments allowing for other to

gain from these experiments. The final step is to introduce the technology in the real context, testing whether the selected technology meets the set hypotheses. This should be done in two steps, pilot projects and finally by introducing the technology into real projects.

With all due respect for new techniques and methods, the biggest challenge related to implementation of change is often on a very different level - the human one. Most people have deeply rooted resistance to change and aim instead at a balanced status-quo. They find security in what is known, in the working methods that they have mastered and can control. Unfortunately, there are many examples of organizations that have underestimated the effect of human factor on implementation of change. Practical work on the process of change in Software-dependent companies is the actual core of Q-Labs operations: "change facilitators"

## **Experiences with Quick State of the Art Mapping**

If a company wishes to start continuous improvement of its software process, it will be important to realize early on that this work is long-term. Profits and results may not be visible before some years have passed. It could take more than 2 years before any changes break through. Reserved and secured resources with a budget for a long time are required in order to establish a sufficient infrastructure for the improvement work.

According to experience from the USA [SEI] shows that about 70% of the assessments carried out did not lead to carrying out an improvement plan. One important reason for this is considered to be failure in the two first stages of the improvement process, i.e. there was insufficient commitment from management and an infrastructure for the improvement work had not been created.

Other reasons for improvement programs dying out or failing could be:

- Profit hunger: It is not realized that an improvement program is long-term. When profits are not forthcoming after, for instance, six months to a year, then the budget is cut and resources redistributed.
- Insufficient resources: Budget may be set aside but if there is not enough personnel with adequate skill the program may fail.
- Schedule: Great expectations and unrealistic plans for the improvement project.
- Inappropriate goals: One example is level 5 in five years. It often takes 2-5 years to move, for instance, from level 1 to level 3. Often, a change in corporate culture and working practices is required, which will not come overnight.
- False priorities: This could either include attempts made to skip Key Process Areas (KPA) on lower levels to instead focus on KPAs on a higher level. The reason for the levels is to group the KPAs such that KPAs on higher levels build upon KPAs on lower levels being in place. It could also depend on a poor understanding on the current software development capabilities in terms of defects rates etc.
- Problems with introduction: Improvements to be introduced entail major or minor changes in working practices. This needs to be introduced with motivation and training. Just introducing through giving orders can entail reluctance and that methods are thwarted.
- Relapses: When the schedule begins to pressurize a project, it becomes easy to relapse to previous working practices, which is also a consequence of a weak commitment from management and staff.

The most important thing is to be aware that improvement work takes a long time. The time elapsing until productivity improvements become visible can be several years and the staff and organization must get used to new working methods.

## **Software Opportunities in Central and Eastern Europe**

The presence of Q-Labs at ISCN97 which main objective is to bridge the gap between western and

central and eastern European IT firms, is motivated by a strong conviction of the software potential of those countries. They seem to enclose the same Software Engineering opportunities as Asia. So why would we go so far when some similar activities can be conducted just across the borders, and start building the new European Union. The IT industry is already and will be an essential asset of a country in the context of the globalisation of the economy. The US have a strong leadership that EU only contests in the area of software-intensive embedded systems (e.g. Telecom).

Europe is even in certain place suffering from a lack of software engineers. Some figures like 20000 software engineers missing in Sweden or 2/3 less computer science students in some German universities than some years ago have been publicized. On the contrary, computer science is still a prestige study in the technical field in central and eastern Europe.

CEE countries enclose the following strengths:

- High level of technical competence
- Low labor cost. The costs are about 1/3 to 1/4 of western costs in countries like Hungary or Poland, even lower further east in Ukraine, Russia.
- Very high motivation of younger generation to succeed.
- Rapidly growing software market e.g. The software market in Poland is estimated to approx. 1.5 billions of US\$ and it is expected that it will grow to 6 billions US\$ by year 2000 (UNISYS study).

However, one should be aware as well of the issues that may occurred:

- The Software Industry is young. Software companies have a maximum age of 7 years. This is positive on the one hand since there are not confronted with the typical western non-software company issue of evolving from a hardware culture to a software culture. On the other hand, companies have grown rapidly and in an uncontrolled manner and experiences in managing larger software team may be lacking.
- Low awareness level about process and quality issues in software development for both suppliers and customer levels. However, this situation will evolve rapidly due to the growing competition. Currently the highest receptivity on those issues is coming from customers e.g. banks.

Q-Labs's objectives in central and eastern Europe are twofold:

- First, support raising of awareness of local software industry on software process, quality and software engineering technology transfer. The ultimate goal is to raise capacity of software industry
- To identify centers of excellence in software engineering to extend Q-Labs network of alliances. Strategic partnerships will be established and worldwide projects set up using each other expertise. This expertise is being so far underused because CEE institutions have been acting locally for political and financial reasons.

These two objectives have started to be implemented with as first target country Poland. A cooperation has been initiated with the department of applied informatics of the technical university of Gdansk (Prof. J. Gorski), which is attempting to build a technology transfer center in the area of software process improvement. A common awareness event sponsored as well by the European Software Process Improvement Foundation (ESPI) will be organized on the 25th of October in Gdansk (Part of INFOMAN97) with the participation of leading international industrial speakers and local software house representatives (e.g. CSBI, Prokom). The newly established Polish initiative on Software Process Improvement (IP3) will be announced during this event.

Q-Labs is wishing to extend the positive Polish experience with other CEE companies and institutions. An essential success factor in deploying those concepts is to support local industry with "local" staff. The cultural issue that non-local companies can face in CEE countries is even stronger than in the

European Union.

## **Joining Q-Labs Network of Excellence**

As emphasized previously, Q-Labs is intending to extend its worldwide network of excellence towards central and eastern European countries. Let's envisage scenarios that highlight a win-win situation between the potential partners and Q-Labs.

**Scenario 1:** You are a leading CEE software engineering research or technology transfer institutions with specific competences. Q-Labs is offering that:

- 1) Your experts join some of our worldwide projects to promote your competencies and apply some new approaches at industrial site.
- 2) Q-Labs experts join some of your local project to complement your expertise.

**Scenario 2:** You are an CEE organisation developing software or software intensive systems. Q-Labs is offering to:

- 1) support you in improving your software practices to fulfill requirements of your customers and especially of western ones
- 2) help you in establishing business partnership in western Europe within our network of customers

**Scenario 3:** You are a CEE software consulting organisation. Q-Labs is proposing you to:

- 1) complement your competencies in software engineering to approach local clients and vice&versa for Q-Labs clients to use your specific competencies.
- 2) help you building further competencies by incorporating your staff within our projects

For those three scenarios, the major requirements to all institutions is local leadership and competencies that would complement Q-Labs set of products and services.

## **Conclusion**

The receipt for a successful technology transfer includes a bit of technological expertise, some management skills and a lot of human-related competences (coaching, cultural knowledge). This is certainly the biggest strength of Q-Labs to possess competencies in all those three components. Technology transfert in software improvement in Central and Eastern Europe is an exciting challenge. Q-Labs intends to take part actively in improving competitiveness of the local software industry: a stronger IT industry in the future EU members will lead to a more competitive European Union of the Year 2000.

## Q-Labs

Q-Labs is a consulting company in the Software Engineering field. Q-Labs works with customers requiring process improvement strategies to increase quality and productivity while at the same time shortening cycle time. The majority of Q-Labs clients are organizations developing and/or purchasing large and complex software intensive systems.

Q-Labs represents one of the most prominent groups of experts within the field of Software Process Improvement, Software Quality Methods and Systems Analysis. The corporation employs 50 experts many with over 15 years industrial experience. Q-Labs is located in Sweden (Lund), Germany (Kaiserslautern) and in the USA (College Park, Maryland). All Q-Labs offices hold an ISO 9000 registration.

Apart from our in-house expertise Q-Labs has a well established working network of academic and industrial experts. Q-Labs has close technical and strategic cooperation with ESI (European Software Institute, Spain), IESE Fraunhofer (Institute for Experimental Software Engineering, Kaiserslautern), SEI (Software Engineering Institute, USA) and the CIO (Center of Software Engineering of Gdansk, Poland). Q-Labs is an elected member of the ISERN (international Software Engineering Research Institute). Q-Labs is a founding member of the STI (Software Technology Initiative, Kaiserslautern) which aims to support SMEs in their software process improvement program. Q-Labs also participates in initiating the Swedish SPIN.

Q-Labs is also a partner to the ESPI (European Software Process Improvement) Foundation. ESPI is aimed at disseminating best practices and lessons learned in process improvements throughout Europe from a business oriented point of view. Other EU projects with Q-Labs involvement are SPIDER, ESTEX, REBOOT, PERFECT, SER and PIE (Process Improvement Experiment) under ESSI.

Q-Labs network of associates includes: Dr. Jesse Poore (SET, Cleanroom Software Engineering), Dr. Bill Curtis (TeraQuest Metrics, CMM and Software Process Improvements), Dr. Vic Basili (University of Maryland, Software Engineering Laboratory NASA-SEL), Dr. Hans Dieter Rombach (University of Kaiserslautern, experience factory), and Dr. Claes Wohlin (Lund Institute of Technology, quantitative analysis and certification).

Examples of specifically developed knowledge and experience areas of Q-Labs are:

**Software Process Improvement:** The corporation has analysed, enhanced, monitored, and improved a large number of software processes for industrial customers producing complex and large real time systems. Within this work CMM has been specifically used as a reference model, performing CMM assessments, setting up and driving improvement programs based on the IDEAL model. Results from the European research project PERFECT have also been applied. Furthermore, Q-Labs has supported the implementation of metrics to monitor improvements, using the goal/question/metrics approach.

**Cleanroom Software Engineering:** Q-Labs has worked since 1990 adapting, developing and transferring the technologies of Cleanroom Software Engineering to industry. When introducing new techniques like Cleanroom the introduction effort is done in close relation with the customer, based on a thorough analysis of the customer's software practices and on the specific customer needs.

**Human factors:** Q-Labs experience of technology transfer and process improvement has shown that even the best methods and processes do not generate full effect unless human factors are dealt with. Hence, many activities at our customers have been devoted to change management, team building, management vision seminar, achievement coaching.

**Architecture and Reuse:** It is being forecasted that a real breakthrough in software engineering will happen through architecture reuse (Boehm). Through Q-Labs participation in ESPRIT REBOOT project, competencies have been developed in reuse strategies, reuse management and process.

**Software Acquisition:** Another special business branch within Q-Labs performs audits and supplier evaluations of large software development projects and development organizations on behalf of a purchasing client. CMM is used as one of the reference models in these projects. Q-Labs also supports organizations purchasing software products in improving their purchasing process. Q-Labs has also played the role of external quality assurance for such organisations.



Q-Labs has a wide range of customers for example Ericsson (Sweden, Canada, Germany etc.), Robert Bosch, VOLVO, ABB, Alcatel, the Association of Swedish Engineering Industries, Siemens, The Ministry of Justice in Norway, Telia (Swedish PTT), Telenor (Norwegian PTT), and Swedish and Norwegian defense industry, just to mention some selected customers.

## Christophe Debou

Christophe Debou received an engineer degree in computer science from the Ecole Nationale Supérieure de Sciences Appliquées et de Technologie of Lannion (France) in 1990. From 1990 to 1994, Debou was working as research engineer and later as group leader at Alcatel Austria Research Centre (Vienna, Austria). He was the Alcatel project leader of the ESPRIT project **ami**® (Application of Metrics in Industry). He also participated in the industrial trial phase of the ESPRIT project COSMOS (Cost Management with Metrics of Specification).

In 1994, Debou moved to Alcatel Network Systems headquarters (Zaventem, Belgium) to become manager of software process and technology. He was also member of the Alcatel Alsthom corporate group supporting all business divisions world-wide in their effort to improve their software development. His major activities focus on organising software process assessment, supporting action plan definition, following up action plan implementation and co-ordinating common actions across the several development centres.

In 1997, Debou has joined Q-Labs GmbH (Kaiserslautern, Germany), a leading software engineering consultancy, as business area manager. In this role, he is investigating market potential in Central and Eastern Europe.

He published several articles in major conferences and journals in the area of metrics and quantitative approach to software process improvement. He is vice chair of the **ami** user group, member of the IEEE as well as consultant of the International Software Consulting Network (ISCN).

His fields of interest are software engineering, quantitative approaches to software project management, measurement, complexity metrics, data analysis, software process assessment and improvement, quality assurance.

## ANDERS GUSTAVSSON

Anders is presently vice president of Q-Labs. In this responsibility, he is member of Q-Labs Management Board which covers the three locations of Q-Labs in Sweden, Germany and the US. Before Anders was operating manager of Q-Labs GmbH (Kaiserslautern, Germany). He has coordinating all European R&D projects namely REBOOT, SER and PERFECT. He is also the Technical manager of the EC PERFECT project. Previous assignments at Q-Labs have included senior project management of quality assurance, process improvement, process modeling and introduction of reuse.

Prior to joining Q-Labs, Anders was a research scientist at Telia research responsible for the integration mechanisms of the SDL/SDT tool set; and information modeling and process modelling for large telecom applications. He has also worked with the Scandinavian Mjoelner project, aimed towards OO technology, following research into configuration management in OO incremental development environments at the department of computer sciences (Lund University). Anders is co-author of the Mjoelner Book, "Object-oriented Software Development Environments" published by Prentice-Hall, and has presented conference papers on configuration management

# A Concept for Managing East - West Outsourcing Business

Miklos Biro, Tibor Remszö  
*Sztaki, Hungary*

Christophe Debou  
*Q-Labs GmbH, Kaiserslautern, Germany*

Alec Dorling  
*IVF, Sweden*

Christophe Debou  
*Q-Labs GmbH, Kaiserslautern, Germany*

Richard Messnarz  
*ISCN, Austria and Ireland*

## **Introduction**

The objective of this paper is to discuss an effective solution to encourage the mutually beneficial cooperation between software and systems development organizations of Eastern and Central Europe and EU countries.

Further we will propose to look at strengths, weaknesses, opportunities and threats (SWOT) from the perspective of the possible exploitation of achievable financial, operating, marketing, and production leverages with East European software firms. The benefits for cooperating partners are the following for example: highly educated workforce, cultural proximity, relatively low cost. Nevertheless, cooperation is difficult to initiate because of perceived threats arising from the former neglect of the development of a quality culture.

The idea is to make use of ISCN as a coordinating agent with the role of assessing and improving the quality systems of potential Eastern European partners, reducing in this way the risks of other partners to cooperate with them.

## **The Concept of a Coordinated East - West Procurement Enterprise Partnership**

It is based on the assumption that the overall competitiveness of Europe in the software industry will increase through a better cooperation between Western and Central and Eastern Europe to take advantage of low-cost and highly educated resources. There is an increasing need for software development (e.g. on the short term, Year 2000 problem, EURO) in Western Europe but the current resources are not sufficient to respond to those needs (e.g. lack of 20000 software engineers in Sweden, decreasing number of students graduating in computer science in Germany). Through outsourcing, the CEE potential can be exploited in that respect. On the CEE's side, outsourcing is an

effective way to absorb modern technologies and to develop its own potential to help introducing high-tech into local industries.

Within the proposed concept ISCN aims at:

- 1) Establishing a network composed on the one hand of Western European companies (consortium members) and on the other hand leading CEE firms that will serve as basis for best practice dissemination in SW procurement and for establishing business relationships between western and CEE firms. The objective is to have all western nodes and the CEE nodes ready for operation by the end of 1998.
- 2) Launching awareness seminar/training in software procurement in both western Europe (from contractor/customer viewpoints) and CEE (from supplier viewpoints). At least, a section of ISCN 1998 will deal with this issue and at least one seminar per CEE partner country is planned to be performed. It is seen as a preparation activity in 1998.
- 3) Establishing a CEE expert skill and company capability database for procurement purposes

Two already existing infrastructure products will be put in place in cooperation between Sztaki and ISCN:

- ESD (expert and company database) which is a configurable database storing company services and experts skill profiles and providing an expert system functionality to select proper experts and companies based on skill and service data and on restrictions (such as language, salary, ...) Refer to <http://www.iscn.ie/esd>. This database will be populated with CEE companies and experts.
- NQA (Network based Quality Assurance environment) which is an intranet based quality assurance system providing quality documentation guidelines, an on-line quality manual, computer supported project administration, templates with industry examples and role plays for software development. refer to <http://www.iscn.ie/projects/nqa>. In the context of the project, NQA can serve as an internet supported quality environment for joint development between EU and CEE companies.

## **A Collaborative Technology Transfer Interface to the EU**

The initiative is a preparatory action towards increased procurement/business cooperation between western and central and eastern European companies.

This will finally lead to the foundation of a technology transfer organisation, jointly funded by all partners and regional governments (applications in regions will be brought forward within 1998) which will:

- use ESD as skill and service database
- use SPICE and BOOTSTRAP as a capability determination tool
- use NQA as a quality documentation environment for distributed teams
- use quantitative methodologies to analyse Eastern European products and software houses before inclusion into the service database
- help West European organisations to recruit people in case of establishment of site locations in Eastern Europe
- help West European partners to identify a proper East European outsourcing partner
- establish a product shelf of east European products which might be of value for Western Europe and which satisfy the European norms after an objective product evaluation

The major argument is to establish an organisation between West and East similar to that established between Europe and India in form of 3SE.

3SE (Software Services Support and Education Centre Ltd) is a coordinating organisation in Bangalore in India which has been promoted by the European Commission and the Government of India, to promote cooperation between the European Union and India in the field of computer software. <http://www.3seblr.soft.net/>

3SE offers the following services to European companies to help them leverage the Indian software industry:

- Identifying the right Indian partner
- Communicating an EU company's need for software to Indian companies
- Arranging European software delegation visits to India
- Promoting sector-specific European IT technology in India
- Conducting seminars in Europe on Indian business values
- "Gateway" services - Recruitment of people to help establish sites in India, Selection of Companies to provide services at low cost for EU companies

The contents of the database will be defined at the beginning of 1998 and population will be performed throughout the project with an objective of 50 companies and 100 experts.

Beyond 1998, it is envisaged that the partners will establish a joint enterprise (based on the 3SE model in India) to establish outsourcing opportunities between west and CEE.

## **Key Activities to Start With the Initiative**

### ***Activity 1: Identify CEE partnerships***

The objective of this task is to identify relevant CEE partners with which the consortium will co-operate locally.

### ***Activity 2: Survey in western Europe towards potential for procurement in central and eastern Europe***

The objective of this survey is to evaluate interest of western firms in procurement/outsourcing of software-intensive systems in CEE. Practices in procurement management will be as well investigated. Positive experiences will be documented in a case study and used during the final awareness seminars.

A questionnaire will be produced in the local language and send to typically purchasing department of private/public organisation and potential companies. A sample of interesting companies will be visited for in-depth interviews on experiences and needs.

### ***Activity 3: Survey of state of practices of CEE companies for procurement purposes***

With the support of local partners (through subcontracting), a survey will be performed to:

- Assess Current state of the practice of CEE in software development (based on requirements for successful procurement) and hence capability
- Identify potentially interested companies and related services

This is the first step towards populating the ESD database with company capabilities and services. Throughout the project, a continuous marketing actions will be performed to further populate the database.

***Activity 4: Establish model (work procedure, procurement practices) of an organisation to support procurement between western and CEE firms***

This task is aiming at preparing a procurement support organisation “model”. Such model will include items like:

- To select suppliers according to capability and risk assessment BOOTSTRAP and SPICE compliant methodologies will be applied by ISCN consultants
- NQA, the Network based Quality Assurance environment, will be promoted for the distributed cooperation support
- ESD will be installed at Sztaki's premises to store company services and experts profiles for further use in business partners matching

**SWOT (Strengths, Weaknesses, Opportunities and Threats) Analysis from the Perspectives of the Four Possible Levers of a Firm**

Levers are means used by a firm to multiply its resources. It is fundamentally the use of levers which can be accounted for the differences in profitability among firms. The four possible levers of a firm are the financial lever, the operating lever, the marketing lever, and the production lever.

Eastern Europe has a number of general strengths including a highly educated workforce able to assimilate new skills rapidly, and able to produce high quality goods at relatively low cost for export. For the same reasons R&D capacity is high as well. Large projects mean new opportunities for both foreign and domestic ventures. These projects are becoming urgent because of the limited possibilities of the earlier economic system.

Operating leverage is the relative change in profit induced by a relative change in volume. Because of the low operating costs, the Hungarian software industry has a high operating leverage, by consequent it can generate more profit than its less leveraged competitors as soon as its volume reaches a given level.

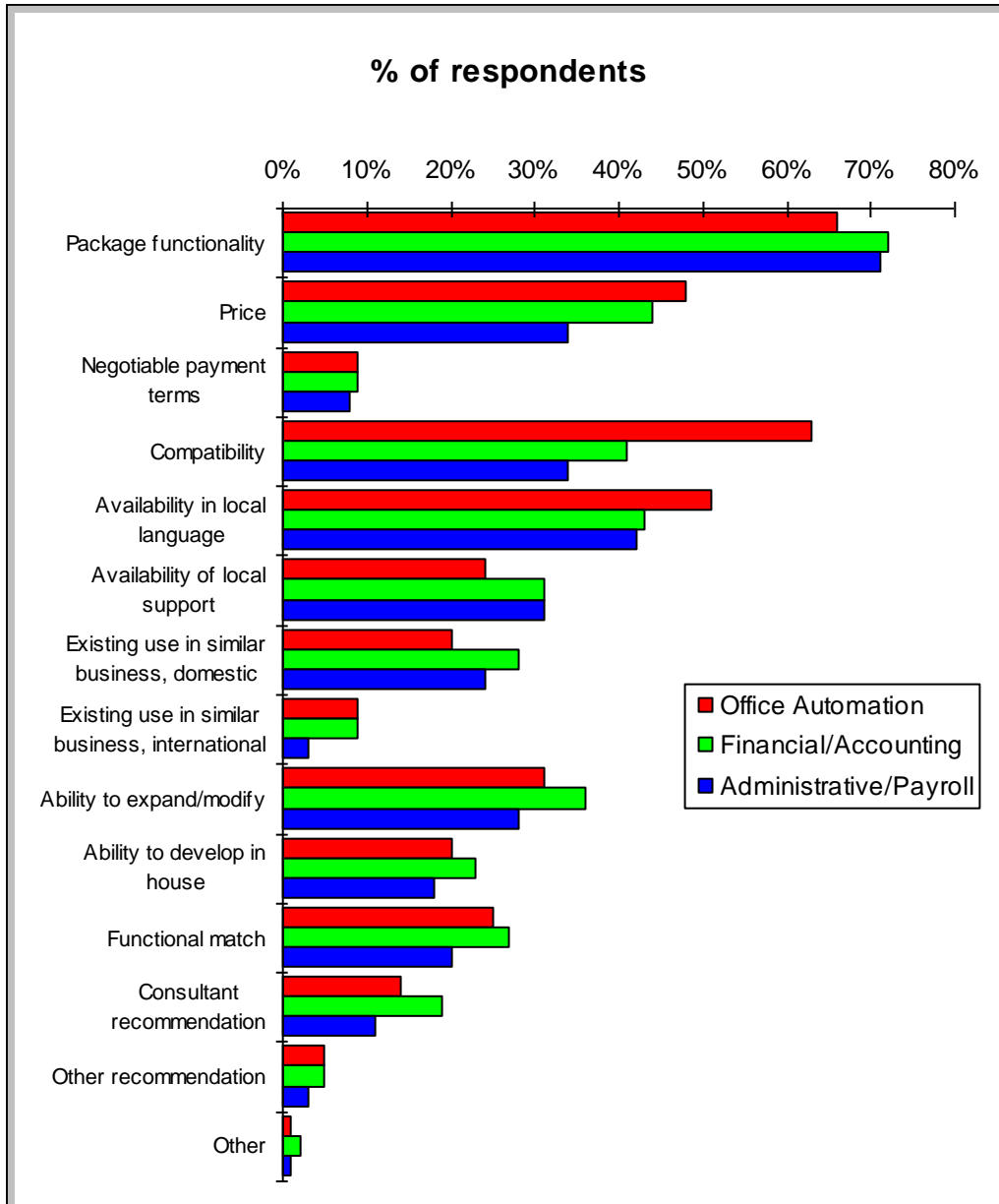
A weakness, already introduced in the previous section, is the relative lack of local managerial skills and experiences. This problem has impact on both the production and marketing leverages.

Production leverage is the rate of growth of profits resulting from cost declines. Production leverage can only be achieved if management is able to properly organize production. Quality management is an important part of this organization.

The two main ingredients of marketing leverage are higher prices and innovative distribution. The achievement of any of these goals requires advanced market management skills.

As far as production and marketing leverages are concerned, Hungary is making efforts in training managers to the necessary skills that were unheard of in the former economic system. The possibility of making use of financial leverage, that is having and exploiting debt capacity, depends on the advent of general economic recovery and lower inflation, which is a rather long-term process.

**Driving Forces of the Market**



Data source: Deloitte & Touche, IDOM

**The coordinating role of ISCN**

ISCN is an Irish firm whose key asset is a pool of experts who represent a wide range of approaches and methodologies allowing a synergetic combination of the skills most suitable to the specific requirements of its customers. ISCN is by consequent well positioned to vitalize a coordination model where mutually beneficial and reduced risk cooperations could be established with Hungarian and other Central and Eastern European software developing firms.

The coordination model is based on the following process:

- ISCN cooperates with the Computer and Automation Institute of the Hungarian Academy of Sciences (MTA SZTAKI) in establishing an expert skill database across Central and Eastern European countries using a Procedure Quality Manual for certifying experts.

- MTA SZTAKI uses the expert pool to evaluate the capability of Hungarian and other Central and Eastern European firms and to register those who have a capability maturity level above 2.5. This means a satisfactory level of cooperation risk for partners who may also obtain more detailed maturity profiles if necessary and agreed by all parties.

- ISCN and MTA SZTAKI promote the outsourcing cooperation and support the establishment of the corresponding contracts.

The above model makes it possible to exploit the opportunity of higher production leverage for Hungarian and other Central and Eastern European firms and of higher operating leverage for partner firms.

## Conclusion

The maturity level for quality software development of Hungarian firms is changing rapidly. ISCN is ready to play the role of coordinating agent in establishing mutually beneficial and reduced risk cooperations with Hungarian software developing firms.

Even though most of the analysis concerned Hungary only, it is clear that similar processes are going on in most Central and Eastern European countries. In order to satisfying the specific needs of its customers, ISCN is also ready to mobilize its relationships in these other countries with high software development potential.

The entire infrastructure is already in place:

- ESD is available
- NQA is available as Beta Release
- Sztaki has the necessary East contacts
- ISCN can provide the necessary West contacts
- East European firms (as the conference shows) are interested in offering their products and services through a defined window to the West European market

However, so far we cannot provide funding, but we offer all partners to use the existing Sztaki and ISCN infrastructure to support making a business interface.

For East - West cooperation please contact:

Dr Miklos Biro  
MTA Sztaki  
Budapest, Hungry  
Tel. +36 1 209 5270  
Email: miklos.biro@sztaki.hu

For information about the infrastructure technology and products please contact:

Dr Richard Messnarz  
ISCN  
Dublin, Ireland  
Tel. +353 87 231 5465  
Email: rmess@iscn.ie

## References

- [1] Biró,M.; Feuer,É.; Haase,V.; Koch,G.R.; Kugler,H.J.; Messnarz,R.; Remzsó,T. BOOTSTRAP and ISCN a current look at the European Software Quality Network. In: *The Challenge of Networking: Connecting Equipment, Humans, Institutions* (ed. by D. Sima, G. Haring). (R.Oldenbourg, Wien, München, 1993) pp.97-106.
- [2] Biró,M.; Sz.Turchányi,P. Systems of Decision Criteria Supporting Software Process Improvement. In: *DSS - Galore, Proceedings of the fifth Meeting of the EURO Working Group on Decision Support Systems* (ed.

- by M. Brännback, T. Leino). (Institute for Advanced Management Systems Research, Turku, Finland, 1995) pp.133-147.
- [3] Biró,M.; Remzsô,T.; Sz.Turchányi,P. Assessment and improvement of the quality of software processes from a decision maker's perspective. In: Proceedings of the VI. National Congress of the John von Neumann Society for Computer Sciences (ed. by I.Tóth). (John von Neumann Society for Computer Sciences, Siófok, Hungary, 1995) Vol.2. pp.671-678. (*in Hungarian*)
- [4] Biró,M.; Feuer,É.; Remzsô,T.; Sz.Turchányi,P. Business Decision Problems Supported by Software Product and Process Assessment. In: Proceedings of the ESI-ISCN '95 Conference on Practical Improvement of Software Processes and Products (ed. by T. Katsoulakos, R. Messnarz). (European Software Institute-International Software Consulting Network, Vienna, Austria, 1995).
- [5] Biró,M.; Sz.Turchányi,P. Software Process Assessment and Improvement from a Decision Making Perspective. ERCIM News (European Research Consortium for Informatics and Mathematics) No.23 (1995) pp.11-12.
- [6] Biró,M. IT Market and Software Industry in Hungary. Keynote paper at the European Observatory on Software Engineering CQS '95 Conference (Rome, 1995).
- [7] Biró,M.; Feuer,É.; Remzsô,T. The Bootstrap Institute: International Organization for Software Quality Assurance in the European Union. In: Proceedings of the IV. Quality Week International Conference (ed. by M.Pákh, A.Róth, T.Turos, Á.Hargitai). (Hungarian Quality Society, 1996) pp.133-136. (*in Hungarian*)
- [8] Richard Messnarz (ed.), Better Software Practice for Business Benefit: Principles and Experience. (PICO book).



# Learning to Improve - the Essential Ingredients

Håkan Wickberg  
*IVF, Gothenburg*

Alec Dorling  
*IVF, Gothenburg*

## 1. Introduction

Software is key to a large number of activities we perform every day in our working from using the Internet, withdrawing cash from cash dispensers, using a mobile phone or driving the car to work. More and more organisations, in all sectors of the economy, are becoming increasingly dependent on software, either as an integral part of their product or to support their business process. For these organisations, developing software effectively and efficiently is a key factor for business success. As Year 2000 and the introduction of the Euro grow closer we are constantly reminded how much software is important to the very infrastructure of European business life.

After 50 years of computing we still have a software crisis. Back in the autumn of 1968, the NATO science committee convened some top 50 programmers, computer scientists to plot out a course to escape this crisis. Nearly 30 years later, the problems in software remain the same [1]:

- 25% of all software projects are killed (Curtis, 1995).
- Companies are releasing products to their customers with 15% of the defects remaining in the product (Jones, 1996).
- Many companies are spending from 30% to 44% of their time and money on reworking software they have already written (Curtis, 1995).
- Companies meet their schedules only 50% of the time (Curtis, 1995).
- 55% of all projects cost more than expected (IBM, 1994)

Whilst the pace of technology and software systems grows almost unendingly, the basic single problem is that adoption of best practices has not evolved as fast. Traditionally organisations have introduced new languages and tools and adopted world-wide standards as a 'silver bullet' to solve the software problem. Organisations need however to pay attention to the introduction of best software practices taking into consideration the business, management and organisational issues as well as the methods and technology. Implementing software best practice can help continue to achieve customer satisfaction and gain a competitive edge. Modest investments in software best practice can produce significant business benefits.

The good news is that change is occurring. Over the past thirteen years, software companies have made significant progress toward understanding how to measure, consistently and quantitatively, their software development processes, the density of errors in their products as well as the programmers' productivity. In 1991, the Software Engineering Institute (SEI), based in Pittsburgh - USA, published

its Capability Maturity Model (CMM). The CMM has persuaded many programmers to concentrate on measuring the process by which they produce software, a prerequisite for an industrial engineering discipline. It is a benchmark which evaluators can grade the ability of a software organisation to create predictably software that meets its customer requirements. Software process improvement (SPI) initiatives are beginning to take effect and experience of implementing SPI is growing. The question is what has been learned, and what are the essential ingredients for success especially in small software developing (SSD) organisations?

Within SSD user organisations software process improvement activities have been seen as an activity for the big players in the market as they see this approach to be too far away from their business reality. Management in these user organisations do not acknowledge the evidence of cost-effective process improvement in bigger organisations, such as that reported from the SEI. Process improvement is seen to be too expensive, long-term oriented and too difficult to achieve. Decision-makers need to see hard evidence that improvement efforts applied to their business units yield a quantifiable return of investment in an appropriate period of time The European Systems and Software Initiative (ESSI), launched in 1993, has a group of actions directed towards Software Best Practice and has gone a long way to providing some of that hard evidence.

## **2. Software Process Improvement (SPI)**

Software process improvement is the action taken to change an organization's processes (to plan, execute, monitor, control and improve software related activities) so that they meet the organization's business needs and help it to achieve its business goals more effectively.

Software process improvement is best considered as a continuous process, where an organisation moves continually around an improvement cycle. Within this cycle improvement is accomplished in a series of steps or specific improvement actions to improve software practices. An important step in the improvement cycle, however, is the execution of some form of data gathering to establish the initial state, and subsequently to confirm the improvements. This is normally undertaken by performing a software process assessment using a defined model. The emerging standard on Software Process Assessment (ISO15504/SPICE) provides such a model [8]. Guidance is also provided for use in process improvement activities [11].

The guidance points to the following factors as being necessary to consider before embarking upon any software process improvement activity:

- software process improvement demands investment, planning, dedicated people, management time and capital investment;
- process improvement is a team effort & those not participating may miss the benefits and may even inhibit progress;
- effective change requires an understanding of the current process and a goal;
- change is continuous, not a one-shot effort & it involves continual learning and evolution;
- software process changes will not be sustained without conscious effort and periodic reinforcement.

The needs and business goals of the organisation determine the software process improvement goals that help to identify improvement actions and their priorities. Software process improvement is accomplished in a series of steps or specific improvement actions as shown in Figure 1.

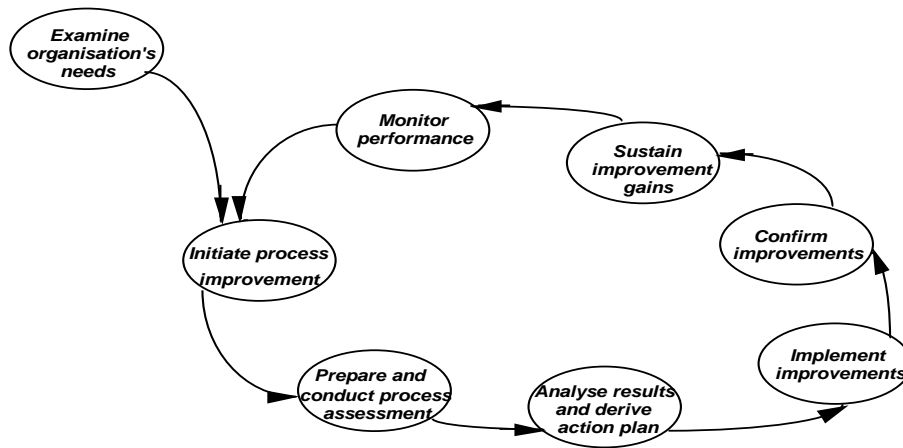


Figure 1 - Software process improvement steps  
(ISO/IEC DTR 15504-7 Guide for use in Process Improvement [11])

The executive decision to undertake improvement, together with the identification of an improvement programme budget and the main process improvement priorities, enable the improvement process to progress typically through the following steps:

- initiate process improvement;
- carry out a software process assessment;
- process improvement project planning with an action plan resulting from analysis of assessment results;
- implement improvements according to process improvement project plans;
- confirm the improvements;
- sustain the improvement gains by maintaining the new, improved level of performance until stability has been reached;
- monitor performance to continue the process improvement programme comparing results against the measurable goals of the process improvement programme plan.

Carrying out a software process assessment will help identify the current status of software development practice in the organisation. Current status can be evaluated against a recognized model showing progressive stages of implementation of good practices. This helps to identify potential areas for improvements upon which decisions can be taken and implemented. Finally, the effects of the actions can be measured and checked before repeating the cycle. A wide international community has adopted this kind of evaluation as a measure of process or organisational capability to provide predictable and reproducible results.

### 3. The ESSI Experience

The European Systems and Software Initiative (ESSI) has a group of actions directed towards Software Best Practice aimed at helping European organisations, in all industrial sectors to:

- increase their efficiency
- provide better quality
- provide better value for money

ESSI was launched in 1993, with over 100 MECU invested in more than 360 funded proposals. Over 300,000 organisations have been reached by ESSI. Here have been over 5000 organisations involved in Software Process Improvements through ESSI PIEs with over 210 events organized to disseminate the results.

The Process Improvement Experiments (PIEs) have provided the opportunity for organisations to demonstrate the benefits of process improvements through controlled, limited experiments. The PIEs are undertaken in conjunction with real projects (the baseline project) which forms part of the normal business of the organisation.

A PIE allows an organisation to try out new procedures, new technologies and organisational changes before taking decisions to replicate throughout the organisation. The lessons learnt are disseminated both internally in the organisation and also to a wider audience at the European level. Many of the results of PIEs are held within a repository maintained by the European Software Institute (ESI) [7].

The ESSI programme has produced a number of case studies showing that not only the efficiency and quality of software production and maintenance has been improved, but also that there have been clear business benefits. The following are referred too in a booklet on ESSI Pilot Case Studies available from the EC [6].

<b>Company</b>	<b>Results</b>
Engineering	60% improvement in accuracy of estimating costs, effort, duration etc.
BBV	6.5 times more efficient migration of applications programs to new platform
Claas	5 million ECU sales boost through better specification and software management
Datamat	time to market decreased and number of errors in products reduced by introduction of configuration management system
ENEL	18% cost reduction in project development by introduction of formal specification method
B & K	75% less error reports through systematic unit testing

Neither the software engineering approaches used nor the nature of the benefits achieved are peculiar to these individual companies. Their experience indicates, that by intelligent use of the large repertoire of management methods and software tools available, any software development operation can make significant improvements in what it delivers, in how soon it delivers it, in its costs of delivery, and in its customers' satisfaction. To achieve this however requires leadership and professionalism, a fact that should not be ignored.

There are a number of lessons learned by organisations having implemented software best practice under the ESSI programme:

- activities should be designed to meet the specific business needs of the company and to satisfy the needs of their customers, the first priority given too strategic business issues
- investment has to be focused on those areas which can deliver results
- there is strong correlation between practice and performance
- it is essential to ensure the support of senior management
- sufficient data is now available to construct a convincing cost benefit analysis
- results do not happen overnight, benefits will materialise in the mid to long term
- the results of software best practice cannot always be quantified easily, often there is no basis for before and after comparisons

- an incremental approach works best and contributes to the avoidance of risk, a portfolio of internal lessons learned should be built up
- the whole organisation should be aware of the importance of software best practice and its anticipated benefits
- training is a key factor in the cultural shift necessary for success

The ESSI programme has reached a good many organisations since its launch in 1993 with most of the companies participating being able to carry out improvement activities themselves. However many of the Small Software Developing (SSD) organisations (those with typically less than 20 staff) have found the procedures over bureaucratic and often time consuming, The needs of SSDs are perceived to be different and new approaches have been developed to help such organisations learn to improve. The essential needs and ingredients for such organisations are often different.

#### **4. Assisting the Swedish Software Industry to Raise Its Standards of Quality and Productivity**

The structure of Swedish industry is essentially hourglass-shaped: that is, polarized between a small number of large firms and a large number of small ones. Unlike Germany, there is no tradition of a Mittlestand, or layer of medium sized companies.

Sweden's engineering industry is responsible for about half the country's manufacturing production, exports and employment. Engineering skills underpin the global strength of many of Sweden's multi-national companies such as Ericsson, ASEA-Brown Boveri (ABB), Volvo, Atlas Copco, Saab, Electrolux, Scania and SKF. Other well known companies in Sweden include Bofors, Telia and Astra. Sweden is in the club of industry nations who are top investors in industrial R & D. Many suppliers, mainly SMEs, depend on these very large companies for their work.

The Swedish Institute for Production Engineering Research (IVF) was established in 1964. It is owned by a non-profit foundation (Stiftelsen for Verkstadsteknisk Forskning) which is jointly owned by the state through the Swedish National Board for Industrial and Technological Development (NUTEK) and the engineering industry, through the Swedish Federation of Engineering Employers (VI). The Institute is deeply involved in networking with industry and the state and is recognised nationally and internationally as a centre of excellence [5].

IVF has been the major force behind most of the technology transfer programs in Sweden for more than a decade. People from almost 30 technology areas are actively working together and exchanging knowledge and ideas about the process. One of the main reasons why IVF is so successful in this business is because of its tremendous skill in not only being able to fully understand these problems but also being able to combine the knowledge inside IVF to solve the problems. IVF sees itself largely as occupying the middle ground between the capabilities of companies and consultants on one hand and that of the university researchers on the other. Informally, IVF management compares IVF activities to a three-stage rocket, with the respective stages being: technology monitoring and acquisition; development; technology transfer.

IVF has along history of working with small companies. IVF operates an SME service on behalf of NUTEK – a programme of advice and short-term consultancy. IVF also administers an Innovation Relay Centre (IRC) for the West of Sweden [2]. The IRC is a direct link to technology and innovation across the European Union. Its primary objective is to promote the spirit of innovation in European companies and research institutions – especially Small and Medium-sized Enterprises (SMEs).

In 1993, IVF established an internal group focused on software engineering activities aimed at helping software development units within manufacturing industry. In 1996, the Centre for Software Engineering was formed with a wider mission 'to assist the Swedish software industry to raise it's

standards of quality and productivity and to be globally competitive’.

To achieve the mission a number of strategies are adopted:

- to be a competence centre and primary source of information, help and advice to the user community
- to be a collaborative partner in research, development and technology transfer projects throughout Europe
- to identify, promote and support the adoption of international standards and best practice in organisations
- to accelerate the process of technology transfer and to promote pilot projects in organisations, especially small to medium size enterprises (SMEs)

The Centre is currently involved with a number of initiatives to help the Swedish software industry improve software best practice with particular emphasis being placed on helping SSDs. These initiatives are:

- the ESSI Task Force 97 funded by NUTEK
- the Technology Transfer in Software Engineering (TIP) project funded by NUTEK
- the Software Process Improvement in Regions of Europe (SPIRE) project funded by the EC ESSI programme [9]
- the Small Company Action Training and Enabling (SCATE) project funded by the EC ESSI programme [10]
- the SPICE project activities in Sweden funded by NUTEK [8]

#### ***4.1 ESSI Task Force 97***

In previous ESSI call for proposals Swedish companies have fared poorly in Process Assessment and Process Improvement Experiment (PIE) proposal evaluations. In order to provide support to applicants in the June to September 1997 call, IVF in conjunction with NUTEK, set out to establish an ESSI Task Force 97 to assist companies in their proposal writing.

The Task Force aimed to help around 10-15 Swedish companies especially SSDs to submit proposals. A team of seasoned EC proposal writers and evaluators were assembled to help mentor and support companies in the proposal writing phase. More than 35 companies made contact which resulted in 13 PIE proposal submissions. The majority of applicants were SSDs.

The proposal writing actually turned out to be a learning experience for the SSDs. In writing the proposals they gained a deep insight as to how process improvements should be planned and directed towards business needs. The mentoring and assistance of the Task Force members proved invaluable.

#### ***4.2 TIP Project***

The TIP (Tekniköverföring Inom Programvarutekniksområdet - Technology transfer in Software Engineering) is a national programme. Its purpose is to increase the software engineering capabilities of SMEs.

The smaller companies have a considerable need for objective information in order to be able to invest sensibly and effectively and to enhance their knowledge and competence. This is particularly important in connection with the choice of development tools, as substantial sums are involved, coupled with extensive work in transferring old and new materials to the new environment.

Customer companies' requirements for a systematic method of working have also led to several software companies introducing quality systems, generally based on TickIT. The large companies have experienced a need for more extensive improvements, and are now involved in ongoing

improvement programmes, such as CMM, which serve the joint purpose of indicating the company's maturity level and indicating where new improvement activities are required. This is too complicated for the smaller companies, so a new view of their systems is essential.

The TIP project addresses some of these problems providing for:

- demonstrating tools, running seminars, courses and technology campaigns
- providing quick analyses (process assessments) followed by expert assistance or mentoring activities on process improvements

The mentoring activities are supportive of the SPIRE project activities.

### ***4.3 SPIRE Project***

SPIRE is an EC ESSI project with partners from Sweden, Ireland, Italy and Austria. SPIRE's objectives are to lower the barriers preventing SSDs from successful participation in SPI by:

- raising awareness of SPI benefits among decision makers and change agents in SSDs
- educating participating SSD managers and staff in practical SPI skills
- helping SSDs to maintain momentum in carrying through their improvement plans

The experience and results generated and disseminated by SPIRE are expected to have a major impact in raising the awareness of the benefits of SPI in a significant proportion of the 100 000 or so European SSDs.

Experienced mentors guide SSDs through an assessment of needs, the preparation of a sound plan for a cost-effective small SPI project (funded by SPIRE to a maximum of 15K ECU) , implementation of the project, and evaluation of results. The experience gained in the most successful projects will be published as short case studies aimed at decision-makers in SSDs in 4 languages (German, Italian, English and Swedish). Data from all the projects will be gathered in a standardized way, to permit analysis from which valuable lessons regarding best SPI practice for SSDs will be derived and published as a report. The results will be disseminated on paper, electronically and through workshops.

A mentor can be compared to a football coach. He provides a company with practical and moral support. A mentor will work with a company as an expert, as a discussion partner and a sparing partner. A mentor's relationship with the company is one of confidential co-operation.

In its mentoring activities IVF has adopted the Synquest tool for use in an SSD environment [4]. Synquest is based on a structured questionnaire, which embodies all relevant aspects for the evaluation of software processes. Synquest is used in a self-assessment mode and uses a guided questionnaire. IVF mentors sit beside a small number of SSD staff whilst they complete the questionnaire clarifying and discussing questions as they are answered.

Synquest provides a comprehensive system of help texts, which are based around recognised best practices. The help system not only explains how questions should be interpreted but also provides the necessary criteria for evaluating state of practice. Once the questionnaire is completed Synquest calculates the process quality of the organisation or project. The results are displayed in the form of a highly lucid set of charts.

### ***4.4 SCATE Project***

SCATE is a training programme devised to help small organisations improve the management of their software development. SCATE's objectives are to:

- teach practical techniques to the team member acting as ‘champion’ of process improvement
- enable the ‘champion’ - through group interaction - to acquire the skills and attitudes that will change the organisation
- provide the ‘champion’ with the opportunity to share experiences with the champions from other organisations
- foster synergies in the local community and create new opportunities for all

Training includes different styles of presentation (lecture, workshops, role playing, working group activities, presentations based on homework, competitions between team etc.). Every session has a different format to keep interest alive for the 9-month training programme. Each organisation is provided with on-line support through email/fax/phone.

The ‘champion’ who is committed to gaining the real benefit will leave the programme with:

- excellent skills in working with people to support planned change
- the basic knowledge of software process improvement including the basics of CMM level 2 and a higher understanding of CMM level 3 Key Process Areas
- the confidence to practice and help others to practice on a regular basis

IVF is acting as a regional co-ordinator in the project for a number of Swedish SSDs participating in the project.

#### ***4.5 SPICE Project***

The international SPICE project is supporting the development of a new International Standard on Software Process Assessment. The project has organisations from over twenty-five countries contributing. The SPICE project is undertaking extensive field trials in parallel with the Standard’s development. The Standard is set to become in the software community as important as ISO9001 has become in the general community.

IVF is actively contributing to the initiative in a number of ways through:

- the SPICE international project management
- the Swedish Local Trials Centre
- national management and co-ordination activities on behalf of the Swedish Standardization body through ITS / AG7 (Software Engineering)

IVF assists companies wishing to participate in the SPICE trials by providing them with the necessary briefings and initial assistance. It is also:

- running awareness events around Sweden
- developing a SPICE tailored model and method for small company use
- providing access to suitable tool-sets to increase the efficiency of assessments

As part of the SPICE project infrastructure, Local Trials Centres have been established in Europe. IVF is the nominated centre for Sweden. This is a very important task, being able to mentor and provide assistance to organisations wishing to take part in the SPICE trials. IVF ensures the data collection of assessment results and briefing of trials participants and undertakes liaison with the Regional Trials Centre.

More and more demands for executive briefings on SPICE in Swedish companies are being made as awareness in SPICE grows. These executive briefings are an important part of getting companies interested in software process improvement, getting them to take part in the SPICE trials, and signposting them to the Centre’s mentoring and support programmes



## 5. Experience with SPI in Small Software Development (SSDs) Organisations

In the work with SSDs IVF has noted that SSDs encounter special barriers to successful SPI, which our technology transfer programmes seek to address.

Firstly, SSDs are less aware than larger firms of the benefits of systematic SPI aligned to their business needs. They frequently hold the view that process improvement is too expensive and difficult to be cost-effective for all but big companies. SSD decision makers are unimpressed by evidence of cost-effective improvement in much bigger firms, such as that published by the US Software Engineering Institute. They need to see concrete results achieved by their peers, applying resources they could find themselves, before changing their view.

Secondly, even if SSDs recognize that their software processes do not meet their business needs, they often do not know how to improve. They lack the in-house skills and experience to determine and implement an appropriate plan, which is typically available in larger companies. They cannot afford the management attention, staff time, or money to be trained or find an appropriate consultant. As a result at best any attempt at SPI is delayed and at worst no action is taken at all. This has probably contributed to under-representation of SSDs in the ESSIE community to date. Mentoring is an effective approach to overcoming this barrier: a few days of on-site advice and tutoring (say 5-10), from an appropriate mentor, who combines facilitation with business skills and software process expertise, can show an SSD how to assess needs and plan and implement a suitable improvement plan. The company is educated and launched on an improvement path, and the mentor also learns from the experience.

Thirdly, because of their size, day-to-day operational crises, financial constraints and changing priorities are more likely to drive an SSDs improvement plan off course, or cause it to be abandoned. This cannot be avoided in general, because it is in the nature of SSDs, particularly those with immature processes, but the risks can be reduced by creating a network of improving SSDs who exchange experience, and support and inspire each other to maintain momentum.

IVF has tried to lower these barriers preventing SSDs from successful participation in SPI by:

- raising awareness of SPI benefits among decision makers and change agents in SSDs

This has been achieved by running awareness events, company visits and advertising in the industry press.

- educating participating SSD managers and staff in practical SPI skills

This is usually achieved by experienced mentors, who will firstly show managers and staff of participating SSDs how to assess their own needs and plan a modest improvement project, secondly guide them through implementation, and thirdly help them to evaluate the results. This form of 'on-the-job' training has been shown to be very effective.

- helping SSDs to maintain momentum in carrying through their improvement plans

This will be achieved firstly by providing supporting funds for a mentor, secondly by the influence of the mentor on SSDs in the use of best practice in project planning and implementation to achieve milestones and deliverables, and thirdly by bringing the SSDs together at workshops where they can exchange experience, seek solutions to common problems, and be motivated to emulate each other's successes.

## 6. The Importance of Soft Factors in SPI

Traditionally software process improvement (SPI) activities have been mainly focused on the further development of the software process and technology. The reason for this focus is that process and technology are more easily to grasp and correspond better to the mindset of software professionals than organizational and human factors (soft factors) like leadership, communication, knowledge transfer, social interaction etc.

Industrial research studies indicate that resistance to change of human behavior and a low degree of organizational maturity of software organizations are major factors, which inhibit a better yield of software process improvements. Making organizational and human factors explicit by measuring helps user organizations to focus their improvement efforts on key issues in software development. For example, there are:

- social interaction amongst software staff and/or clients
- understanding of the organization's policies and culture
- view of staff regarding training and alignment of its personal goals
- acquisition, exchange, interpretation and transferring business relevant knowledge within organizations
- teamwork as a key discipline in software development

Making the right management decisions about software development process improvement requires an awareness of the benefits that can be expected from different process improvement initiatives. Failure to invest in the most effective areas can result in disappointing results, disillusionment, and a reluctance of executive management to support continued investment in improvement of software development processes and improvement of the way the processes are being deployed in terms of daily practices.

By analysing both the software development practices and the performance levels of several hundred software development organisations throughout Europe, IBM has identified the correlation that exists between the different practices employed and the performance levels achieved.

The project, initiated by IBM in late 1994, was launched to benchmark software development organisations throughout Europe. The results of the IBM benchmark of software development practices indicate that practices that correlate most strongly with performance are non-technical [3]. They relate to the overall culture of the organisation and the approach to human resource management. In an industry that, by its very nature, is technology-based there is perhaps a tendency to overlook the fact, that software development is still primarily a human-centric activity. One example of a key finding is that employee morale has a significant impact on the way an organisation is performing. Organisations that encourage and reward innovation and entrepreneurial behaviour achieve significantly higher levels of employee morale. The way that we manage software development people will therefore have a substantial impact on their level of performance.

The successful software development organisations have recognised this fact. However, many organisations will continue to look for technology solutions alone as the answer to their problems. If management does not recognise the importance of these "soft" factors, they will be unable to look critically at how they are managing the organisation. The overall performance of the organisation will remain limited, as the key performance enablers remain unchanged. The practices that are weakest in their correlation with performance are CASE tools, the use of higher-level languages, and testing effort in terms of time and resource spent.

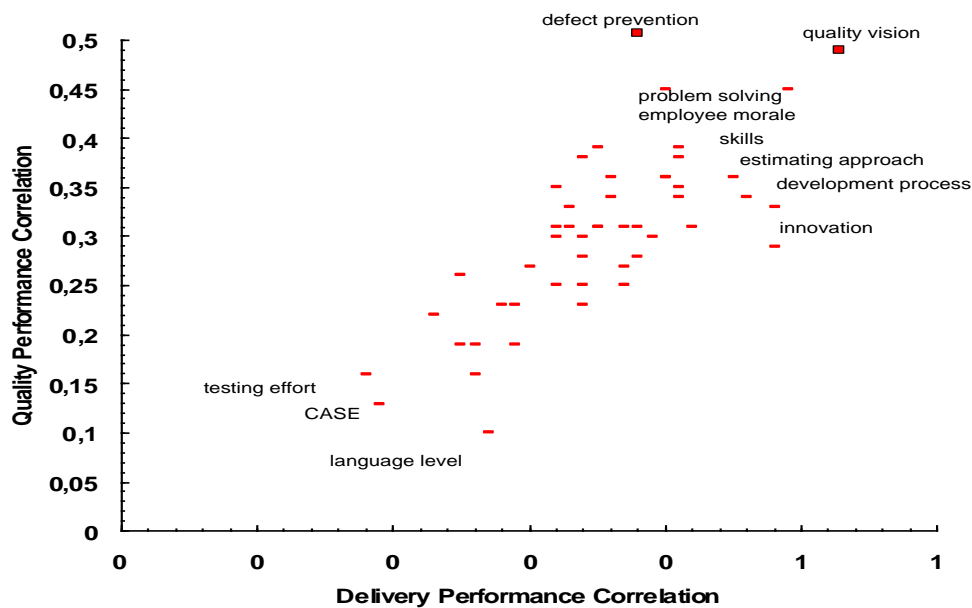


Figure 2 - Correlation of Software Practices with Quality and Delivery Performance  
(Re-produced with the permission of IBM Europe)

In October 1997, IVF in co-operation with IBM Europe, has launched a Swedish National Benchmarking Study. The questionnaires have been sent out to over 5000 software professionals. Each company will receive a benchmark of its practices against the European benchmark. In addition, in late February 1998, we will gain a picture of the Swedish national situation against the European average. This will be of significant benefit to IVF in planning its future activities to help the Swedish software industry.

From experience with helping SSDs at the low maturity end of the scale, IVF has a firm belief that in order to accelerate SPI in organisations from level 1 to level 2 on the CMM maturity scale for instance, it is important that organisations are made well aware of the soft factors. With this in mind IVF is working to define an organisational learning diagnostic which can be used in conjunction with a software process assessment method or model compatible with SPICE.

SOKRATES is a diagnostic self-assessment instrument to evaluate the capacity of software companies for effective organisational change. It can determine key factors that influence the uptake of best practices and change of behaviour. Each organisation has its specific characteristics and different business requirements resulting in a unique position to start organisational improvement actions.

The SOKRATES Diagnostic provides a comprehensive overview of the organisation as perceived at various levels in the company, a comparative analysis of the organization's synergy of three levels of people (executive, management, and staff) and a benchmarking option. The SOKRATES Diagnostic measures 34 critical success factors grouped into 4 aspects of an organisational framework - Strategy, Organisation, Human Resources and Knowledge. The assessment process consists of four phases: Briefing, Diagnostic, Analysis, Feedback and Action Planning.

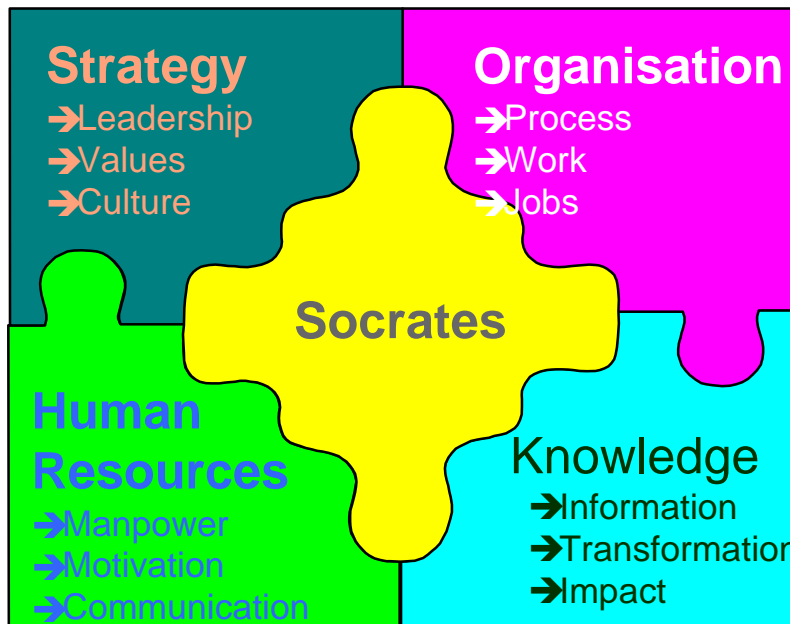


Figure 3 - The SOKRATES Diagnostic Model

It is hoped that experimentation with SOKRATES will further yield data and experience that will be of interest and use to those performing software process improvement in the millennium. For providing the necessary skills and experiences, it is expected that there

will be a growth in the use of mentors to facilitate on-site advice and tutoring. Selected experts in software process improvement acting as mentors showing user organisations how to correctly assess the organisation, interpret the evaluation data, planning and assisting in the implementation of tailored improvement plans.

## 7. References

- [1] A Business Case for Software Process Improvement, A DACS State-of-the-Art report, 1996, <http://www.dacs.com/techs/roi.soar/soar.htm>
- [2] Innovation Relay Centre - Western and Southern Sweden, [http://www.ivf.se/irc/irc\\_w-s.htm](http://www.ivf.se/irc/irc_w-s.htm)
- [3] IBM's Software Development Benchmark: A Powerful Competitive Edge Through Software Process Improvement, <http://www.uk.ibm.com/software/benchmark/paper.html>
- [4] Synquest, <http://www.synspace.com/>
- [5] Howard Bush, Michael Hobday, John Bessant, Erik Arnold, Robin Murray: Technology Institutes: Strategies for Best Practice - Chapter 3: Sweden - integrated policy-making, International Thomson Business Press 1996, ISBN 0-415-125030
- [6] EC ESSI Office: The Business Benefits of Software Best Practice: Pilot case studies
- [7] ESSI VASIE Software Best Practice Repository, <http://www.esi.es/VASIE>
- [8] SPICE project, <http://www-sqi.cit.gu.edu.au/spice/>
- [9] SPIRE project, <http://www.cse.dcu.ie/spire/>, <http://www.ivf.se/tekomr/progtek/spire>
- [10] SCATE project, <mailto:scate@catalyst.ie>
- [11] ISO/IEC DTR 15504-7 Software Process Assessment -Guide to Process Improvement, 1997

# Best Regional Technology Transfer - A Learning Concept for Innovation Management Organisations

Bernhard Posch, Christine Stöckler  
*APS, Graz, Austria*

Richard Messnarz  
*ISCN, Dublin, Ireland*

## The White Paper on Education and Training

At first sight every paper on training and innovation issues published by the Commission seems to be very boring. And many of them are. Some however are explosive in its effects if they were implemented only partially. At the same time they can make you wonder whether they are meant seriously.

Both is true for the White Paper on Education and Training. It appeared parallel in time with the announcement of the European Year of Lifelong Learning in 1996. Is it not a contradiction in itself that this principle was tightly framed in a time span reaching from January 1 to December 31 of one particular year? Why were we not talking of 1996 as the **first** European Year of Lifelong Learning?

BESTREGIT is an example that all the same it is possible to gear papers into action and that the Learning Society and Lifelong Learning are not hollow phrases but have to do with real life, in this case the lives of my staff and me.

BESTREGIT is a European project funded under the LEONARDO da Vinci Programme and coordinated by our organisation APS - an Austrian service organisation in the area of European programmes for innovation, technologies and training.

The White Paper claims education and training as the latest means for tackling the employment problem or more generally, the problem of the competitiveness of industries and services. Clearly distinguishing between responsibilities of the Commission and the Member states it formulates 5 ambitious objectives

- encourage the acquisition of new knowledge
- bring school and the business sector closer together
- combat exclusion
- develop proficiency in three European languages
- treat capital investment and investment in training on an equal basis

My talk will deal with some of these objectives, but mainly focus on the first objective and deal with the question of acquiring new knowledge in the context of technological innovation. Therefore it will also refer to the Commission's First Action Plan for Innovation in Europe, this one following the publication of the Green Paper on Innovation and the reactions to it.

Above all, for understanding a learning concept for innovation management organisation it is necessary to give you

- the general background
- an exemplary insight into the purpose, the structure and eventually also the common mistakes of innovation management organisations being very often a result of their structure and their history

## **Background**

The service sector for companies is rapidly growing because companies have to deal with more and more complex problems like environmental issues, an increasing need to cooperate with research institutions like universities and the introduction of information technologies. There are, however, hardly any training possibilities for these intermediaries that are responsible for the technology and innovation transfer between larger and smaller companies and between universities and companies. This is due to the fact that knowledge about technology transfer is uncodified, tacit knowledge that cannot be simply taught and learnt.

The jobs for the technology and innovation transfer intermediaries are undergoing two changes: new jobs are created, the old jobs are changing parallel to the rapid changes in the technologies used.

Paradoxically innovation management institutions themselves have difficulties in adapting to the industrial change and the globalisation of the markets and often work with obsolete methods and tools, especially in the field of organisational development and information transfer.

In a way innovation managers are like teachers. They always think they know better. They have the best solutions for others but none for themselves. They live in chaos. They are hypocrites as described in the bible only the other way round. While the former tell the people to drink water and drink wine themselves, innovation managers water down the hard drinks they offer to their clients within their own organisations.

And this is the starting point of a project like BESTREGIT - Best Regional Technology Transfer. And this is also an opportunity to say what BESTREGIT is not.

### ***What BESTREGIT is not and what it is***

BESTREGIT is not a software project but it has to bring together powerful computer tools to be able to work together effectively. In the long run service providers in Europe working on the same issues could establish a virtual service organisation. Having said this we are of course in a more or less complex software technology problem which is called configuration management.

Bestregit is among other things a project about self-criticism and bravery. Most of the institutions do not dare to make their processes so transparent because they think it makes them vulnerable. But we want you to learn from our deficiencies and mistakes. BESTREGIT should become a concrete example of Eastern/Western cooperation by giving our friends the chance to start at a higher maturity level in the setting up of their innovation agencies.

Before looking into the future let us take a brief look back into history:

### ***The Birth of an Innovation Management Organisation***

Once upon a time there was a European Action Programme called COMETT. Its mission was to

further the cooperation between universities and industry. This does not sound extremely exciting, but the programme installed an intelligent infrastructure, so called University-Enterprise Training Partnerships. APS was one of about 200 all over Europe. And the ingredients were concentrated in a way and of that sort we still can find in all European initiatives. A regional base bringing two sectors - universities and industries - together, staffed by young ambitious people, eager to learn and not only to make money from Yuppie consulting, a strong network of homogeneously working partnerships and lots of efficiently organised mobility projects between industry and university involving students and experts.

The programme seemed to be a success but for some reasons or others the network was eliminated and is now starting again within different initiatives and General Directorates. Amidst these troubles APS was growing because the public authorities in Austria could see the value of an independent and autonomous, suprainstitutional organisation working for the sake of SMEs.

Gradually our organisation is turning in what we call an innovation management organisation because we do not only inform our members and clients about European programmes but are offering a whole package of funding possibilities after having analysed the organisation and the technological needs and have started a long term cooperation with the SMEs in our region.

Innovation transfer means helping companies with the acquisition of an appropriate technical and organisational knowledge whose use allows its owner to achieve an expected goal.

To give an example: partner search is one of our core activities because forming Strategic Alliances is a main factor for innovation. Coming from the consulting on R&D programmes it took us a while to realise that knowledge sharing (joint R&D) to reduce time to market is only one form of an alliance, another important one being allying with local partners to access new markets.

It seems to be quite characteristic that organisations like ours work in an intersection of private and public interests and this is also reflected in their working approach.

### ***Innovation management organisations often work on a not for profit basis***

Politicians naturally are very interested that their ideas are supported by adequate structures. As most of the states are undergoing a severe budget crisis the tendency to outsource certain services increases. Very often these new support structures are working on a not for profit basis which leads to a number of results wished for as well as to a number of problems.

#### **Positive effects:**

- organisations are more open to innovative ideas and help on ideas that cannot immediately be materialised
- their focus is on good performance and not on making money fast, although these two components should go together in the private sector as well - they often do not
- It is a balancing act, but with some luck and some intelligence they can achieve a relatively autonomous position which would be impossible for departments within ministries
- having less bureaucratic procedures to follow they can concentrate their efforts on clients and their requests

#### **Negative effects:**

- they lack the back up of big organisations concerning access to resources and know-how

- they are mostly understaffed and have to struggle to achieve a critical size
- the quality of their service depends very much on the knowledge and experience of a very limited number of persons
- too much of their time is wasted by convincing their sponsors in continuing to give them funding
- having contracts with public institutions they have the duty to deal with any inquiry including the most pointless suggestions and hopeless clients

## ***Innovation and Organisation***

According to the First Action Plan for Innovation in Europe innovation is dependant on a management strategy capable of anticipating needs, monitoring technology, controlling lead times and costs, promoting flexibility, cooperating with external centres of expertise amongst other things. So called agile enterprises which are capable of implementing such a strategy appear to be the most capable of coping with the demand of innovation. To supplement the efforts of Member states and to help industrialists meet such demands, the Commission promises to play a dual role: it will implement a system of comparative evaluation (benchmarking) to assist enterprises in identifying internationally proven factors of success, and it will provide support for management training in innovation.

BESTREGIT shows that there was no need to wait for a top down approach. Fulfilling the latter promise LEONARDO da Vinci funded a bottom up approach drawing heavily upon the experience of the ESSI PIE projects.

### **Goals**

BESTREGIT aims at substantially contributing to the identification and acquisition of new skills for people working in the service sector by developing a well-structured, interactive training package that bases on models of best practice from 3 European regions, by testing them in these regions and two other countries. The countries involved are Austria, Ireland, Spain, Italy and England while the organisations involved include companies, professional bodies and universities during the training development process. The cross organisational approach reflects that innovation transfer is not a linear process: : scientific discovery - technological application - industrial application, nowadays it is a pervasive process that includes smaller companies, public and private strategic operators such as local authorities, entrepreneurial associations, development agencies etc.

### **Who is BESTREGIT for?**

Beneficiaries are those innovation transfer intermediaries that have not specialised in one field of innovation transfer but have to cover a broad spectrum of technology branches and needs requirements in their daily activities. They may work inside a company as proprietors of SMEs, human resources or R&D managers or in private, semi-public or public interface organisations of companies, universities, local, regional or national authorities.

### **General structure of BESTREGIT**

The trainings and the workshops will be developed in three stages:

- The first is to establish three innovation transfer institutions as learning organisations
- The second is interaction of five European countries in training networks to improve the material
- The third is industry feedback to shape professional training material

Thus the structure follows the crucial factor for innovation and this is the link between research (the production of knowledge), training, mobility, interaction (the dissemination of knowledge) and the ability of companies to absorb new technologies and know-how.



### ***BESTREGIT Products***

- a best practice model of a regional Austrian technology transfer organisation
- a best practice model of a regional Spanish technology transfer organisation
- a best practice model of a regional Irish technology transfer organisation
- a guideline for the establishment of such best practice models published and available on WWW
- at least 3 one-day tutorials (the A, E, or IRL model)
- a course in English with
  - one module about guideline for the establishment of best practice models
  - one module about the regional Austrian model
  - one module about the regional Spanish model
  - one module about the regional Irish model
  - one module about the workshop: How will this work for me?

### ***Dissemination and transfer of results***

The project foresees broad dissemination, starting immediately after the first year with leaflets, 10 regional field test, 6 conference tutorials and 2 train the trainer workshops and production of the training material in 4 languages (English, German, Italian and Spanish). The structure of the multiplayer partnership guarantees the widest possible dissemination: WIFI Steiermark represents 30 000 companies, APS works closely together with 15 regional technology transfer partnerships and like FUEVA is a partner of the 52 European Innovation Relay Centres, TU Graz and HIBERNIA work in the SOCRATES network, CTA and ISCN have been partners of ESSI and ESPRIT projects, CRACA represents 75 000 SMEs in the Veneto region, Italy.

We sincerely hope that we can start with dissemination activities in the Eastern and Central European states very soon as well.

### ***Impact***

We expect the following impact through our dissemination activities

#### Business

companies will get more effective service by technology transfer intermediaries trained in their business and companies themselves will be in the position to organise their innovation process in a better way

#### Training

novel training offer for the growing service sector - no trainings offered so far in this area  
companies and innovation transfer institutions will become learning bodies by establishing best practice models  
training approach is completely interactive and practice based

#### System/structure

The two different worlds of business and training become an entity - this impact reflecting the second objective of the White Paper on Education and Training  
to bridge the gap between inventors and innovation transfer managers  
Regional models are transferred European wide  
Technology intermediaries will be identified as important target group for public and private training providers

The chances for achieving impacts in this areas are high because the partnership comprises multiplayer

regional organisations with close contact to national authorities.

### Evaluation

Evaluation is a separate work package within the project management and will be carried out throughout the project by a senior staff member (with more than 10 years experience in technology transfer between universities and companies ) of the Technische Universität Graz which itself is not involved in the development of the training work packages. The evaluation criteria will address training and project progress criteria.

### ***BESTREGIT and its link to Process Improvement Experiments***

There are many interesting questions in the world. One is: 'Is there life after death' which is not a topic for this conference. Another one is: 'Is there a world beyond PIEs'.

Yes there is but in some mysterious ways even BESTREGIT has again to do with PIEs

It is common knowledge that technology is well ahead of organisational structures in industry

It is common knowledge that there are many people around who want to tell us that we have to start measuring our business activities, especially in software industry.

It is also common knowledge that most of us feel reluctant to this demand, arguing that it is a good idea only in theory, but too bureaucratic for real business life. The additional work factor - once things have been streamlined the context will change - is the main prejudice held against it

The Best-Regit project mainly involves non-IT partners from regional innovation transfer offices, but it re-uses an approach from IT programmes called PIEs (Process Improvement Experiments) in which innovative ways of work are experimented to find best practices. The best practice experiment follows these steps:

#### Phase 1: Establishment of a Learning Culture

- outline and compare the business goals of the involved regional innovation management organisations and to jointly work on a translation of business goals into operative and process oriented goals (which action lines are to be established to achieve which business goals ?)
- outline and compare the teamwork and organisational processes of each involved regional innovation management organisation, *using a common notation* to describe roles, teamwork communication, and workflows allowing each partner to understand the other partner's roles, teamwork approach, and established workflows.
- run a workshop with Austria, Ireland and Spain at which each partner presents his goals, organisational architecture, method of work, and infrastructure approach *using a common notation*. The common notation forms the basis for comparison and each partner acts as a consultant for the other partner. The goal is to prepare the partners' know-how in a pragmatic way which allows to exchange experience and to think jointly about an optimum structure of an innovation management organisation.
- establish a questionnaire for goal and teamwork analysis allowing each partner to benchmark against the attributes and checkpoints defined in the questionnaire. This leads to an identification of strong and weak attributes. The strengths and weaknesses form the basis to decide about improvements to be performed in phase 2.

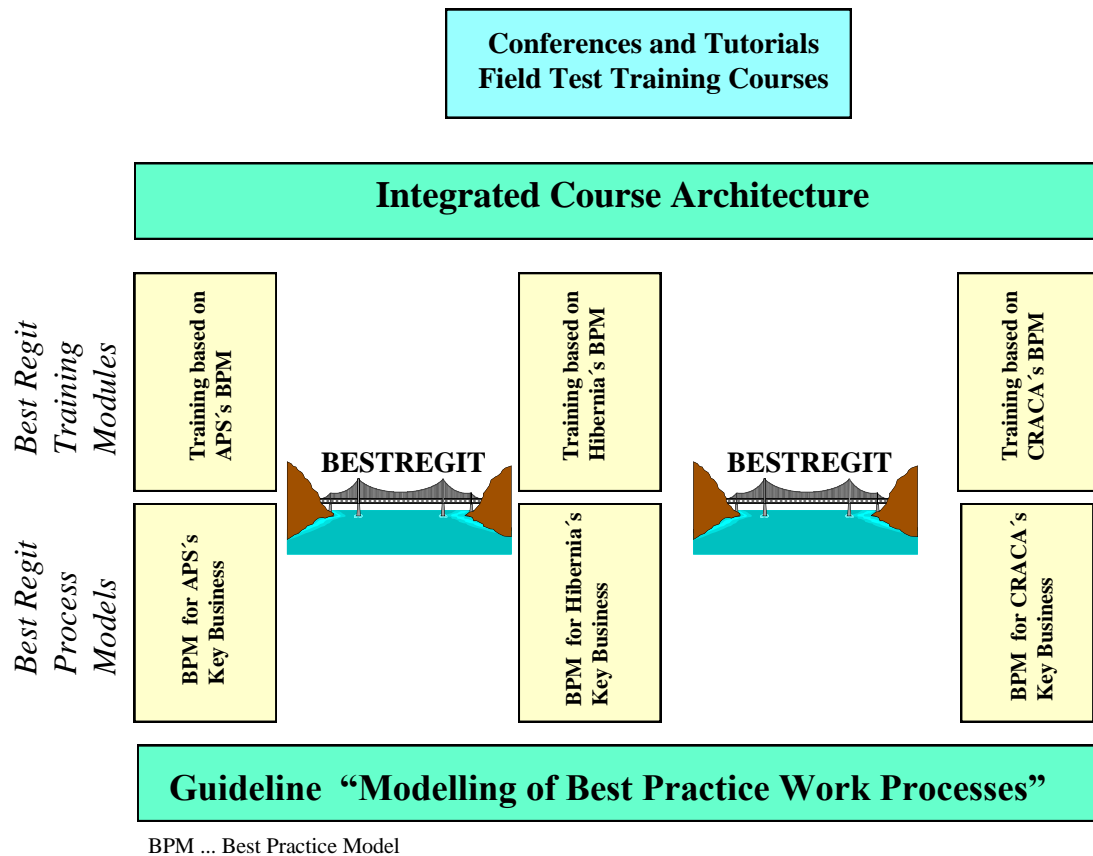


Figure 8: The Best-Regit Product Set

### Phase 2: Implementation of the Experiment

- use the questionnaire and benchmark results to draw a strengths and weaknesses profile for each partner based on the answers given to the checkpoints in the goal and teamwork analysis questionnaires.
- define a list of improvement actions to overcome the weaknesses, as well as to prioritise the actions. This shall include measurable goals: e.g. to reduce the average cost per student exchange by 25%.
- to actually plan and perform those improvement actions which have highest priority.

### Phase 3: Dissemination of Results and Lessons Learned

- summarise the steps performed and lessons learned. To compare the actual measures with the forecast values : e.g. the actual cost per student exchange only decreased by 18%.
- produce a tutorial based on the gained experience and feeding the material of each regional partner into an integrated workshop as "Best Practices Workshop" for regional innovation management organisations. The workshop does not focus on one model, it focuses on the set of steps to be performed to improve the method of work of a regional innovation management organisation.

The BESTREGIT logo is a bridge (see Figure 8) because BESTREGIT- is developing, modelling, and improving the key processes of distributed regional innovation transfer centres enabling them to set up a bridge for experience exchange and co-ordination and integration of joint activities.

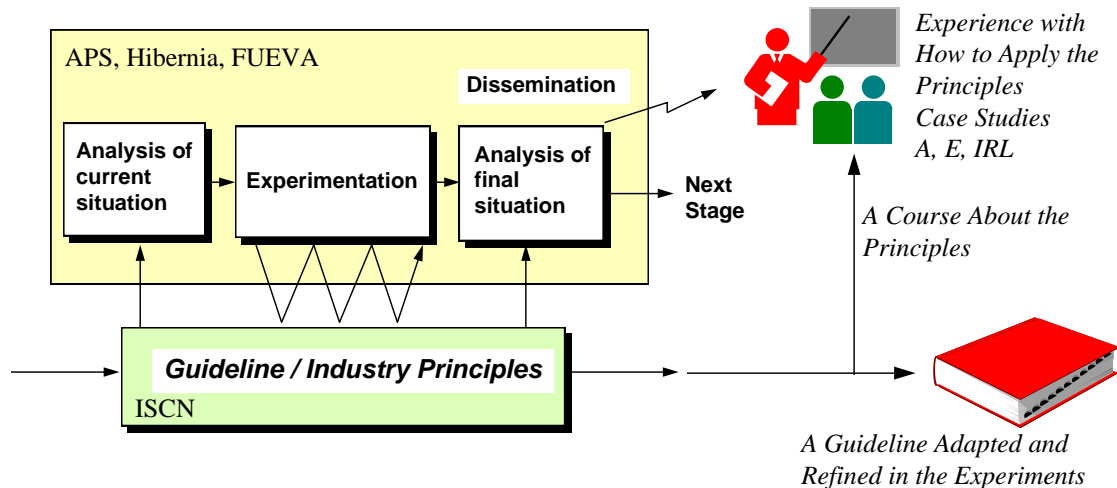


Figure 9 : Analysis (Modelling) - Experiment (Implementation) - Dissemination

## The Best Practice Experiment Guideline (BPEG)

The BPEG is a key tool in the project. It describes a set of steps for regional innovation management organisations, to

- analyse goals and teamwork capabilities
- benchmark against a set of attributes defined in goal and teamwork analysis questionnaires
- identify a set of potential improvement actions and to prioritise them
- actually plan and perform the improvement

The guideline will first start with a 100% industry approach. However, it is expected that the guideline will continuously be adapted and refined based on the feedback from the experiments (see interaction between guideline and experimentation in Figure 9), because firstly not all principles from industry are applicable to innovation management organisations and secondly there might be success principles in innovation management not used so far in industry standards. Thus the communication between ISCN and the experiments will lead to a continuous evolution of the best practice guideline which finally shall lead to a “Best Practice Industry Guideline Adapted to Innovation Management Organisations”.

For the below described 11 steps the guideline contains a discussion of the principles applied in industry, industry examples, and a recommendation about how to adapt and apply these principles in innovation management organisations.

1. INSTALLATION OF A PROCESS IMPROVEMENT MANAGER (TEAM)
2. IDENTIFICATION OF THE MISSION, BUSINESS GOALS, AND SUBGOALS
3. DEFINITION OF A SET OF MEASURES TO BE COLLECTED AND ANALYSED
4. IDENTIFY THE ROLES AND THE COMMUNICATION BETWEEN THE ROLES
5. IDENTIFY THE WORKSTEPS AND CREATE A DEFINED WORKFLOW
6. IDENTIFY THE RESULTS PRODUCED BY THE WORKSTEPS
7. ASSIGN YOUR PERSONNEL TO THE ROLES AND INACT PROCESSES
8. PRESENT AND DISCUSS THE MODEL WITH ALL PARTNERS AND EMPLOYEES AND ACHIEVE CRITICAL MASS
9. DEFINE A SET OF DATA TO BE COLLECTED
10. DEFINE A SET OF MEASURES TO EVALUATE THE DATA
11. ESTABLISH IMPROVEMENT ACTIONS BASED ON LESSONS LEARNED

These steps are well aligned with the goal oriented paradigms and the process maturity model. However, they are interpreted for innovation management in general and not for software development, which means that we (as mentioned above) focus on the learning strategy.

- as soon as the mission and goals of the regional transfer organisations have been structured it becomes visible which are the key areas in innovation management, and we can compare the Irish, Spanish, and Austrian model and learn about regional transfer organisations' goals.
- as soon as the teamwork, roles, and workflows are analysed and established for the key areas it becomes possible to discuss improvements, to compare the work practices with other regional institutions, and to jointly find a best practice way in a learning and cooperation approach
- as soon as data are collected it becomes quantitatively visible in which areas which impact on innovation transfer in the region has been achieved, allowing to make proper decisions for the future. And it becomes possible to compare data from Spain, Ireland, and Austria on a regional level.
- finally we plan to install the guideline as a learning tool inviting other regional institutions to join the learning and process improvement society.

Software people have very technical view points and mostly think about technological solutions, but they tend to forget economic, social, people, etc. factors which have major impact on the productivity of people and the efficiency of organisations. Innovation management organisations have to deal with a vast set of issues related with innovation transfer which reaches from human resource transfer and mobility actions (placing an expert into a company for pushing innovation), workshops about new ideas in a certain branch, etc., and technical things such as dissemination of information about new technologies available.

## **Conclusions after having started with the learning concept**

This paper has tried to give a survey about a learning concept for innovation management organisations. In fact many SMEs need them to make the first necessary steps in the innovation process, but very often they are disappointed because they do not get the service they expect. There are many explanations for this, including completely wrong expectations from the side of the companies, we are trying however to spot some of the mistakes within the service organisations.

- From their public oriented structure many innovation management organisations try to do everything - all singing and dancing is the opposite of being able to put our priorities right
- To measure things you need a convincing cost benefit analysis not only for your main business fields but for every tiny project
- The most ambitious staff shows reluctance because they think the time they invest in improving their work time will lack for clients. Especially part time staff is affected by time -consuming team workshops - at the other hand full information and involvement is especially important for them as they are not here all the time
- Some of the staff can see no direct link with their work

So far it seems that many of the things that have been said about Small Software Development Organisations (SSDs) are true for small Innovation Management Organisations but we assume that differences will turn up as the project is moving on.

There is much talk about excellent companies of the 21st century, which are called Companies of the Future. The most important resources of a COF is knowledge and wisdom.

A COF should have its own core competence both technological competence and managerial competence. To achieve the last one a close collaboration between top and middle managers is required.

Our ultimate goal must be to create IMOFs:- Innovation Management Organisations of the Future parallel with the appearance of COFs - Companies of the Future. As these kind of organisations will be working with their clients in virtual companies very soon and as the supply of knowledge increases continuously they only can apply and transfer know-how effectively if they organise themselves around strategic priorities, business opportunities and key competencies. Furthermore they will have to be able to access and use the most modern information technologies.

We are very optimistic that BESTREGIT will help us to become an IMOF by finding a well balanced interaction between work and learning processes. We would like to invite you to join this exciting process by becoming a client or a partner.

## References

- [1] EC, DGXXII, White Paper on Education and Training. Teaching and Learning. Towards the Learning Society, 1996
- [2] EC, DGXIII, First Action Plan for Innovation in Europe, 1996
- [3] Briand L., Differding C., Rombach D.: Practical Guidelines for Measurement-based Process Improvement, Technical Report of the International Software Engineering Network, ISERN-96-05, 1996
- [4] Debou C., M. Kuntzmann-Combelles A: From Business Goals to Improvement Planning,.: Practical Use of ami® In: Proceedings of the SPI96/ISCN96 conference, Brighton, December 1996
- [5] Haase V., Messnarz R., Koch G., Kugler H.J., Decrinis P., BOOTSTRAP: Fine Tuning Process Assessment, IEEE Software, pp. 25-35, July 1994
- [6] Messnarz R., Kuvaja P., A Modern Software Process Assessment Methodology, in: Proceedings of the 5th European Conference on Software Quality, September 1996, Dublin, Ireland
- [7] Messnarz R., Kuvaja P., Experience with Process Improvement Planning, in: Proceedings of the ISCN'96/SP'96 Conference in Brighton on 2-6 December 1996, December 1996, ISCN Ltd.

# PICO - A Training Concept for Creating a Learning Culture and Initiative in IT Organisations

Dr Richard Messnarz  
editor of joint article  
*ISCN Ltd., Dublin, Ireland*

The PICO Team  
<http://www.iscn.ie/projects/pico/>

## **Introduction**

PICO's Mission is to address process improvement from analysis to success. To achieve this, PICO has developed a comprehensive set of configurable training courses packaged with a book as a basic information pool and with a tool for supporting automated generation of analysis data.

PICO is based on the "learning by doing" training principle and on the paradigm of a learning organisation. Only those systems who are able to continuously adapt themselves to new situations and environments have a chance to survive. Learning and dealing with these new situations is a key success factor. Life that stops learning stops living.

This article outlines the different PICO components and products, the learning strategy, how the concept of a self learning organisation is supported, and will give organisations a detailed insight into PICO as a collaborative and learning based project.

*The project is carried out with the financial support of the CEC under the Leonardo da Vinci Programme [11].*

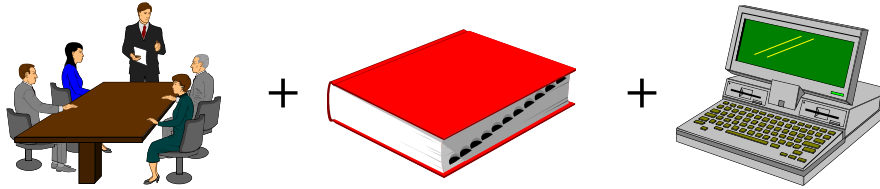
A key advantage of PICO is that it is based on three components which are complementary. Either someone buys the book and orders later the training (if needed) and downloads the public domain tool. Or someone downloads the public domain tool, tries it out and orders the book to get access to industry cases about how to apply the tool's results in industrial environments. Or someone attends a PICO training course and receives the book and tool as supporting material.

So PICO experience can be accessed from three different directions, and each component is self consistent or can be combined and supplemented through the other two components.

## PICO's General Concept

PICO's Mission is to address

Process Improvement from Analysis to Success



This achieved by developing a comprehensive set of configurable training courses packaged with a book as basic information pool and with a tool for supporting automated generation of analysis data

**Figure 10 :** PICO's General Concept

The three formulas

- a) Training = Course + Book + Tool
- b) Public Domain Tool → Ordering Book + (eventually course if not applied by oneself)
- c) Book → Downloading Tool + (eventually course if not applied by oneself)

are used to address IT people from three different directions

1. Personal contacts and consulting leading to training courses and the execution of above formula a.)
2. Wide and public audience with interest in employing analysis methodologies in their own organisations leading to a download of the PICO tool set and the execution of above formula b.)
3. Wide audience addressed by the publisher with interest in generally enhancing their background information and to know more about principles and real case industry examples leading to the acquisition of the book and the execution of above formula c.)

All formulas might lead to the acquisition of all three complementary components (Figure 10). However, each component is self consistent and can be used without the other two.

Another key approach of PICO is that the training modules cover different target groups within an organisation, and they specifically attempt to bridge the gap of understanding between practitioners and technical managers, and between technical managers and business managers. By building the bridge a joint understanding of the need for process improvement is achieved.

### **Definition of Software Process Improvement**

SPI (Software Process Improvement) requires system thinking, all issues like *business, processes, people and learning factors, cultural aspects, management and measurement paradigms, and infrastructure environments* must be integrated into a consistent framework. SPI is everyone's job, and any system (also the term SPI) underlies a continuous evolution and needs therefore continuous improvement.

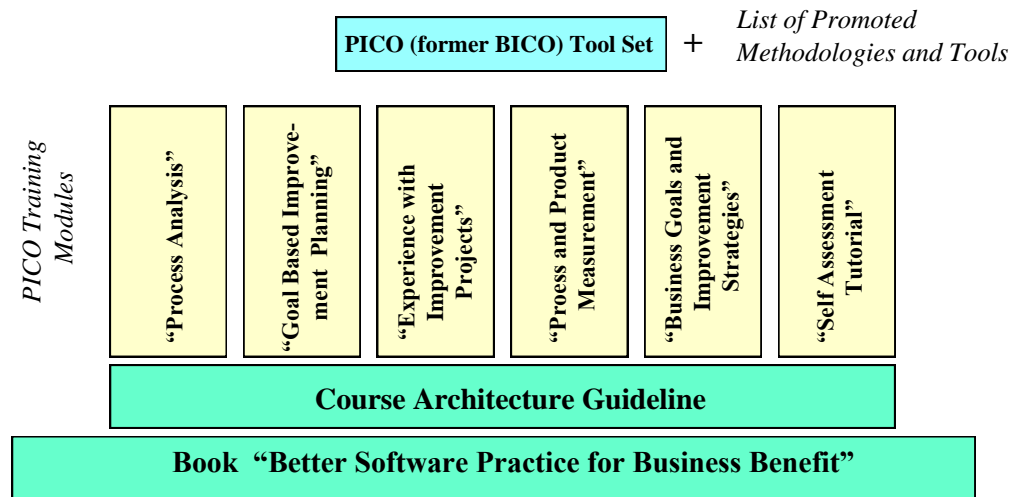
SPI is the philosophy which provides organizations with the necessary understanding and motivation to build up frameworks that efficiently integrate business goals, work processes, people, infrastructure and equipment, customer demands, and measurement paradigms into a consistent system to produce



faster at lower cost with sufficient quality satisfying the customer requirements with a realistic break even point to achieve *Return on Investment*.

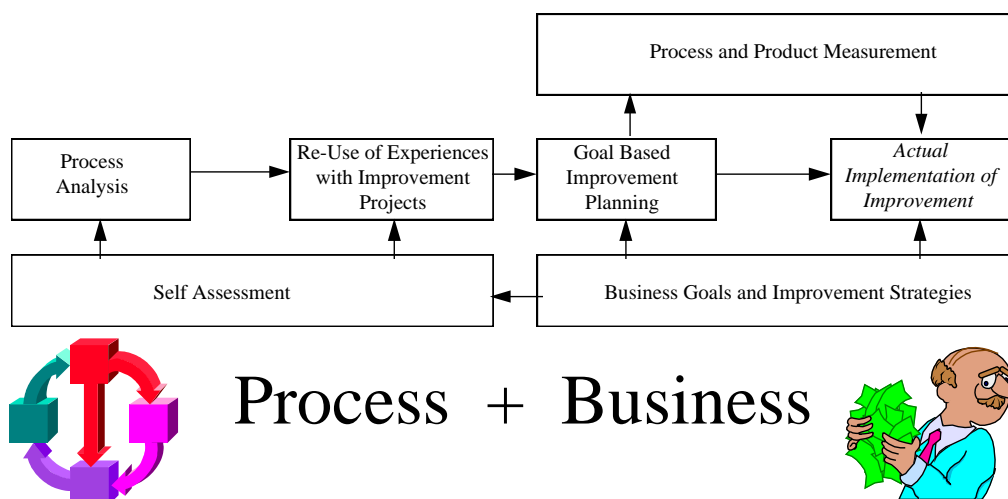
**There are 6 training modules**

- Process Analysis
- Goal Based Improvement Planning
- Experience with Improvement Projects
- Process and Product Measurement
- Business Goals and Improvement Strategies
- Self Assessment Tutorials



**Figure 11: The PICO Product Set**

Figure 12 shows how the training modules in Figure 11 relate to each other forming a road map for the overall PICO training course set. Of course, each training module is self consistent, but again it can be combined with the other modules into a coherent course with a minimum of 1/2 day training up to a maximum of 1 week (5 days) training course.



**Figure 12 : The PICO Road Map for Training Modules**

Depending on the previous skills and experience of the audience there might be different modules used as entry points. However, Figure 12 assumes that (basing on the idea of addressing different

levels of people in an organisation) there are two basic entry points for newcomers in the fields of process improvement.

After realising a software related problem (delivery times, product quality, rework, etc.) a technical manager might look to SPI for a solution and get interested in how to assess the current situation. For evaluating the efficiency of the work processes a procedure for assessment and benchmarking could be applied, resulting in a self assessment performance. The PICO course module about *Process Analysis* addresses this issue by providing a workshop for guided self assessments, also providing the PICO tool, as well as an assessment result interpretation as a second part of the workshop after the attendees have done the self assessments as homework.

If the technical manager then realises that the self assessment was not sufficiently analysing the entire organisation the PICO training module about *Process Analysis* also contains an introduction to a set of recommended large scale assessment approaches and methodologies available on the market. This way PICO aims to act as a kick off for improvement initiatives and as a window to a set of available improvement methodologies.

After the organisation has run through a self assessment or a full scale assessment (based on the PICO *Process Analysis* training) the technical manager might attend a course about *Experience with Improvement Projects* to see from real case industrial examples how to proceed with action planning and improvement implementation after the analysis.

A business manager usually firstly gets interested in how to achieve ROI by investment into the improvement of processes, infrastructure and people's skills. They do not have time to listen too much or to read large sets of material, they want to see in short time from business case examples why and how it could work, which decisions to be made, and which approach in general to be followed.

The PICO course module about *Business Goals and Improvement Strategies* consists of two half day workshops: *Business Motivation for SPI*, *Management Decision Support for SPI*. The workshop about *Business Motivation for SPI* discusses 5 key process improvement issues and their business context, includes a focused review session in which all participants exchange and discuss their experience with respect to the 5 key issues, and provides background material about how other business managers dealt with the 5 key issues.

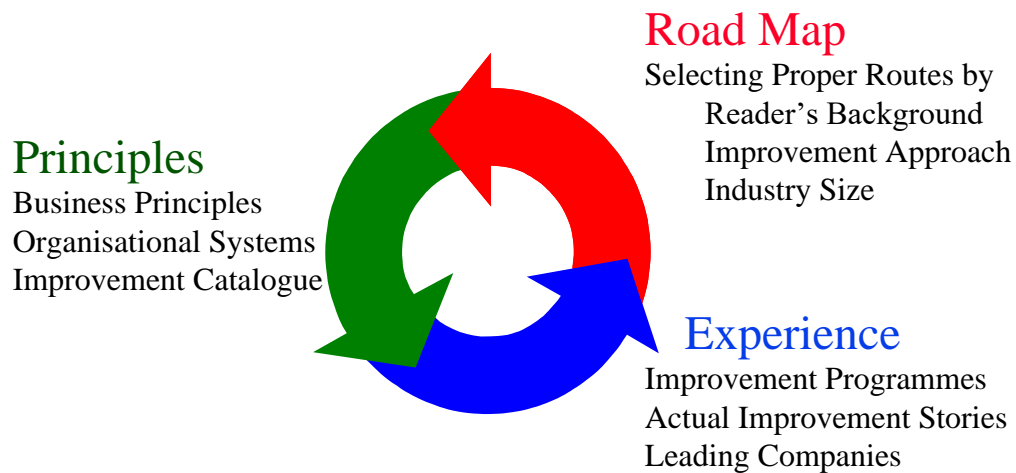
*Management Decision Support for SPI* illustrates the different approaches to start an improvement programme, provides a support for selecting a certain methodology, and illustrates which decisions to be taken in general and how to prioritise improvement actions once they have been proposed by the technical manager.

A key element in the PICO course set is the module *about Goal Based Improvement Planning* because it functions as the translator of the business manager's strategic goals into a set of technical goals for the technical manager. This situation usually occurs (Figure 12) after the business manager has decided to invest into process improvement and has defined a set of strategic goals, and the technical manager has performed the process analysis and proposed a set of improvement actions. It is now important to align the proposed process improvement actions with the strategic business goals using an approach for *Goal Based Improvement Planning*.

The course module about Process and Product Measurement then teaches which measures to collect and analyse to be able to evaluate if the business and technical goals have been achieved, and to provide a quantitative objective basis for deciding about good experience to be re-used and bad experience to be avoided in future.

## The PICO Book

The PICO Book "Better Software Practice for Business Benefit" is structured in three general parts, with Part I Principles, Part II Experience, and Part III Synergy and Conclusions. Part I describes a set of business strategies and improvement methodologies, whereas Part II illustrates with industrial case studies from different companies how these methodologies were implemented and used, and Part III provides a road map for readers, summarises the lessons learned, and gives a future outlook.



**Figure 13 : Book = Principles + Experience + Road Map**

Part II Experience is like a show case for the methodologies presented in Part I Principles. However, there are three types of dependencies and links.

*Principles and Experience*

- linking principles with experience to have illustrations

*Principles and Principles*

- Establishing a framework into which the methodologies fit and avoiding redundancies and contradictions

*Experience and Experience*

- Any variety is allowed. With the same starting situation sets of different improvement methodologies were used in the industrial cases.

Chapter 1 of the book discusses process improvement from the business manager's viewpoint speaking about return on investment, fixed cost, variable cost, break even point, market share, and leveraging.

Chapter 2 is about process models, process evolution, process analysis paradigms and defines the basic technical terms used when starting a process improvement initiative.

Chapter 3 outlines the currently available methodologies for process analysis and assessment establishing a catalogue of improvement models.

Chapter 4 deals with benchmark models allowing to compare organisations on the market concerning software performance, productivity, and business performance.

Chapter 5 deals with the GQM (Goal Question Metric) Paradigm and illustrates the ami approach as a framework for goal based improvement planning and measurement.

Chapter 6 focuses on a basic set process and product metrics which are used in measuring productivity, size & complexity and quality.

Chapter 7 illustrates cost / benefit examples from process improvement studies.

Chapter 8 presents Siemens's assessment and improvement programme, and chapters 9 and 10 illustrate the experience with improvement projects starting with the Siemens assessment approach.

Chapter 12 discusses Alcatel's experience in the same application domain as presented in chapters 9 and 10 by Italtel.

Chapter 13 deals with experience with the ISO 12207 process modelling standard for which a well defined reference to the architecture of the SPICE (Software Process Improvement and Capability dEtermination) methodology is available, and chapter 16 presents the experience with the SPICE trials.

Chapters 14, 15, and 17 represent industrial case studies from SMEs (Small and Middle sized

Enterprises) and VSMEs (Very Small Enterprises) covering process analysis, improvement planning, measurement, and benefit analysis.

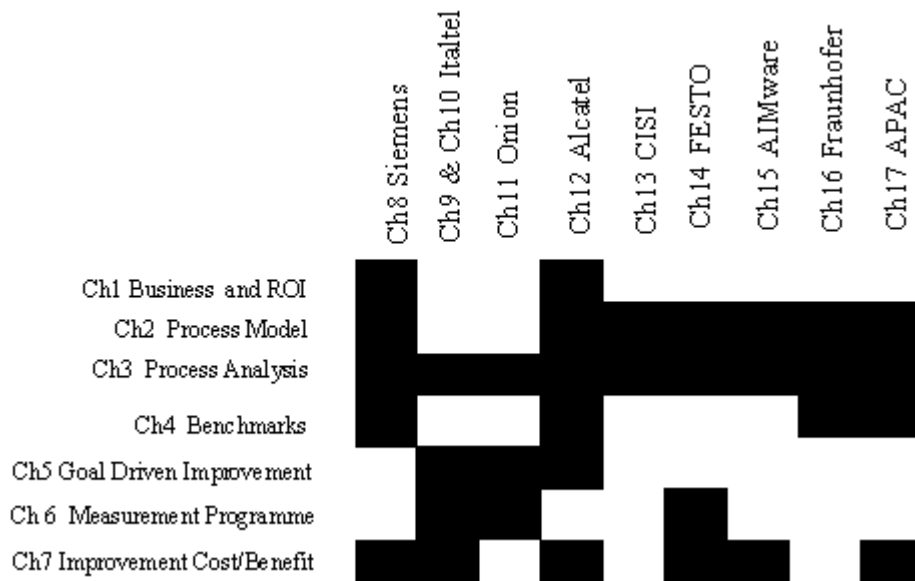


Figure 14 : Overview of Relationships Between Principles and Experience

Chapter 18 deals with a classification of the experience presented in this book and provides a guideline for readers to establish their road map for reading the book and for extracting the experience which is most applicable and beneficial for them.

Chapter 19 concludes that we must change the technical movement into a business movement so that top executives start to invest into process and product quality aiming at Return on Investment.

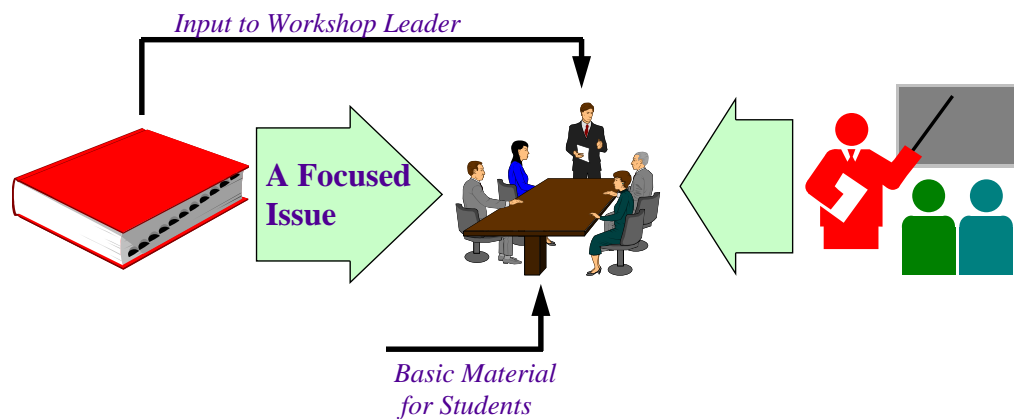


Figure 15 : The Book's Role in PICO

## The PICO Training Portfolio

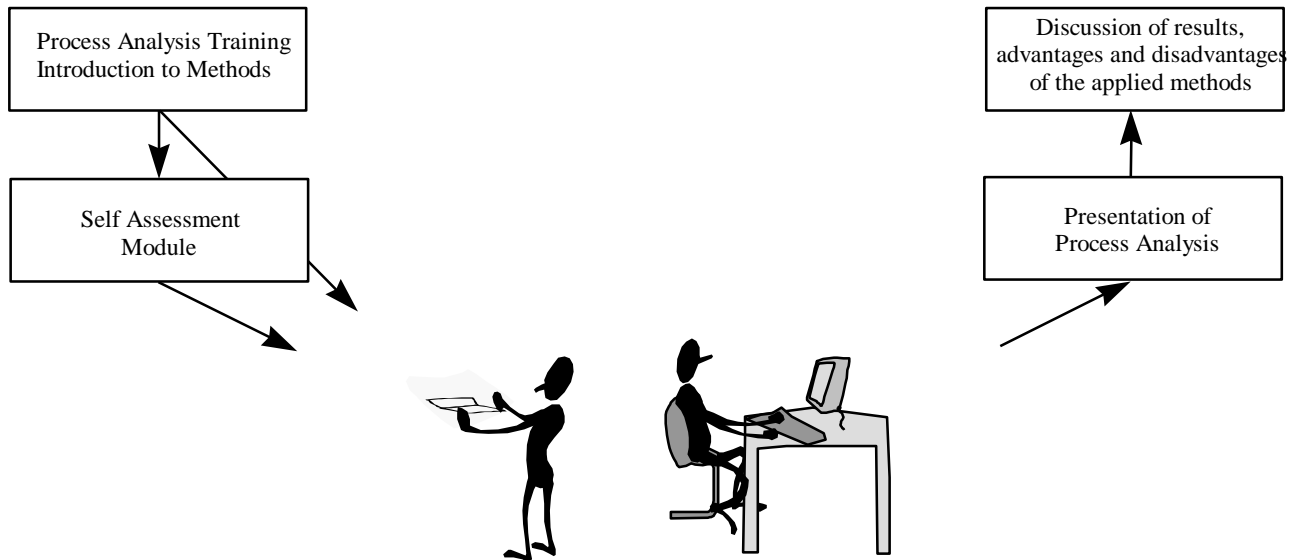
### Process Analysis

Target Audience

The main target audience are Quality Managers, Software Engineering Process Group Members Consultants and those responsible for the implementation of process analysis and process improvement.

### Training Approach

The workshop consists of two half day modules.



**Figure 16 : The Process Analysis Workshop Approach**

After the first session in which the different approaches are presented and discussed each workshop attendee will be assigned, according to a selected approach, an analysis task involving some processes within their own organisation.. After one week at the second session each attendee should briefly present the results and discuss the methods used to assess the processes in their company. This approach was selected according to the "Learning by Doing" principle.

It is recommended that the attendee of this workshop attends in parallel, a half day training on the MM Tool which should be organised in the afternoon after the first session to allow them to use the tool for their "homework".

### Contents Overview

#### First Session:

- I. Introduction and Definition of the Content of the Workshop  
The Workshop starts with an introduction to PICO itself. During this introduction the structure of this Workshop, the objectives and an overview of the models presented will also be given.
- II. Introduction to Process Analysis
  - A. Objectives of Process Analysis
  - B. Required abilities for Process Analysis
  - C. A Well Defined Process of Process Analysis
- III. Introduction to pre-defined Models
  - A. SEI
  - B. SPICE
  - C. Trillium

- D. Bootstrap
- E. Crosby's maturity grid
- F. ISO9001 (TickIT), ISO900-3
- G. BICO
- H. Self Assessment - PICO
- I. Self Assessment - Synquest
- J. Self Assessment - EQA
- IV. Generic Approach
  - A. Process Capture
  - B. Process Documentation
  - C. Process Analysis
  - D. Process Improvement, Re-Design
- V. Comparison of the different Models

Second Session:

- I. Presentation of the results of the homework
- II. Discussion of the results
  - A. Lessons learned
  - B. List of the pitfalls (most probably problems)
  - C. Workshop Feedback

References:

PICO Book (Chapter 2, 3 and 4)

### ***Goal Based Improvement Planning***

Objectives:

This course addresses the critical phase of translating the results of a software development process analysis into adequate improvement actions. From experience, a successful improvement planning starts with the identification of the company strategic goals defined by business management and their translation into more technical goals for the software part.

During this one-day workshop, students will learn :

- How to structure a software process improvement program in a goal-oriented manner
- How to produce an improvement action plan
- How to select measurement as follow-up mechanism of the SPI program
- How to define contingency actions to overcome typical obstacles to the success of the program.

Target Audience:

The main target audiences are improvement program co-ordinators and Software Engineering process group members.

Training Approach:

The training is based on the "Learning by Doing" principles. Students will apply what they've learnt on their own case study: they will select from on-going/future work a given scenario

(e.g. organisation of an overall improvement program, production of action plan, set up of a measurement plan, set up of an improvement pilot project). This scenario will be used as starting point for the exercises. Hence students will try to solve their own real issues during the course and can start implementing solutions as soon as they come back at their office.

Contents Overview:

### 1. Introduction (30 Min)

Collect experiences of the group in Software Process Improvement and select case study.

### 2. Obstacles to SPI success (20 Min)

Here the main issues and the related causes for successfully implementing process improvement are highlighted.

### 3. Goal-Oriented Improvement Strategies for Action Planning (45 Min)

The main approaches for process improvement action planning (e.g. SPICE, ami, QIP, SEI IDEAL) are briefly highlighted. Emphasis is given on the ami approach which principles will be applied in the next part.

### 4. Establishing an Action Plan in a goal-oriented manner (3h)

This is the kernel of the course. Some pragmatic steps for goal-oriented action planning will be introduced. Exercises will focus on the schedule of action planning, definition of a goal-tree linking business goals, process improvement goals, improvement actions and follow-up metrics. Students will also investigate how to solve issues like commitment, ...

The detailed steps are:

- Importance of measurement in SPI
- Schedule for the action planning phase
- Analysis of assessment results
- Establishing software process improvement goals
- Check goals with constraints
- Exercise -1
- Define high level improvement actions
- Defines follow-up metrics
- Build action plan outline based on resulting goal-tree
- Obtain management buy-in
- Establish detailed action plan
- Exercise - 2

### 5. Case studies (1h)

A couple of case studies from industry in applying goal-oriented approach for process improvement are presented.

### 6. Conclusion/Wrap-up (30 Min)

References:

The essential reference to this workshop is the ami Handbook and articles on its industrial application. Ideas have been taken from The G/Q/M approach from NASA/SEL, the SPICE guide for process improvement and the SEI IDEAL model.

## ***Experience with Improvement Projects***

### Target Audience

Quality managers, Development managers, SPI program managers, SPI consultants and SPI team leaders

### Training Approach

Given that the learning objectives of this workshop are

- to provide attendees with a clear understanding of the different types of improvement project by sharing practical experience of these approaches
- to enable attendees to reuse the experiences presented based on an understanding of the principles involved and a *practical* knowledge of how to implement them
- to maximize reuse of successes and minimize repetition of failures in improvement projects
- to help attendees decide on the appropriate strategy for their own organizations

the training approach adopted in the workshop is one based on 'learning by doing'. The attendees are presented with the starting scenarios of two case studies in process improvement. Through exercises the attendees identify what they would have done in each situation. A description of the approaches actually implemented by the case study organisations, including results/findings from their experiences, is then presented. Attendees then discuss the application of lessons learnt to the situations within their own organisations. Built around the case studies there are presentations on the different improvement approaches, analysis methods and SPI critical success factors. The final exercise involves the attendees developing and discussing their improvement plans/strategies to address their organisational-specific goals and issues. The plans/strategies re-use the lessons learnt from the practical experiences as presented and discussed with the tutor.

### Contents Overview

- ⇒ Introductions
- ⇒ Exercise: to get attendees thinking about the problems they face in their own environment so that they will recognize any similarities in the case studies to be presented later
  - Attendees to consider and present the most significant problems they face in their organizations
  - Discussion on each attendees situation
- ⇒ Brief Overview of the different analysis methods
  - CMM, ISO9001/TickIT, Bootstrap, SPICE, Etc.
- ⇒ Overview of the different improvement approaches
  - process modeling based
  - training based
  - technology introduction based, etc.
- ⇒ Case Study A : Schaffner Ltd. - A company designing and manufacturing power supply automatic test equipment. Case study mainly concerned with a focused improvement in the area of peer reviews.
  - company background, business objectives/issues
  - problems faced, audit results against ISO9001
  - initial initiatives and results achieved
- ⇒ Exercise/Discussion to get attendees to consider how they would go about improving the situation



- which problems to address
  - prioritization of improvement actions
  - choosing an improvement approach
- ⇒ Description of what Schaffner actually did and the results achieved
- identified project management, configuration management, requirements management, testing and design peer reviews as the priorities
  - during the improvement choose to give peer reviews a significant focus
  - details of this focused improvement on peer reviews
    - why chosen
    - approach adopted
    - results achieved
    - lessons learnt
- ⇒ Exercise/discussion: on how this experience relates to the attendees situation
- 
- ⇒ Case Study B: QFS - A company which develops financial products for dealers. The case study is about the improvement initiative undertaken by QSF within the ALCAST project. This was an ESSI funded Process Improvement Experiment focusing on software testing in a groupware environment (ALCAST stands for Automated LifeCycle Approach to Software Testing).
- company background, business objectives/issues
  - problems faced, self-assessment results
- ⇒ Exercise/Discussion to get attendees to consider how they would go about improving the situation
- which problems to address
  - prioritization of improvement actions
  - choosing an improvement approach
  - adoption of improvement lifecycle
- ⇒ Description of what QFS actually did and the results achieved
- tailored the V-model/STEP testing process to fit into their groupware based development process
  - formalized test planning with input from marketing and customers
  - automated regression testing
  - results achieved
  - lessons learnt
- ⇒ Exercise/discussion: on how this experience relates to the attendees situation
- 
- ⇒ Critical success factors in planning and implementing an improvement project
- ⇒ Exercise: for each attendee to develop an outline action plan or strategy for their improvement project
- action plan should contain problem areas and their prioritization, improvement lifecycle and approach to be used, issues to highlight to ensure reuse of lessons learnt from this course, etc.
  - discussion of each attendees strategy
- ⇒ Summary and reference to additional case studies in book (WP24000)

## References

ALCAST final report  
 TRI-SPIN Schaffner Case study ([www.cse.dcu.ie](http://www.cse.dcu.ie))  
 PICO book

## **Process and Product Measurement**

This module provides knowledge and understanding of what is involved in introducing and performing measurement in order to improve software processes and products. It explains the types of process and

product measurement that can be made during software projects, and includes a Case Study which illustrates some of the practical issues arising from a software measurement programme. Throughout, it is based on the “Goals-Questions-Measurement” approach to linking measurement with business goals.

There are three underlying principles to the module:

1. improvement must be based on measurement, and therefore real quantitative information
2. measurement must be directly related to business goals and performance or improvement targets (not done just for its own sake)
3. measurement must be based on existing information and practices

## Target Audience

The principal target audience for the module is software development managers, project managers, team leaders and software engineers who will be either:

- directly involved in developing software products and running projects (and therefore performing measurement)
- responsible for product performance and project achievement (and therefore making use of the measurements)

It will also be useful for

- senior managers who need to understand how a software measurement programme can provide value and benefits, and in particular in a way that is linked to their business goals
- quality or technical staff involved in implementing a measurement programme or improvement initiative.

## Training Approach

The training approach is based on a series of instructional sessions presented in an interactive way, assisted by Case Study sessions aimed at reinforcing the information provided by the sessions and illustrating the practical issues involved in running a software measurement programme.

A selection of relevant methods are described and illustrated. However the module is not dependent on specific measurement methods, tools or techniques and if preferred it can readily be adapted by an organisation to include reference to its own methods.

## Content Overview

The module covers five main topics, each presented as a single session:

1. **Measurement in the Software Engineering Process.** This topic provides an overview of what aspects of software processes and products are practical to measure and how this can be approached. It explains the reasons for making measurements and the advantages to be gained, particularly by relating the measurements performed to business goals and the associated need for process improvement. The session draws partly on the experience of recent European projects such as METKIT, PYRAMID, AMI and ESPITI.
2. **Specifying and Measuring Software.** Software which is specified fully and in a way that can be evaluated quantitatively is an important benefit of a software measurement programme. This session explains how this can be done, with examples of requirements and how they can be evaluated quantitatively as a consequence of a measurement programme.
3. **Cost Estimation of Software Development and Maintenance.** One of the important benefits of

measurement should be the ability to produce accurate and reliable estimates of important project parameters such as time, effort and cost and product parameters such as size and complexity. Effective estimating greatly assists both developers (suppliers) and purchasers of software in the procurement and implementation of new applications and software and to manage its subsequent maintenance. This session explains typical quantitatively-based estimating processes.

4. **Setting up an Improvement Programme.** A measurement programme requires careful planning and implementation if it is to be introduced effectively and result in worthwhile benefits. It requires the support of senior management and staff, linkage to business objectives and careful planning, implementation and support. This session explains how this can be achieved.
5. **Process Benchmarking and Optimisation.** One of the benefits of an implemented measurement programme should be the ability to compare achieved performance with other projects, other areas of the organisation or even external sources. Analysis of measured performance (e.g. for consistency) can also identify opportunities for process improvement and optimisation and support continuous improvement. This session also explains some of the techniques available to support analysis of process measurement.

Two workshop sessions are included, which present the experience of a software development company which introduced a software measurement programme. After explaining and analysing the way in which this was done, attendees provide their own solutions to particular scenarios which arose from the programme. Their solutions are compared with what actually happened. The first session illustrates linkage between a measurement programme and business objectives, while the second session illustrates the linkage with quality improvement.

#### References

METKIT: Metrics Education Toolkit (industrial package) 1994. (METKIT Consortium)  
 PYRAMID: Quantitative Management; get a grip on software! 1991 (PYRAMID Consortium)  
 AMI Handbook: A Quantitative approach to software management 1996 (Addison Wesley)

### ***Business Goals and Improvement Strategies***

“Business Goals and Improvement Strategies” is decomposed into 2 half day workshops

- *WS1 Business Motivations for SPI*
- *WS2 Management Decision Support for SPI*

#### Target Audience

The target audience are executives, business managers, and middle managers who are considering whether or not to invest in process improvement. It also addresses those who continuously are concerned with decision problems concerning process improvement.

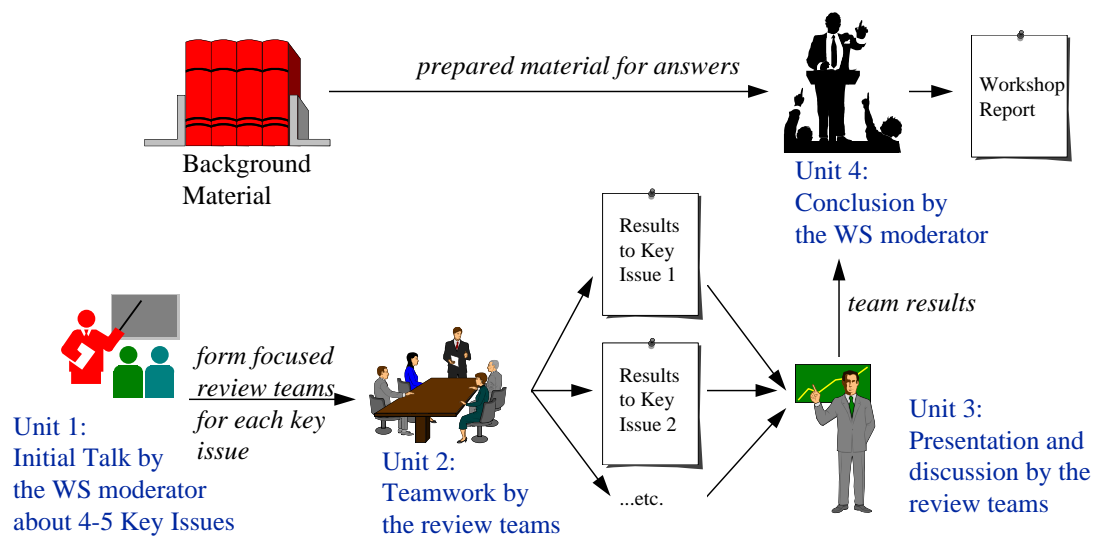
The workshops are highly interactive and thus they address those managers who are interested in an interactive teamwork-based workshop to discuss and summarize key business aspects and process improvement.

*WS1 Business Motivations for SPI* addresses managers with not much previous experience in SPI and interested in business focused SPI actions.

*WS2 Management Decision Support for SPI* addresses managers who decided to invest into SPI and who are interested in knowing how to start with SPI and which technical approaches are available

#### Training Approach

Each workshop starts with an initial talk by the workshop leader addressing key aspects and motivating the discussion of the different topics presented. In unit 2 the workshop participants are assigned to focused review teams working on the addressed topics. Each team receives one topic to work on. In unit 3 the teams present the results of their focused review and discuss them with the workshop leader. In unit 4 the workshop leader presents the prepared set of answers to the different topics and compares the prepared answers with the answers given by the teams. Finally all discussions and conclusions are summarized in a workshop report which is distributed to the workshop participants about 1 week after it.



**Figure 17 :Workshop Procedure - Workshop Units - General Learning Approach**

Each unit takes 50 minutes (see . The initial talk addresses 3 to 5 key topics motivating discussions concerning these topics. Due to the fact that in unit 2 the attendees are assigned to a focused review team per topic the minimum number of attendees is 10 to guarantee that each focused review team at least consists of 2 members.

The initial talk comprises a set of 15 to 20 transparencies. The background material bases on the PICO book, a set of 20-30 transparencies, and a number of selected articles and references.

For the workshop report a standard template is used which will ensure high quality and facilitate the integration of the teams' results with the prepared answers.

### Contents Overview

#### *WS1 Business Motivations for SPI*

The workshop starts with a 50 minutes initial talk about (Figure 17) Software Process Improvement addressing: SPI and Business Performance, SPI and SW Processes, SPI and Measurement, SPI and People Management, SPI and Infrastructure Issues, resulting in an up-to-date definition of SPI taking into account the different process improvement movements. The initial talk mainly bases on real business case examples and tries to highlight 5 major success principles.

#### *WS2 Management Decision Support for SPI*

The workshop starts with a 50 minutes initial talk about (Figure 17) selecting the proper Software Process Improvement methodology, managing the corporate decision process, and setting priorities and ensuring commitment. The initial talk mainly bases on real business case examples and tries to highlight 3 major success principles.

In the focused review session each attendee will work within a team on one topic, and each team will present and discuss their results with all other teams and the workshop leader. Ideally there will be a team for each of the topics which would, however, require a sufficient number of participants (at least 2 per topic).

Once a team has been established for a certain topic, the workshop leader provides it with the prepared focused review material. For each team a set of prepared focused review material slides is available. This material comprises pre-designed slides asking teams a number of questions and allowing them to present answers on slides following a standard format.

## References

- Chapter 1 of the PICO Book: „The Software Process in the Context of Business Goals and Performance“, in: Better Software Practice for Business Benefit 1997
- Chapter 2 of the PICO Book: „Software Process Analysis - Concepts and Definitions“, in: Better Software Practice for Business Benefit 1997
- Chapter 3 of the PICO Book: „Software Process Analysis and Improvement - Methods and Models, in: Better Software Practice for Business Benefit 1997
- Chapter 4 of the PICO Book: „Software Process Analysis - Frameworks and Benchmarks“, in: Better Software Practice for Business Benefit 1997
- Chapter 7 of the PICO Book: „Cost Effectiveness of Process Improvement Initiatives“, in: Better Software Practice for Business Benefit 1997
- Thamhain H.J., Wilemon D.L., Criteria for Controlling Projects According to Plan, in (eds.) Nahouraii E., Butler J.T., Petry F.E., Richter C., Tham K., *Software Engineering Project Management*, pp. 15-54, IEEE Computer Society Press, USA 1990
- Andersen W.L., *Technology Transfer is a Social Phenomenon*, in (eds.) Przybylinsky S., Fowler P.J., *Transferring Software Engineering Tool Technology*, pp. 56-57, IEEE Computer Society Press, USA 1987
- Dewayne E. Perry, Staudenmayer N.A., Votta L.G. *People, Organisations and Process Improvement*, pp. 36-45, IEEE Software, July 1994 1994
- Aaker, D.A. *Strategic Market Management*. John Wiley & Sons, Inc., 1995. 1995
- Goodhew, P. Achieving real improvements in performance from software process improvement initiatives. European Software Engineering Process Group Conference 1996 (C306). 1996
- Biró, M.; Feuer, É.; Remzsó, T.; Sz. Turchányi, P. Business Decision Problems Supported by Software Product and Process Assessment. In: Proceedings of the ESI-ISCN '95 Conference on Practical Improvement of Software Processes and Products (ed. by T. Katsoulakos, R. Messnarz). (European Software Institute-International Software Consulting Network, Vienna, Austria, 1995). 1995
- Messnarz R., Kugler H.J., BOOTSTRAP and ISO 9000: From the Software Process to Software Quality, in: *Proceedings of the APSEC '94 Conference*, Comput. Soc. Press of the IEEE, Tokyo, Japan 1994 1994

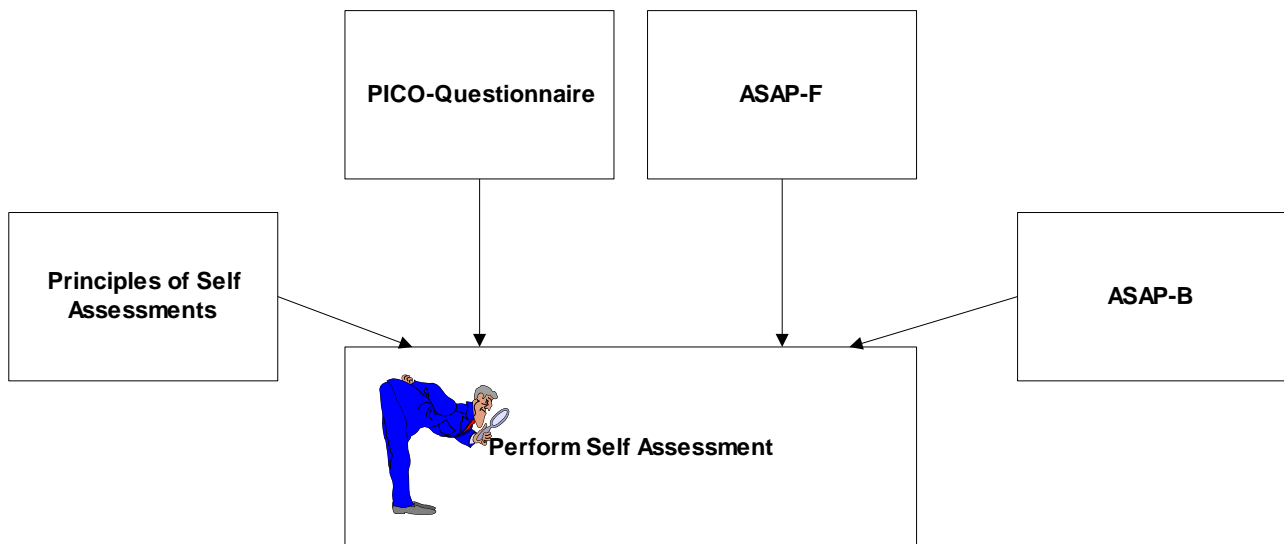
## Self Assessment Tutorial

Target Audience

The main target audience are Quality Managers, Software Engineering Process Group Members, Consultants and those responsible for the implementation of process analysis and process improvement.

### Training Approach

This workshop shall give the audience a guideline to perform a Self Assessment. Therefore, the methods, standards and tools will be presented. After a short introduction the basic principles of Self Assessment will be explained. The main part of the workshop deals with the PICO-Questionnaire. After this part the attendees should be able to continue on with the questionnaire and interpret the questions. The goal of the third part is to explain and demonstrate the workflow of a Self Assessment supported by the tools ASAP-F (Audit Support and Analysis - Frontend) and ASAP-B (Audit Support and Analysis Program - Backend). During this session the attendees will be also be taught to interpret the graphical output of ASAP-B.



**Figure 18 : Self Assessment Tutorial Approach**

The audience is encouraged to perform a Self Assessment during the course “Process Analysis”.

### Contents Overview

- I. Introduction and Definition of the Contents of the Workshop  
During this introduction the structure and objectives of the workshop will be given.
- II. Introduction to Self Assessment
  - A. Objectives of Self Assessment
  - B. Required Abilities for Self Assessment
  - C. Process of Self Assessment
- III. Introduction to PICO-Questionnaire
  - A. Capability Levels
  - B. Architecture of Questionnaire
  - C. Scoring
- IV. Introduction to ASAP-F, ASAP-B
  - A. Objectives of ASAP-B, ASAP-F

- B. Workflow ASAP-F
- C. Workflow ASAP-B
- D. Self Assessment with ASAP-F, ASAP-B

## References

PICO Book (Chapter 2, 3 and 4)

## The PICO Framework Tool

The PICO Tool has been designed to meet the following objectives:

- to be open for all questionnaires
- to be open for different algorithm
- what if - simulations

The PICO Tool consists of the two different programs ASAP-F (Audit Support and Analysis Program Front End) and ASAP-B (Back End).

ASAP-F is used for the collection of data. It runs under DOS and therefore can be used on any Palmtop which supports MS-DOS (e.g. HP). The tool accepts any questionnaire as long as it meets certain syntax requirements. Within the PICO-Project the tool will be used together with the self-assessment questionnaire.

The tools ASAP-F produces two files which contain the input information for ASAP-B.

ASAP-B runs under Windows95/NT. The evaluation result can be displayed in different ways and one of which is shown in the following picture. The tarts represent the scoring on every level of each attribute displayed. This evaluation method uses six levels as SPICE does. The questions can be answered by not applicable, absent, partially, largely and fully.

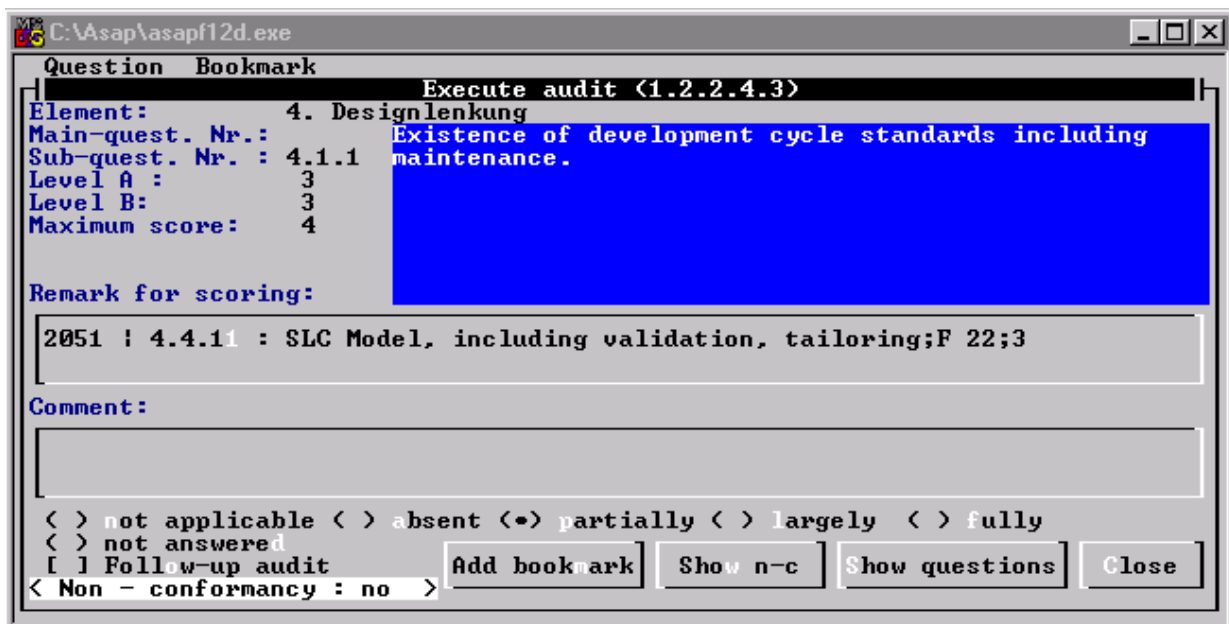


Figure 19 : The PICO Configurable Electronic Questionnaire

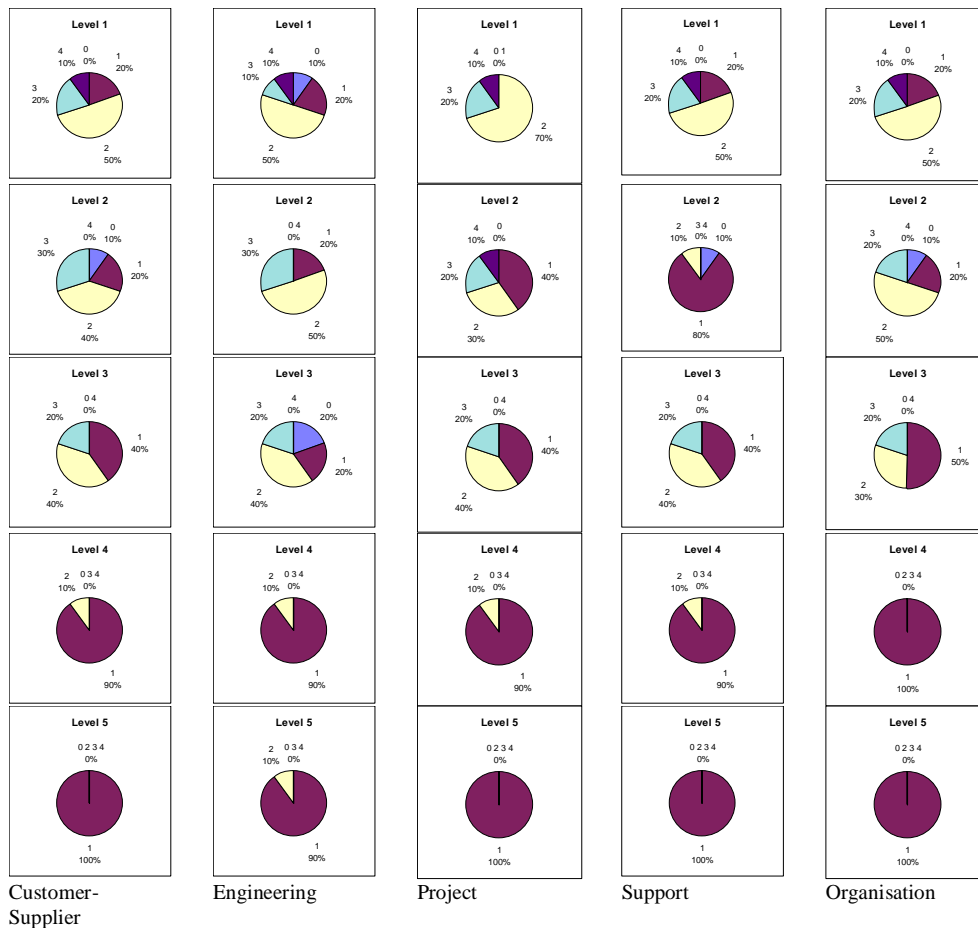


Figure 20 : Sample Results

## The PICO Exploitation Plans

During 1998 PICO is running into the exploitation phase in which

- courses will be performed in at least 10 different EU countries
- an electronic PICO newspaper on WWW will be established
- a discussion forum (also WWW supported) for the PICO trainers will be established
- a discussion forum (also WWW supported) for the PICO users will be established
- a professional brochure will be made available Europe wide
- a PICO user and trainer workshop will be organised at ESI & ISCN 1998 which is probably organised in Stockholm in Sweden, and from that on annually in European capitals.
- all partners will work on a joint agreement for further development and exploitation.

## The PICO Team

The PICO team consists of a kernel group of training course developers, in cooperation with leading companies who contributed to the book.

*The kernel and development team comprises:*

AIMware (Ireland), APAC (Austria, Prime), ami User Group (Belgium), APS (Austria), Brameur (UK), CISI (France), Q-Set and FOH (Ireland), Hibernia Learning Partnership (Ireland), ISCN (Technical Coordinator, Ireland)

*Book contributors were:*



Alcatel (France), CTA (UK), Festo (Austria), IESE (Germany), Leansoft (Finland), Onion (Italy), Siemens (Germany), Sztaki (Hungary)

Especially we acknowledge contributions to this article by Miklos Biro, Kevin Daily, Christophe Debou, Heinz Eckam, Fran O'Hara, Susanne Lanzerstorfer, Bernhard Posch , Hans Scherzer, Eric Trodd.

# PICO's Multi-Method Framework Tool-Set for Self- Assessment

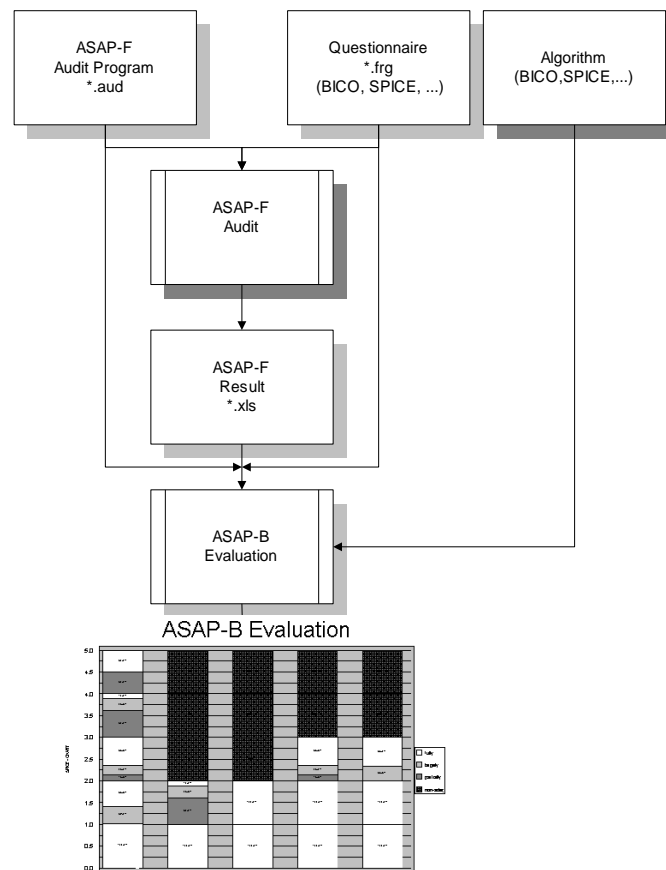
Susanne Lanzerstorfer, Hans Scherzer  
APAC, Vienna  
*apacshz@ping.at*

## I. Overview

The tools ASAP-F and ASAP-B are designed to plan, perform and evaluate assessments and to visualise the results. These assessments are performed in order to determine the capability level of one or more processes or the conformity with ISO 9001.

The assessment process is supported by ASAP-F. During the assessment the questions will be displayed by ASAP-F. For each question the evaluation results and comments, if appropriate, are entered into ASAP-F. These results are then stored in a result file which is the input file for ASAP-B. The information about the organisation assessed and the information about the assessment itself will also be entered into ASAP-F and processed by ASAP-B.

ASAP-B provides graphical representation of the characteristic profile of the processes of a company. Based on the input information provided by the result file of ASAP-F and the questionnaire, ASAP-B calculates the results according to the definitions in the questionnaires and user defined parameters.



**Figure I-1 Relation between ASAP-F and ASAP-B**

Most of the users will be related to the area of quality management and therefore, they usually want to determine the improvement potential. The majority of the users will not use the tools frequently but they are used to working with computers and for this reason the user-interface is designed for easy learning and use.

Some users of the tools will be experts and consultants who may use the tools frequently. Consequently, the tools also offer a user-interface with quick access features (e.g. shortcuts).

## II. Programme Description

### III. ASAP-F (Audit Support & Analysis Programme - Frontend)

Features:

- To support the data collection during the assessment ASAP-F can be operated in a DOS-Environment. It runs on a handheld pocket-computer such as HP 100 or HP 200
- For usability reasons the tool has nearly the same look and feel user interface as Windows 3.1 (for the navigation either "TAB" , Function Keys or "Alt" and Highlighted Character can be used)
- The user can easily navigate through the questionnaire
- The tool offers the possibility to perform a ISO 9001 conformance assessment
- The user can create non-conformance reports

This tool was developed for performing BICO-Audits (Benchmarking & ISO 9001 Combined). As a consequence the naming conventions of the labels are derived from ISO 9001 (e.g. assessment = audit

). Originally the questionnaire of the assessment was divided into 20 Elements of ISO 9001. Therefore, all items concerning “Process Categories” are labelled with “Element”.

The planning of the assessment begins with the collection of some data concerning the environment of the assessment. This information includes facts about the organisation assessed, number of employee, scope, etc. The user also can choose the desired questionnaire from a list.

The next step in planning the assessment is to create the assessment programme (Figure II-1).

**Edit audit-program (1.2.1)**

Date: 20.07.1997

AuditNr.: 1  
 Auditttype: PICO-Audit  
 Scope: SW-Engineering  
 Auditttitle: Audit 1997  
 Lead-Auditor: LAZ  
 Audit-team: [STE]

<New Date: F1>  
 <Edit date: F2>  
 <Delete Date: F3>

Audit-unit:

Number	Time	Contact person	Elements
1.0	8:00-10:00	Binder	1 2 3 4 5

<New auditor: F4><Delete auditor: F5>

<New unit: F6> <Edit unit: F7> <Delete unit: F8>

Nr       Division/Project  
 Time     Contact person  
 Place     Elements

Cancel    OK    Authorize

**Figure II-1: Creating an Assessment Programme**

This includes the determination of the

- date
- auditors (= assessor),
- time,
- place,
- organisational unit,
- contact person of the assessor and
- the set of processes to be assessed.

To fulfil the needs of the ISO 9001 the user can fill in information about the authorisation (compiled, checked and approved) of the assessment programme. After the assessment plan has been compiled, the user can start the assessment. Firstly, the actual instance of the assessment programme has to be chosen. After selecting the date and time, the assessment is started by offering a list of those process subsets fixed in the assessment programme. After choosing the entry point of the assessment in the list, the window “Execute Audit” opens.

Execute audit (1.2.2.4.3)	
Element:	1 . Customer-Supplier Relationship
Main-quest. Nr.:	Document the approach to performing the process
Sub-quest. Nr. :	category "customer-supplier" in standards and/or
Level A :	procedures.
Level B:	
Maximum score:	4
Remark for scoring:	
Tasks to be performed; inputs and outputs ; entry / exit criteria ; control points ; internal and external interfaces ; process measurements	
Comment:	
<input type="checkbox"/> not applicable <input type="checkbox"/> absent <input type="checkbox"/> partially <input type="checkbox"/> largely <input type="checkbox"/> fully <input checked="" type="checkbox"/> not answered <input type="checkbox"/> Follow-up audit	
<input type="button" value="Add bookmark"/> <input type="button" value="Show n-c"/> <input type="button" value="Show questions"/> <input type="button" value="Close"/>	
< Non - conformance : no >	

**Figure II-2: Performance Assessment (example of an PICO-Assessment)**

To determine and document the degree of fulfilment of each question the following aids are offered:

- For every question the Question Number, the Process Category, the Level and a Remark for Scoring is visible. The Remark for Scoring should help the assessor to determine the scoring.
- The scoring can be easily selected by choosing the desired option box.
- Both non-conformance's and comments can be documented in two separate text boxes.
- The programme also offers the possibility to mark questions for a follow-up assessment. (This possibility is used for ISO 9001 assessments).

To help the assessor to navigate through the questionnaire a wide variety of features are implemented. Some of the essential features are:

- The user is able to jump to all desired questions. On the one hand, shortcuts to go to the next, previous, first or last question can be used. On the other hand, the user can jump to a specific question, whereby, the question number must be known.
- If the question number is not known, the programme offers the possibility of selecting the question from a list, showing the first 80 characters of each question.
- Another feature is the possibility of creating and using bookmarks within the questionnaire. The list of bookmarks contains both the question number and the first part of the question text.

To ensure that no data is lost in the case of a failure (e.g.: Notebook gets an empty battery), the user is able to make backups of the database during the assessment.

After an assessment ASAP-F provides two types of output:

Interface-files:

To provide all information for ASAP-B the following output-files are generated by ASAP-F:

- an XLS-file with all scoring and comments for ASAP-B and
- an AUD-file which contains the Assessment Programme and general information about the assessment

Reports:

The following reports are offered (both to printer and to file):

- the list of non-conformance's and
- the list of comments

#### **IV. ASAP-B (Audit Support & Analysis Programme - Backend)**

Features:

The tool ASAP-B

- visualises the results of an assessment,
- analyses the results of an assessment,
- simulates the results of an assessment,
- provides data exchange with other programmes and
- is designed for Win 95 and Win NT.

The first step in performing an evaluation is to initialise the ASAP-B database with the XLS- and AUD-interface file from ASAP-F. Then, the user has to assign a clustertable to the questionnaire (in the clustertables the questions are grouped). After choosing the evaluation-algorithm the user can start the evaluation. As a consequence, the tool calculates the profile according to the definitions in the questionnaires and parameters defined by the user (these parameters will be mentioned below) and the graphical representation of the result.

Besides this "normal" evaluation this tool provides many features to help the assessor analysing and simulating assessment results:

#### **V. Analysis and Visualisation**

- The tool visualises the results and the information about the assessed organisation and the information about the assessment itself.
- ASAP-B offers enhanced printing and visualisation capability (Header, Footer,...).
- The user is able to determine the improvement potential for each attribute.
- The user is able to compare the results of different assessments.
- The user is able to take and store notes concerning the input data.
- The user is able to define and alter cluster tables.
- The user will be able to access comments, to enter comments and to display the questions relevant for the scoring of an attribute.
- The user can easily change the dependencies of questions, their assignment to clusters and the scoring for simulation.

# ASAP-B Evaluation

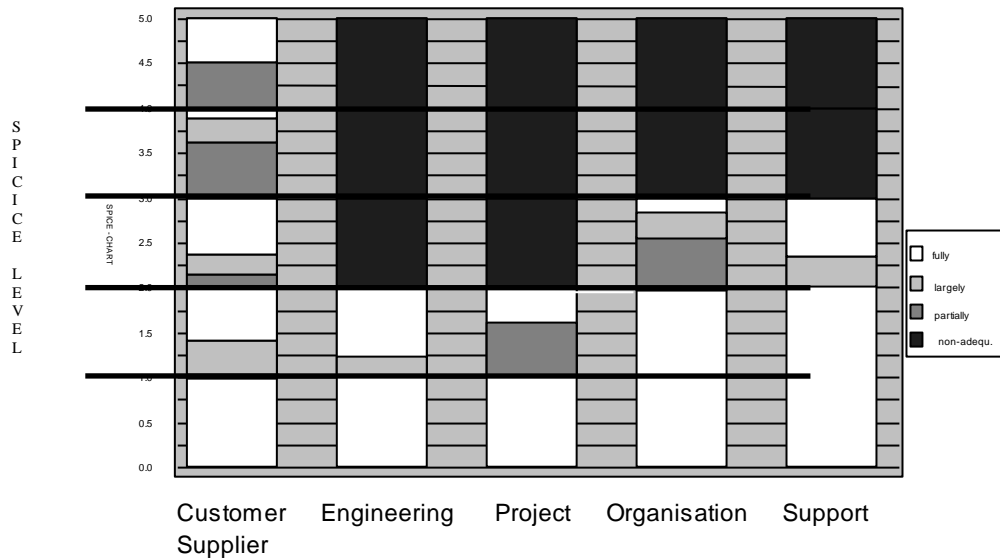
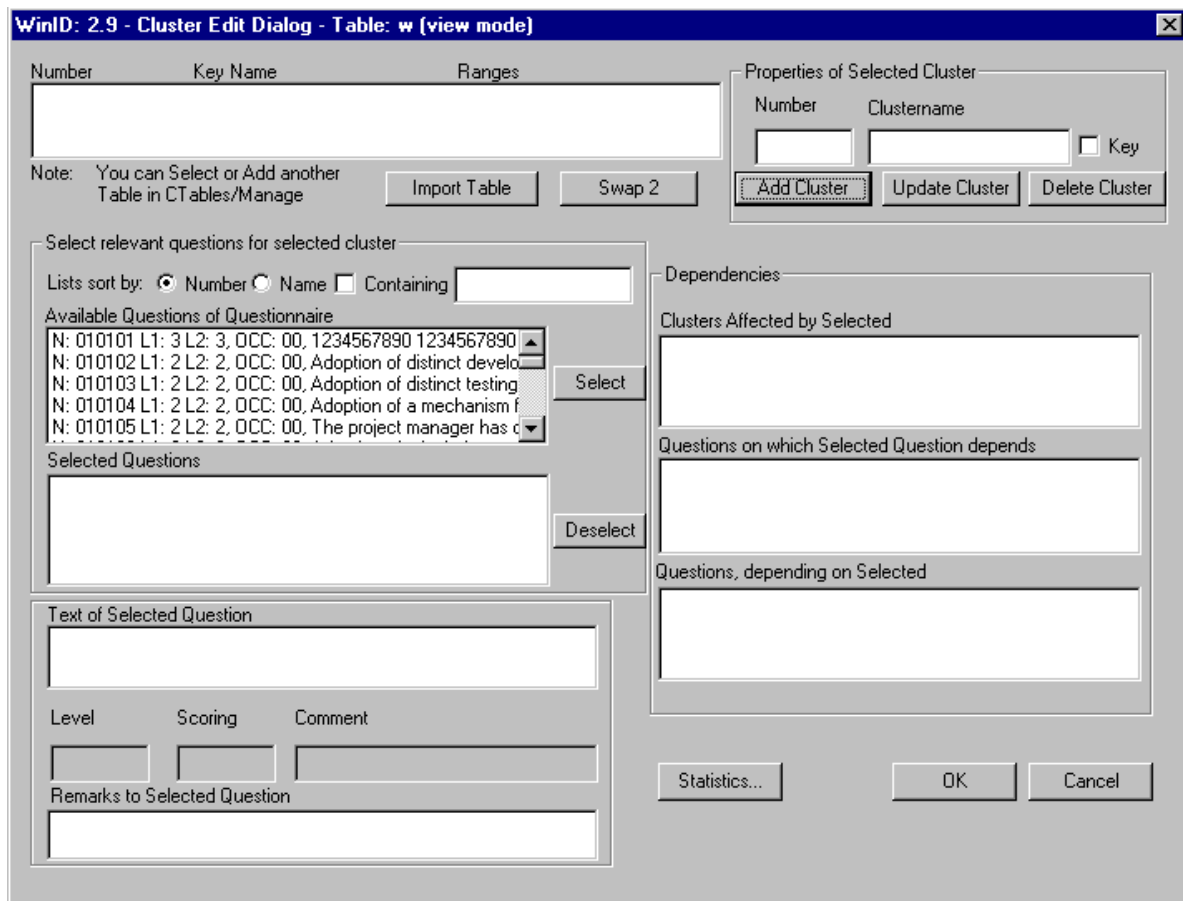


Figure II-3: Visualised Result of a PICO-Assessment

## VI. Simulation

- The user is able to clone the active archive into a new window (for simulation-purpose), to alter parameters and to evaluate the clone with the changed data to compare this simulation with the unaltered archive.
- Simulation data will be stored without modification of the original input data and these changed values will be displayed in a different colour.
- The user has the option to select either the original input data set or a simulation data set.



**Figure II-4: Create and edit Clustertables, edit Scoring**

## VII. Extensibility

In addition to the implemented SPICE algorithm the tool offers a very flexible interface for the implementation of different algorithms for the calculation of the results. A standalone program may be programmed which calculates the results to be visualised by ASAP-B. The program is launched from within ASAP-B and ASAP-B communicates with the standalone program via an ASCII-file. The calculated results may be visualised by ASAP-B (histograms).



# EXCUSES

## An Experiment for Use Cases in Capturing User Expectations in Software Development Projects

Alessia Billi

Franco Correrini

*Sodalia SpA, Trento, Italy*

The EXCUSES Project has experimented, with respect to a more traditional and less formal approach in the domain in which Sodalia operates (i.e. telecommunications), the effectiveness of Jacobson's Use Cases technique, enhanced by a rigorous formalism, for capturing the user expectations and preliminary requirements in the specific context of the initial phases of an iterative development process which is object-oriented and reuse-oriented.

The success of the experiment has been assessed both qualitatively, by having all participants to the experiment (i.e. customers, requirements team members, requirement users) fill in a questionnaire, and quantitatively, by measuring the productivity of the project staff (in terms of Function Points per person/month) and the quality of the requirements produced (in terms of number of defects introduced by the Use Cases).

### 1. Introduction

EXCUSES is an ESSI Process Improvement Experiment (n. 21532) which has been carried out from January 1, 1996 to March 31, 1997. The main emphasis of the experiment was the introduction of the Use Case methodology [1] into Sodalia's development process in order to deal with users' expectations and requirements in the early phases of development. The Use Case methodology constructs a user-centric view of a system represented by a model containing a representation of the system behaviours and roles filled by outside actors (humans or machines) who interact with the system in various ways. Since it is user-centric, it is an efficient means of communicating with clients about expectations and requirements, and since it is an object model, it gives a head start to downstream efforts for subsequent analysis, design, and implementation by already having partitioned/structured a complex system in a straightforward way. Thus, the model is robust enough to stand up to technical demands, but simple enough to be related to a user's view of the system.

The objectives of the project were to:

- Adopt the "Use Case" and "Scenario" techniques to model a complex software system by capturing its functional requirements, identifying its logical high level architecture, producing the sub-systems diagram for depicting the static structure of the system, and tracing the events which explain the interactions among the sub-systems. Such model has then become the basis for developing the entire system and the project staff (i.e. analysts, architects, designers, testers, manual writers) has been called on judging the pros and cons of the technique by filling in a questionnaire.
- Assess the results of the activity analysing constraints, difficulties, and benefits.
- Report the conclusions in order to improve the current methodologies and techniques for performing the process aimed at capturing requirements.

Such objectives have been measured by applying the Sodalia Software Metrics model to the Concept Exploration phase and to the subsequent activities. Such measurements have been achieved in

comparison with those of other comparable projects (internal to Sodalia) to assess the success of the experiment.

## **2. Starting Scenario**

Sodalia produces telecommunications software. From its inception, Sodalia's upper management has committed to proven leading edge technologies & methodologies for the efficient software production, including iterative design with rapid prototyping, object oriented methods for analysis, design & coding, and development for & with software reuse. So far, Sodalia has developed a reuse repository management tool for company-wide usage, and the Process & Methodologies Group has laid plans for a series of process innovations including a continuous reformulation of the Sodalia's software development process (SIMEP: Software Integrated Management and Engineering Process). These innovations take the form of guidelines which are designed to be compliant with the SEI Capability Maturity Model Level II and III prescriptions and address such areas as project management, configuration management, verification&validation. Sodalia has been recently assessed at Level III of the SEI CMM and has also achieved the ISO 9001 certification in 1996.

At present, Sodalia's development process (i.e. SIMEP) does not adequately address the process step by which the user's expectations are gathered, structured, and re-synthesized into requirements. Moreover, the methods already identified to support SIMEP (e.g. OMT) do not provide (yet) any support for such user expectation directed activities.

Currently, Sodalia produces, in close co-operation with the customers, the expectations/requirements of the software systems that have to be implemented. The expectations/requirements description is textual, and its generation follows only ad-hoc methods without any formal approach. The only methodological support is represented by a well defined requirement form which summarizes all meaningful information regarding each requirement, such as its stability, its understandability, its acceptance criteria, and so on. From past experiences, Sodalia has observed that an Object Model of the requirements is too complex for the practical discussions with users that must occur to verify, correct, and augment the requirements. On the other hand, supplying the users with a mere textual description of the requirements proves insufficient to communicate the requirements in an understandable way and does not facilitate in gathering and discussing a complete and consistent requirements set. Moreover, textual requirements do not facilitate traceability to subsequent object-oriented models of the life cycle.

## **3. Description of the Experiment**

The experiment has been structured on the following main steps.

- Production "Guidelines for Use Cases in User Expectations Gathering" by tailoring the document to the specific needs of an internal baseline project.

The purpose of the guidelines document was to provide the necessary directives to the people involved in the project in a step-by-step application of the Use Cases technique in order to define the system requirements and, at the same time, to outline the macro-functional components that make up the system.

- Conduction of the experiment by the baseline project staff, ensuring the necessary training by the methodologies area.

Through a daily support by the methodologies staff, the baseline project staff succeeded, first, in the identification and definition of very high level system requirements (i.e. expectations) by application of the Use Cases, and, then, in the production of more detailed requirements/Use Cases to reach a necessary level of understandability, clearness, and conciseness such to make possible and unambiguous their implementation.

- Analysis of achieved results both on a qualitative level (through a set of questionnaires filled in by the project staff who dealt with the Use Cases) and on a quantitative level (based on measurements taken at the end of the requirements definition phase).

A survey involved all project staff who dealt with the Use Cases (from the customer to the analysts and to all requirements users) to qualitatively evaluate the outcomes of the experiment (e.g., whether the expectations were met, whether the customers were satisfied, whether the perceived product quality were improved from past projects). The survey was based on a set of questionnaires produced as part of the project activities. The quantitative assessment, instead, was based on measuring the quality expressed by the System Requirements document (i.e., the number of defects detected in peer reviewing it) and on measuring the productivity of the baseline project staff (measured in function points per person/month).

### ***The Baseline Project***

The experiment baseline project is developing a product, called AutoMapper, to be used within a broad Billing application in order to translate a general input data format to an output format according to a set of configurable translation rules to be defined and provided by the user. AutoMapper is an intelligent translating system. This system is able to compile translation protocols defined by a config user into translation source code. This source code, when compiled, is able to translate data of one format into data of another format. The principle underlying AutoMapper is simple. Using a user-friendly graphical user interface the user specifies the information required to perform a specific translation. AutoMapper then automatically compiles this information into source code. The source code is transferred to the mainframe environment and compiled into an executable. This executable is now available on the mainframe and ready to perform data translation. Any number of translation executable can be created in this way, each one tailored to a specific translation process.

Although the interaction between AutoMapper and the user, being limited to the graphical user interface, is not heavy neither complex (that would have exploited the Use Cases capabilities at the best), this baseline project has been considered in any case as an opportunity for verifying the capabilities of the Use Cases method on (almost) any type of product.

## **4. Results Assessment: Qualitative Analysis**

At the end of the experiment, all concerned participants (customers, requirements team member and other people involved in the software process in direct contact with the analysts) have been involved in a survey. The survey has allowed to qualitatively evaluate the outcomes of the experiment: whether the expectations have been met, whether customers have been satisfied, whether the perceived quality has been improved from past projects. The results obtained are a qualitative gauge of the efficacy of the Use Cases method at improving the process according to those interviewed with respect to their own past experiences and to what they have observed during the experiment.

The survey has been based on a set of questionnaire and conducted by the University of Trento, which has acted as an unbiased observer external to the software development process. Information on the qualitative improvements due to the introduction of the Use Cases method has been collected taking statistical data targeting different aspects of the experiment:

- relative satisfaction with the communications process (direct contacts among people);
- relative improvement of the quality of the process and of the results achieved;
- relative improvement in productivity (considered as time efficiency).

Different roles in the software development project have been identified and the questionnaires have been customized for each role defined aiming at collecting different and detailed data. The roles which have been identified are: client/user, analyst, any other role involved in the software process activities that dealt with the Use Cases (e.g., architect, test designer).

Moreover, in order to correctly interpret each questionnaire, the compiler's knowledge and background concerning the Use Cases (and possible experiences coming from the application of Use Cases in past projects) have been considered and the answers, recorded in each questionnaire, have been adequately 'normalized'.

### ***Questionnaire Results***

The results of the survey have been organized into three different and slightly overlapping categories:

- relationship between customers and analysts;
- technical aspects (Use Cases method and representation of the problem/solution);
- time efficiency evaluation.

A fourth set of questions has covered a more general evaluation of the experiment and possible suggestions for the introduction of the Use Cases method in other projects.

The outcome of the survey can be considered quite positive, as pointed out by all people involved in the experiment.

- It is a common understanding that the Use Cases provide a well-defined requirement description, more comprehensible and logical rather than the traditional textual form which can be easily misunderstood.
- The communication among customers and analysts and, in particular, its clarity, has been improved by the use of Use Cases.
- The representation of the problem and its solution, up to the formalization of the requirements, have been easily understood by the users.
- Customers have perceived that their expectations have been taken into account by the analysts, and this has allowed to simplify the relationships with the analysts, and to make direct contacts more effective.

In general, there has also been a positive response concerning the improved perceived quality of the work and the time efficiency reached with the introduction of the Use Cases technique.

#### ***Customers-Analysts relationship***

In the customers' view, their negative feedback and objections concerning the description of the problem and of the solution proposed by the analysts have been taken into account and promptly translated into real changes. The analysts have organized and carried on the concept exploration phase and the respective communications effectively.

From the analysts' point of view, the customers have easily understood the representation of the problem, without pointing out particular troubles. Effective discussion with the customers has taken place concerning modifications and corrections to the description of the problem and the formalization of the requirements. Where, instead, the analysts' expectations have been slightly disappointed lies in considering the Use Cases as a means for structuring the interviews with the customer: this might lead to the identification of a clear limitation of the Use Cases technique, although we have also to take into account the poor familiarity of the analysts with it.

#### ***Technical aspects***

The Customers have reported that in some situations the representation of the problem has not been completely exhaustive, and perhaps this is connected with the structure of the interview that has been conducted by the analysts.

On the other side, the analysts have emphasized that the representation of the problem given to the customers has matched exactly their expectations, and has been exhaustive in this sense. The representation based on the Use Cases has allowed the client to understand all the aspects of the problem and to identify further characteristics of it, deepening the analysis in a more straightforward and intuitive way. The analysts have found the Use Cases extremely useful in the initial stages of the concept exploration activities, i.e. in identifying the external actors, in envisioning how the product would be used, in identifying blocks/areas of functionalities. The analysts have also pointed out some difficulties in identifying an adequate decomposition level of the Use Cases, especially at the beginning of the concept exploration phase. Moreover, some users appeared to have troubles with the Use Cases terminology, probably due to a lack of experience and specific knowledge of the Use Cases technique.

Other people involved in the experiment has indicated that the Use Cases method has helped in understanding all the aspects of the problem and has allowed to identify further characteristics of it.

Moreover, it has improved the definition and formalization of the requirements and of the input received from the analysts.

An aspect that could be improved is the traceability of the requirements: it has been indicated that the Use Cases technique could be more appropriately used, or perhaps expanded, in order to really support the traceability of the requirements in the software process and product.

#### ***Time efficiency evaluation***

As for time efficiency, the customers have spent less time (compared to previous experiences in projects carried on without the Use Cases method) in direct contacts with the analysts. This is confirmed by the analysts' point of view with a declared reduction of time spent in direct contacts with the customers. Less time has also been spent by the customers to understand the solution (or better, its representation) proposed by the analysts.

Finally, other people involved in the experiment have claimed a substantial saving in time spent in direct contacts with the analysts and/or the customers.

## **5. Results Assessment: Quantitative Analysis**

The quantitative assessment was based on measuring the quality expressed by the System Requirements document (i.e., the number of defects detected in peer reviewing it) and on measuring the productivity of the baseline project staff (measured in function points per person/month).

### ***Quality Measurement***

The first measure taken into account in evaluating the experiment results is the quality of the System Requirements document.

Sodalía established a measurement program related to the product quality that allows significant comparison among various projects results. The value considered for measuring the quality is the number of defects detected during the Verification and Validation activities.

As the Use Cases technique is used in the System Requirements Definition phase, for our evaluation we considered the number of defects detected during the System Requirements Inspection. The comparison is possible and the results are significant for the following reasons.

- The Inspection process and techniques are clearly defined and have been being adopted by all projects for over a year. In particular, an Inspection is always performed on the System Requirements as required by Sodalía's Quality System. We can reasonably assume that the same effectiveness for all Inspections.
- The results of the Verification and Validation activities, and in particular of Inspections, are collected, and, by June 1996, they are inserted into a Software Process Database and easily retrievable and comparable.
- All the data related to the size of the developed products are also collected; in this way it is also possible to compare, by data normalization, the results of the Inspections performed in projects of different sizes.

The first comparison has been based on the number of defects found during the System Requirements Inspection of the baseline project with the average number of defect found in four other similar projects; all the values have then been normalized with respect to the final product sizes. The results show that there is a dramatic reduction in the number of defects: the defects found in the System Requirements of the baseline project are 79% less than the average number of defects found, on the same artifact, in the other projects.

A second comparison has been based on the weight of the detected defects in terms of defects criticality. This comparison has been possible because the criticality of each defect type is not subjective, being pre-defined in the standard checklist to be used during the System Requirements Inspection.

The other four projects provided quite similar results and the following average percentages of defects criticality distribution:

CRITICAL 2%    SERIOUS 36%    MODERATE 43%    ESTHETIC 19%

The following are the results of the baseline project:

CRITICAL 0%    SERIOUS 0%    MODERATE 33%    ESTHETIC 67%

According to the standard checklist used in Sodalía for the System Requirements Inspection, most defects types classified as critical and serious are related to the contents of the Requirements, whereas most defects types classified as moderate and esthetic are related to the method and the form of the Requirements. For this reason we can interpret these result as a great improvement of the Requirements specification in terms of contents and understanding of the client/user expectations; the larger number of moderate and esthetic defects number shows that some difficulties have been encountered in the representation of these requirements by the Use Cases, basically due to the lack of experience in using them by the baseline project staff. Another fact to be considered is the uncertain applicability of the checklist for the Inspection of the System Requirements to this new kind of Requirements representation.

### ***Productivity Measurement***

The second measure taken into account in evaluating the experiment results is the productivity, measured in terms of Function Points per person/month. The comparison among the productivity of different projects has been possible only by also examining a set of other characteristics of the product, project, and management. In particular, we have assumed that the productivity can decrease when the product size increases [2]; for this reason we have compared the results of our baseline project with the following reference set of productivity values, depending on the product size, presented by D. Herron, David Conculing Group, to ASM'96:

	50-150	151-300	301-500	501-750	751-1000	> 1000
Productivity Parameter (FP/pm)	18	16	14	12	10	6-8

The figures taken on the baseline project indicated a size of 720 FP and a productivity of 12 FP/pm. This result shows that the productivity of the baseline project falls well within the boundary values for a project of that size.

In this computation we must consider that the baseline project had to tackle the problem of using a new methodology, problem which has only partially been resolved by an adequate training and support provided by the methodologies staff involved in carrying on the experiment who have worked directly in developing the use cases, and whose effort has not been considered in the productivity computation. Nevertheless, one can assume that, if the Use Cases were fully exploited, with familiarity, an improvement in productivity could be expected.

## **6. Lessons Learnt**

The following are the lessons learnt in carrying out the experiment and that have to be interpreted as the benefits and problems related to the Use Cases in the requirements definition activities.

### ***Problems***

- Lack of a rigorous formalism in defining the Use Cases.

According to Ivar Jacobson [1], there is no need of a rigorous formalism for connecting the flow of different Use Cases when a "use" relationship exists between them. At the same time, whereas the relationship is clearly depicted in the Use Cases diagram (where a dotted arrow connects the Use Cases), a solution was needed also for representing the same situation within the steps sequence of the "using" Use Case, in order to show at which point of the process the "used" Use Case starts. The solution we have adopted is to place a direct reference to the identification code of the "used" Use Case in the suitable point of the process depicted by the "using" one (enabling, in this way, a "navigation" through the Use Cases even without looking at the Use Cases diagram).

- Uncertainness in deciding the suitable granularity of the Use Cases.

The risk to be avoided was to obtain a set of Use Cases with too many levels, difficult to be navigated, with a loss of the general context of each Use Case, or, on the contrary, a small set of Use Cases providing a good idea of the general behavior and potentiality of the system, but lacking the necessary details. The final granularity level has been achieved after some attempts. Probably, the current Use

Cases are rather too detailed and structured than less, favouring the completeness of the information rather than its usability.

- No value added by the StP/OMT tool.

The use of the StP/OMT tool has been limited to a very high level definition of the Use Cases representing the functional requirements, whereas all necessary details have been specified in the requirements document which is a contractual document and also includes the specification of the technical, operational, and quality requirements.

### **Benefits**

- 'Modular' starting point for the system design.

The Use Cases has represented the first step toward the identification of all design components of the system.

- 'Common language' for various members of the development team.

A large part of the baseline project staff had to deal with the same Use Cases: the analysts, the architects, the designers, the test case developers, the manual writers.

- High level definition of the requirements during the initial stages (i.e. the requirements team is forced to think at a 'high level').

Dealing with the Use Cases, the analyst was forced to think in the same way as the customer did, that is, s/he had to describe the services provided by the system without mixing up the functional requirements with implementation details.

- Speedy identification of requirements.

Requirements were finalized earlier by improving the clarity of the communication channel back to the user and by covering the dynamic aspects related to the requirements.

- Good overview/understanding of the system to be developed.

The Use Cases provide a more natural partitioning of a complex system into more manageable pieces at the beginning of the project, according to the customer's expectations, rather than later in the design phase, according to arbitrary design choices.

- Starting point for discussion with the client.

The ability of the customer to fully understand the concepts expressed by the Use Cases enabled the analyst to fully grasp the functionalities of the system and to discuss and agree with the customer the proposed solution. Moreover, the Use Case models will be valid in dealing with the same customer again and again for subsequent extensions or new developments - thus having added reuse potential with large pay-off.

- The customer has the early perception of a better quality product.

By improving the clarity of the communication channel back to the user and by covering the dynamic aspects related to the requirements (e.g. the Use Cases can be seen as a sort of prototype on paper, describing the single steps to implement the system functionalities), the perceived quality of the product is improved by impacting the customer satisfaction earlier. That is, the user can see from the clear representation of the requirements that they do satisfy his/her need (rather than vaguely suspecting that they satisfy the need, but waiting for a complete prototype to verify the suspicion). In this way, the total business volume for a firm could increase as a leveraging of customers' confidence in the quality of requirements, since it is widely perceived that good requirements lead to good products.

## **7. Impact on Sodalia's Software Process**

The results of the experiment have confirmed what claimed by Ivar Jacobson, that is, that the Use Cases are the basis for defining functional requirements, designing the user interface, performing integration testing, defining test cases, composing user documentation and manuals. The success of the experiment has proved the importance of the role played by Use Cases in a methodological context and the need for such technique within Sodalia's development process, even more because the system to be produced by the baseline project is not characterized by an intense interaction with the user (situation that would have allowed to get even better results out of the application of the Use Cases technique).

The confirmation that the Use Cases technique is able to satisfactorily integrate and complete the SIMEP guidelines supporting the Requirements Definition phase is already driving the finalization of the next release of Sodalía's development process which will be made official by the end of the year and where the entire Concept Exploration phase (i.e. requirements definition and high level architecture sketching) is supported by the Use Cases technique as it has been tuned in the course of the experiment.

## 8. Conclusion

The general evaluation expressed by all peoples involved in the experiment has been positive, especially regarding the relationships and contacts. Informal feedback gathered during the conceptual exploration phase from the customers and analysts involved in the baseline project has been confirmed by the survey results: the Use Cases method has contributed to speed up their job, especially concerning the relationships among them, making more effective the collection of user expectations. There has been an increased confidence that the object model derived from the Use Cases better implements what the customers really wanted and provides a partitioning of the system on the basis of the customers' expectations, rather than of subsequent design choices.

The effort expended in producing the Use Cases has been judged reasonably low, as compared to the effort spent using traditional techniques. Moreover, the effort to learn the Use Cases method has been reported as moderately low (about 30 hours of training per person).

Most importantly, our quantitative analysis reached the conclusion that, with the same productivity, but using the Use Cases technique, we have drastically improved the quality of the System Requirements. We can reasonably expect that this quality improvement will propagate into the other life-cycle activities and to the final product as well.

## References

- [1] Jacobson, I. et al. "Object-Oriented Software Engineering - A Use Case driven approach" Addison Wesley, 1992
- [2] Jones, C. "Software Benchmarking", Computer Vol.28, No 10, October 1995

## Appendix A - Authors' CVs

Alessia Billi

Working for Sodalía for four years, she has played an active part, within the Methodologies Area, to the definition of the process model to be deployed by the company to discipline and guide project management and software engineering activities. She has been particularly involved in the definition of the company measurement program and of the guidelines supporting the requirements specification and the technical review activities. A previous experience in another company was devoted to the collection and specification of requirements in telecommunications domain. She is *Certified Function Point Specialist* from June 1997.

Franco Correrini

Working for Sodalía for the last two years, he has played an active part, within the Methodologies Area, to a comprehensive, operational, and rigorous definition of the process model to be deployed by the company to discipline and guide software management and software engineering activities. Lately he was specifically involved in the activities concerning the achievement of SEI CMM Level 3 for the production of guidelines to tailor the organisation's standard software process for a specific project. He is currently working for reinforcing the methodological support to the initial phases of the software development life cycle. Previous experiences in other companies include the involvement in the



definition of a company quality system, and in projects funded by the European Space Agency for the definition and construction of a software development environment.

## **Appendix B - Company Description**

Sodalia is a young company specialising in the development of advanced software for the management of telecommunication services and networks. Despite its youth, Sodalia has already achieved a prominent position in the world of telecommunications where it is known as an advanced and innovative “Software Factory”.

Established in May 1992 as a joint venture between Bell Atlantic Corporation and Gruppo Stet of Italy, Sodalia started operations in July 1993 in the city of Trento in northern Italy. In just a few years, Sodalia has established itself as a leading company in the area of telecommunications software development.

Today, Sodalia is a multicultural and multilingual company of 300 people. Its very experienced management team has a combined 145 years in the software industry and 153 years in the telecommunications industry. This team is complemented by a balanced technical staff of 250 software engineers, technical managers, and business support personnel.

Sodalia has adopted the industry standard Software Engineering Institute Capability Maturity Model as a means to achieve excellence in Software Engineering. Sodalia achieved its SEI CMM Level 2 assessment in December 1995. A few months later, in May 1996, the company was awarded the ISO 9001 certification for the development of innovative software for the management of telecommunication networks and services, and for the development of advanced methodologies and software engineering tools.

Most recently, Sodalia was assessed as a SEI CMM Level 3 company, thus making Sodalia one of the very few companies worldwide to have achieved ISO 9001 certification and Level 3 assessment.

# Experience With Teleworking and Outsourcing Management

Martin Hollinetz

Director

TPI : Technologies - Projects - Integration

Graz, Austria

*m.hollinetz@magnet.at*

## Overview

TPI has established an office functioning as virtual enterprise in which different companies work together. It specialises on teleworking concepts and outsourcing projects and has established tele-workplaces through which

- People can work in TPI's office but in fact operate on machines within partner firms
- Companies can connect with video technology to make distributed workshops

The Know How is used to consult regional government in establishing work places for disabled people, as well as to make outsourcing projects for a large German sales company, as well as operating with eastern Europe already, namely the Czech Republic.

For obtaining the slides from this talk and exploiting outsourcing contacts please address:

Mr Martin Hollinetz

TPI

Annenstrasse 23/I

A-8020 Graz

Austria

Tel. +43 316 764615-0

Fax. +43 316 763793

Email: [m.hollinetz@magnet.at](mailto:m.hollinetz@magnet.at)

## **Process Improvement Through AMI (PITA) (a Process Improvement Experiment under ESSI)**

Contact person: Dr. Vassilis Kopanas

INTRACOM S.A.  
P.O. Box 68, 19002 Peania, Attika, Greece  
vkop@intranet.gr

### **Abstract**

*This paper reports on the experiences from an ESSI project, called PITA, carried out at INTRACOM's department developing software for digital telephony applications. PITA introduced the Goal-Question-Metric (GQM) approach via the AMI method and tool, focusing on a typical project as a baseline. Expected benefits from PITA include: reduction of software development cost, shortening of the time-to-market for software products and increased product reliability at customer's site. This paper presents the rationale of the experiment, a brief account of the GQM method and the AMI approach, the objectives and the organisation of the experiment, its relation to other ongoing improvement efforts, the results and conclusions from the experiment, lessons learned and plans for future activities.*

### **1. Introduction**

In recent years one can observe a growing interest in the application of measurement techniques in software project management. This trend can be justified as a systematic approach to overcome the effects of software crisis, such as budget and schedule overruns, low quality, reliability and high cost for software quality [1, 2].

According to many studies on the application of metrics and models in industrial environments, measurements should be focused on specific goals, applied to all life-cycle products, processes and resources and should be interpreted based on understanding of the organisational context, environment and business goals. This means that measurements must be defined in a top-down fashion, they must be focused and based on goals and models.

In line with the above trends and studies, and in attempting to alleviate shortcomings of INTRACOM's existing software development metrics framework, the Goal-Question-Metric (GQM) approach was introduced to the company's software development practices through the AMI method. The introduction of GQM was carried out as a Process Improvement Experiment (PIE) named PITA, in the context of the CEC programme ESSI (European Systems and Software Initiative).

This paper aims at presenting the rationale of the experiment (sec. 1, 2), a brief account of AMI and GQM (sec. 3), the objectives and the organisation of the experiment (sec. 4), an analysis of experiment activities performed (sec. 5), lessons learned from introducing GQM (sec. 6) and a set of conclusions and plans for further activities (sec.7). Section 8 finally, provides a basic set of references related to software measurements, GQM and AMI.

### **2. The software development practice**

The PITA experiment introduces the GQM method to INTRACOM's software development practices. INTRACOM S.A., the leading Greek telecommunications and electronics firm was founded in 1977. It employs 1500 people (650 are specialised engineers). The company produces a wide range of products (both hardware and software) and turn-key systems covering the needs of modern networks: digital switching and radio links, energy control, telecommunications network management.

Other activities include support of installed systems, engineering services, and involvement in several cutting-edge projects in areas such as multimedia systems, image processing and advanced transport telematics applications.

The main area of software development at INTRACOM is the Software Design Centre (SWDC), which mainly designs high quality software products for digital telephony applications. SWDC which hosted the PITA experiment employs 180 highly qualified and specially trained software engineers / designers. Various process improvement activities are pursued at this department within a formal Policy Deployment scheme. Also, software process capability assessment has been effected and upgrade activities have been initiated. Inspections are carried out using a specific process integrated in all stages of the model used for software development. Testing is carried out in specially operated test sites. The utilised software development method is a variation of the classic 'waterfall' paradigm. The software designers attend an initial training course on the development method, tools and the application field.

Prior to the experiment, a basic set of metrics were being used providing a high-level view of the design process and product quality, as well as a basis for bench-marking. However, they did not constitute a set adequately developed and closely linked to actual experiences and problems. Measurements taken within that immature metrics frame had shown some trends but the necessary analysis was insufficient and the results were inconclusive. In addition, the utilised metrics tended to reflect only generic upper management concerns and viewpoints. Thus, metrics usefulness and suitability was often questioned and their acceptance by involved personnel could not be taken for granted, especially when these metrics contradicted everyday experiences. At the same time, there were not adequate measures for monitoring and controlling the progress of improvement actions.

### 3. Overview of GQM and AMI

Several efforts related to software measurements are reported in the literature [3, 4, 5, 6, 7, 8, 9]. Among them, the **Goal-Question-Metric (GQM)** approach [8, 9] provides a well-defined measurement mechanism. This approach has been developed and proposed for application at software development organisations and is based upon the assumption that for an organisation to measure in a purposeful and efficient way, it must firstly specify the goals for itself and its projects, then it must trace those goals to the data that will define those goals operationally, and finally provide a framework for interpreting the data with respect to the stated goals. GQM was originally defined for evaluating defects for a set of projects in the NASA Goddard Space Flight Centre environment. Since then, the use of GQM has been expanded to a larger context.

The result of the application of GQM is the specification of a measurement system targeting a particular set of issues and a set of rules for the interpretation of the measurement data. The GQM measurement model has three levels:

1. *Conceptual level* (GOAL): A goal is defined for an object (product, process, project or resource), for a variety of reasons, with respect to various models of quality, from various points of view, relative to a particular environment.
2. *Operational level* (QUESTION): A set of questions is used to characterise the way the assessment / achievement of a specific goal will be performed based on some characterising model.
3. *Quantitative level* (METRIC): A set of data is associated with every question in order to answer it in a quantitative way. The data can be objective (e.g. person-hours spent on a task) or subjective (level of user satisfaction).

A GQM model is a hierarchical structure starting with a goal, that specifies purpose of measurement, object to be measured, issue to be measured and viewpoint from which the measure is taken.

The goal is refined in several questions, that usually break down the issue into its major components. Each question is then refined into metrics. The same metric can be used in order to answer different questions under the same goal. Several GQM models can also have questions and metrics in common, provided that when the measure is actually collected, the different viewpoints are taken into account correctly (i.e. the metric might have different values if taken from different viewpoints).

GQM provides a method for top-down metric definition and bottom-up interpretation. The GQM approach can be used as stand-alone for defining a measurement program or, better, within the context of a more general approach to software process improvement.

The **AMI method (Application of Metrics in Industry)** [10] is based on the GQM approach and provides a common-sense framework for quantifying software projects. This framework comprises four principal phases:

1. *Assessment* of the organisation's environment to define primary goals for measurement.
2. *Analysis* of the assessment results and primary goals to derive sub-goals and the relevant metrics. The analysis is formalised as a goal tree with a corresponding set of questions to which these metrics are linked.
3. *Metrication* by implementing a measurement plan and then processing the collected primitive data into measurement data.
4. *Improvement* as measurement data are exploited and actions are implemented. Comparison of the measurement data with the goals and questions in the measurement plan provide guidance towards the achievement of the immediate project goals. When measurements show that a goal has been achieved, there is enough improvement attained to reassess the initial primary goals. One improvement cycle in the AMI method is then considered to be complete.

AMI is designed to be flexible enough to employ existing, proven techniques, such as GQM and Software Process Assessment based on SEI's CMM [11]. With the AMI method, the number of metrics that need to be collected is focused on those that correspond to the most important goals. Thus, data collection and analysis costs are limited to the metrics which give the best return. On the other hand, the emphasis on goals and business objectives establishes a clear link to strategic business decisions and helps in the acceptance of measurements by managers, team leaders and engineers.

AMI has a quite flexible structure which allows for it to be easily integrated with other quality and improvement models that may have been already adopted by an organisation prior to AMI. Such is the case, for instance, with CMM which is actually suggested by AMI as the assessment framework to be introduced, if it is not already in use. AMI is designed to support CMM in an effective way by facilitating the implementation of the 'Measurement and Analysis' Common Feature for all Key Process Areas of CMM.

Another such case is ISO 9001 (together with ISO 9000-3 guidelines or even the newer framework provided by ISO 12207 for Software Life Cycle Processes). In this case, AMI provides the necessary foundation for addressing requirements for 'statistical techniques' (ISO 9001), 'measurement' (ISO 9000-3) or 'process improvement data' (ISO 12207). At the same time, during AMI's first step (assessment), besides applying techniques such as CMM assessment, it is recommended to take into account information produced by audits and reviews clearly relating to ISO standards compliance verification.

Based on the experience of AMI users [10], the average cost for the metrication of a project using the AMI approach ranges within 2-5% of the total cost for the software development, decreasing dramatically with the maturing of the metrication process itself (due to reuse of experience). In any case, this cost is much less when compared to the cost of bottom-up metrics approaches, which collect and analyse vast amounts of data.

The AMI method was developed in the context of the AMI project which was funded through the ESPRIT programme to make the technology and techniques of software metrics and measurement available in a simple, straightforward and understandable form that could be easily implemented. The consortium which developed AMI was led by GEC-Marconi and included Alcatel, Bull AG, Objectif Technologie, TECHNOS and South Bank University's Centre for Systems & Software (CSSE).

#### **4. PITA objectives and organisation**

The PITA experiment introduces the Goal-Question-Metric approach via the AMI method and tool to INTRACOM's software development practices. The objectives of PITA experiment can be grouped as follows:

- establishment of a systematic GQM-based framework for software metrics
- evaluation of the use and effectiveness of AMI approach to support software metrics

- introduction of a formal assessment (SEI/CMM based) to establish software development process maturity
- identification of process improvement areas and monitoring / support of corresponding improvement actions
- dissemination of experience and know-how on AMI and GQM-metrics, both within INTRACOM and its group, as well as in the Greek and international markets
- continuous monitoring, evaluation and dissemination of results collected from mechanisms established by PITA

A number of benefits are expected from the implementation of the PITA experiment. These include: enabling and focusing improvements, better support and focused follow-up for scheduled CMM assessments and increased capability in achieving goals and controlling processes. Furthermore, it is expected that PITA experiment will result in better software project planning and control, prevention of defects due to requirements and improved product quality. Finally, PITA will contribute to increasing the awareness and motivation of personnel for software process improvement (SPI), as well as, to providing more opportunities for technology transfer and business partnerships.

In the context of the PITA experiment, GQM was integrated as necessary with the selected baseline project. Baseline project selection was based on the availability of new projects, their 'typical' character, and their schedule in relation to the PITA effort. To support the smooth and efficient uptake of AMI method, Objectif Technologie's consulting services were used (training, consulting).

As an **initial step**, setting up of co-ordination and, where required, integration of the experiment with the baseline project was effected. This was done both with the project leader and other key responsible persons, as well as with affected line management and quality engineers. During this step the above personnel was trained in the AMI method. From then on, the basic improvement steps as defined in the AMI method were followed, being implemented at the baseline project. The acquired experience and the confidence built-up for the method allowed similar practices to be gradually taken up in other starting projects as well, even before the experiment was completed.

The four main phases of the PITA experiment, correspond roughly to the 4-step Deming cycle: Plan-Do-Check-Act, supported where possible by the AMI method and tool.

The **first phase** in the PITA experiment, involved the performance of an assessment focused mainly on the baseline project. All relevant information from any preceding projects, any special restrictions, previously available measurements and audit results were taken into account together with the SEI CMM model requirements, ISO 9001 requirements and ISO 9000-3 guidance. The outcome was the definition of primary goals for measurement. In this phase, managers were involved coming both from the baseline project and from other projects as well, representing the viewpoint of the whole software development centre. This activity was co-ordinated by a metrics promoter together with the external AMI consultant.

The **second phase**, concerned the analysis of the primary goals to arrive at sub-goals and relevant metrics. This was formalised as a goals tree with corresponding questions leading to the metrics. This activity was generally carried out by the metrics promoter, the AMI consultant, the project leader, the quality co-ordinator and other experts as necessary.

The **third phase**, concerned the actual metrication, with the introduction of a measurement plan, collection and verification of primitive data which were processed into measurement data. These tasks were initiated and co-ordinated by the metrics promoter with the participation of project personnel.

The **fourth phase** concerned managing the improvement resulting from the implementation of the measurement plan. Goals and questions were used as a guide for the evaluation of measured data and the implementation of corrections or modifications, where necessary. Different sets of data were correlated to validate underlying hypotheses.

In parallel to performing the fourth phase - focused always on the baseline project - the overall evaluation of the degree of success of the experiment took place. All kinds of data stemming from the AMI pilot implementation were utilised for the evaluation. In a follow-up to this phase, data are also becoming gradually available from other AMI implementations in newer projects. As part of the fourth phase, information was disseminated and expertise was transferred to other areas of application, as well as in other departments involved in software development.

The **baseline project** for the PITA experiment was part of a wider project called *FM phase 4.1 Design Project* (FM-p4.1). The purpose of this project was to develop a telecommunications software system for the French market, with a scheduled delivery for June 1997. This system is an improved version of the general part of the operating system for digital telephony exchanges focused on some additional intelligent network functions such as handling of service requests and voice messaging. Additionally, ETSI-ISDN specifications were taken into consideration. This software system will run on Ericsson's AXE-10 digital telephony exchanges.

Baseline project activities were divided among a group of Ericsson subsidiaries and co-operating companies, each undertaking the development of parts of the product's functionality (multi-site development, is a common practice for Ericsson's software development operations).

In the FM-p4.1 project, INTRACOM was responsible for developing a part of the Operation & Maintenance subsystem of the final product. INTRACOM's work focused on Functional Design, Detailed Level Design, Unit Level Test and Function Test phases of the software development process. Function testing was carried out as an autonomous activity. Average resource allocation for the FM-p4.1 at INTRACOM was 10 persons (including a management function) and the budget expressed in development effort was 1,400 person-days.

## 5. Conducting the experiment

As a first step in the AMI loop, the software process in the software development department was assessed, based on the CMM model. CMM was chosen for the following reasons:

- its recognition as a reference model by many industrial companies and particularly by Ericsson
- the results of a CMM assessment are considered as a credible picture, indicating priorities and general recommendations, as well as being a reusable reference in the future
- CMM is highly compatible both with general INTRACOM's software improvement efforts and the AMI approach

The CMM assessment produced findings that concerned the whole software development organisation. As a reaction, a Software Engineering Process Group (SEPG) was established to address the findings, promote and co-ordinate improvements. The implementation of AMI was since assigned under the overall monitoring authority of this SEPG.

After the CMM assessment, findings were checked with baseline project members and used as a first source of information. Additionally, several other sources of goals were explored, such as Business Goals, project specific priorities, Policy Deployment initiative results, management review findings and audit results (for compliance to ISO 9001 and ISO 9000-3), reports, conclusions and recommendations from previous development projects or past improvement initiatives.

In any case, no attempt was made to systematically cover full CMM compliance with the use of AMI, because of the experimental nature of PITA and for overheads to be minimised. Therefore, only priority issues were covered, based on the judgement of the baseline project team. Still, in the future, AMI could be applied to systematically address full CMM compliance.

In line with the AMI method, a Goal Tree was constructed, aiming at a better development process, but taking also advantage of previous experiments results and integrating different points of view. This tree which is shown below (see Fig. 1), reflects concerns (valid for both the baseline project and the software development department) about simultaneous search of quality and repeatability in planning and reduced time-to-market. As a first attempt, it was deliberately kept simple, while the possibility remains open for other goals to be later included.

The approach taken for implementing AMI was influenced by two different but interrelated factors: the nature of the product from a technical / engineering viewpoint and the business environment together with its priorities.

The software engineering aspect concerns implementation of functionality through software which abides to strict rules, procedures and requirements. Implemented software will have to be integrated within very large systems incorporating critical operations and therefore, demonstrating high level of reliability.

The emphasis is put on early defect detection and prevention, with main focus on implementing systematic and extensive inspections. In this respect, even though there were some goals reflecting

CMM Level 2 key process area improvements (e.g. planning and tracking of a project), Peer Reviews from CMM Level 3, which relate closely to technical and product engineering considerations were also addressed as an issue of high priority.

The same argument applies to test efficiency. This is because as long the software design process does not achieve a highly formal and standardised character, there will always be reliance on testing to improve product quality.

Concerning the technology involved in this experiment, even though there exist vast legacy libraries of implemented functions, most of their content is far from a status that could be reused as standard building blocks. Thus modifications required are rather extensive, leading to potentially high defect rates and requiring extra measures for achieving the required quality.

**Primary Goal 1. Improve meeting of internal & final delivery date with adherence to process**

- G1.1 Improve preparation and maintenance of the sub-project plan
- G1.2 Decrease impact of early phases (delays, affected quality) to the later phases
- G1.3 Improve project monitoring

**Primary Goal 2: Improve quality as experienced by the customer**

- G2.1 Improve effectiveness of inspections, reviews
- G2.2 Improve function test to prevent faults slipping through
- G2.3 Improve effectiveness of design
- G2.4 Efficient, adequate technical co-ordination
- G2.5 Increase competence of people

**Figure 1. The PITA experiment Goal Tree.**

Improving software quality is thus a key concern of INTRACOM's business as a result of the intensive competition inside the telecommunications domain. Quality is measured based on customer's perception and at regular check-points. At such intermediate points, the measurements / indicators that were introduced reflect several points of view (quality co-ordination, test, design, technical co-ordination). The selected factors should affect positively both the quality directly built into the product (due to design process and personnel competence), as well as the activities that reduce defects (testing, inspection).

Time-to-market is another key element of INTRACOM's competitiveness and is expressed here through delivery dates. This goal has been expressed as improving achievement of delivery dates both internally and externally. Such an objective should be pursued while keeping a coherent and controlled process.

To reach this goal, two major areas of assessment have been addressed simultaneously: planning and tracking. Since achievement of delivery dates is also the result of the whole project life-cycle, the early phases shall be taken into account as real factors, even if in some cases they are not directly controllable. This is, for instance, the case with requirements related to the baseline project which are handled outside INTRACOM. The impact of early phases is therefore taken as a control factor which cannot be directly improved (being out of the scope of the baseline project), but which is measured and monitored. As shown in the assessment report, higher accuracy in estimation and planning lead to increased plan reliability, ability for efficient use of resources and better co-operation.

This work was documented following the general track of AMI, using documentation templates already used internally or recommended by AMI. An example of tabular representation of a particular goal (which is under the primary goal for improving time-to-market) together with its breakdown is given in Figure 2.

The next step in refining the GQM plan involved identifying metrics for the quality focuses and impacting factors. As an example, Figure 3 illustrates the case of the goal for improving the effectiveness of inspections and reviews (within the primary goal for improved quality). In this Figure, a combination of focus-impacting factor-question-metrics is presented.

The approach taken in identifying metrics did not involve some advanced process control model but it reflected the actual procedure used and the experiences and maturity of the development team.



This principle was observed throughout the experiment since it is a precondition for ensuring acceptance, participation and commitment of the development team, for which improvement goals and opportunities should derive naturally and in a non-obstructive way.

On the other hand, if one introduces an advanced technique such as Gilb’s inspections, as an improvement towards the goal in Figure 3, it will only be natural to use more advanced metrics, based on the mechanics of the new method. This latter approach was outside the scope of the experiment described here, but may be used in the future, AMI inherently having the potential to support such an improvement.

In PITA, the objective was to assess the current process maturity and establish metrics that would entail improvement through better control and exploitation of standard processes or enhanced processes already introduced prior to this experiment.

Analyse	early phases	<i>OBJECT</i>
for the purpose of	decreasing	<i>PURPOSE</i>
with respect to	adverse impact	<i>QUALITY FOCUS</i>
from the viewpoint of	project team	<i>VIEWPOINT</i>
in the environment of	FMP4.1 Project	<i>ENVIRONMENT</i>

Quality Focus	Factors Impacting on Quality Focus	Change/ Control
1. Rework effort due to early phase impact	A. Requirements stability B. Late events, reviews C. Open technical issues	A. Control B. Control C. Control
<b>Baseline/ Current/ Estimated Quality Focus</b> 1. <Current Value of Indicator> <b>Expected Quality Focus</b> 1. < Indicator Target Value>	<b>Factors' Impact on Quality Focus</b> (Note:+/- means positive/negative trend) A. The less the requirements change, the less the project is affected (-T) B. Delaying events can delay decisions and lead to more rework (-T) C. Open issues can force to rework (-T)	

**Figure 2. Goal Breakdown Template.**

At the same time, relatively simple metrics were identified and basic statistics and graphs were used as analytical tools, avoiding complex approaches, obscure to the development team who are the main beneficiaries of the introduction of AMI. Thus, quantification did not advance to a level higher than that of existing processes or beyond personnel experiences and maturity.

<p><b>Goal:</b> Improve effectiveness of inspections, reviews</p> <p><b>Quality Focus: #3</b> Relative Effectiveness of Inspection Defect Detection in all life-cycle</p> <p><b>Impacting Factor:</b> Inspection preparation</p> <p><b>Question:</b> To which degree process and criteria are respected prior to the execution step?</p> <p>[<b>Hypothesis:</b> The document(s) for inspection, according to applied procedure is (are) to be made available to inspectors at least 5 days in advance; if not so, then a postponement is required and the process is still respected according to this factor.]</p> <p><b>Metrics:</b></p> <ol style="list-style-type: none"> <li>1. Percentage of documents handed-over at least 5-days before inspection</li> <li>2. Average actual preparation per inspector</li> <li>3. Average preparation rate per inspector (inspection object size/effort)</li> <li>4. Average total preparation per meeting</li> </ol>
---

**Figure 3. Metrics identification example**

After the metrics definition step, a measurement plan was set-up including a measurements list (as a table), mechanisms for data collection, roles and responsibilities and a detailed list of possible attributes for measurement description as shown in Figure 4.

<p><b>Name:</b> (as in a summary and quick reference table)</p> <p><b>Goal:</b> reminds goal as a textual information</p> <p><b>Collection procedure:</b> how the metric is collected, if more information is required (form and frequencies are defined in the table)</p> <p><b>Presentation:</b> the way data are shown, with practical examples</p> <p><b>Analysis procedure:</b> how the metric is used, how trends (+/-) are analysed</p> <p><b>Responsibilities:</b> for collection, for analysis</p>
---

**Figure 4. Measurement Description Attributes.**

Regarding the effort spent for the above tasks, the CMM assessment was performed during one week (excluding the time spent for assessment-related preparation and training). It involved approximately 50 people from all the levels of INTRACOM's software developing organisation and 5 full-time team members. Goal tree derivation and documentation up to the measurement plan preparation involved partially 7 persons, and lasted approximately 4 months. From then on, the project team proceeded with measurement collection, presentation and analysis. The final evaluation, involved a wider improvements addressing team (within the SEPG scope of activities).

A particular aspect of this effort involved establishing an automated on-line environment for data collection, which greatly facilitated the work of the development team and greatly reduced the measurements overhead.

## 6. Experiences from the experiment

Due to INTRACOM's lengthy software development cycle (18 months till field operation is well under way), data for metrics that have been defined in the baseline project, are still flowing in. The data collection period will be considered finalised towards the end of 1997.

At the same time, other later projects where AMI has been or is being introduced are producing data and thus, a measurement baseline for process capability indicators is gradually being built and allows a realistic and systematic planning and tracking of quantifiable improvement goals.

Metrics collected from the baseline project indicate an improvement in many indicators, from the ones estimated on partial data available from the past. However, it is not possible at this point, to correlate those improvements, with the AMI application (since this a first application). On the other hand, since this was a pilot application, one should consider the possibility for a Hawthorn effect (enhanced motivation, carefulness which may diminish in future applications).

In any case, the results need several application cycles to become conclusive and stable. In the long run, trends will appear and will be possible to be validated in a positive way, that could be correlated to the result of analysis, feedback, identification of problem areas by development teams, corrective actions etc.

During this exercise and based on experiences and results obtained up to this point, some issues proved to be of particular importance and lessons were learned thereof. A number of important points are summarised below.

AMI method's power has been proven as a means to consolidate various ongoing improvement activities. It is a fact that an organisation like INTRACOM found it necessary or useful in the past to embark on several such initiatives. These were planned and effected in a more or less formal way and conformed to different models and approaches. One problem often experienced with these activities is that they were not consistent with each other. Also for each activity, planning and monitoring were rather inadequate and their overall effectiveness was limited. On the other hand, measurements been collected were not clearly associated to particular objectives or sometimes variations of metrics were used for similar goals, increasing complexity and decreasing comprehension by development teams, eventually having little or no use. The AMI approach provided the unifying structure that was missing for clearly identifying and quantifying goals and for gaining visibility towards their progress.

A particular strength of the AMI approach is the systematic identification of goals based on an assessment of current practices. A suitable form of assessment is the SEI/CMM model (or equivalent). The performance of SEI/CMM assessment at INTRACOM led to a report that presented the findings of weak process areas and included correction / improvement recommendations. Both findings and recommendations in the report were evaluated by INTRACOM management and key personnel, leading to plans for actions to achieve improvements.

This course is co-ordinated and monitored by the SEPG which was instituted as the main improvement body. Implementation of such actions are effected by adopting practices and formalisms being successfully introduced within the PITA baseline project. In this way, process improvements are being significantly facilitated and supported by the application of AMI, combined with CMM-based assessments. In addition, other objectives concerned with quality assurance (ISO-9001 based), productivity and lead time reduction are also being integrated in a unified AMI approach to software improvement / metrication.

In cases where there are no extensive and reliable historical data to establish baseline process performance and/or the process is immature and unstable, it was necessary to use essentially *knowledge* instead of *improvement* goals. The focus in this case is on establishing a baseline process performance (to enable better control) instead of attempting big improvements and changes.

Still, the use of even tentative target values for the various measured goals is important to enable indicative analysis of results and validation of the GQM model that was derived. Thus, in all cases during the PITA experiment, target value ranges were defined, either based on historical data when available, or on assumptions and hypotheses that had then to be validated themselves.

This is more or less a common occurrence at this stage of maturity in applying AMI. In such cases, more than one cycle of application will be required before the GQM plan and respective performance baseline data stabilise and are validated enough to become reusable in a way that could support planned improvements.

The experience from PITA shows that various goals and measurements may exist before a systematic measurement plan is introduced based on AMI. Thus, as a first step, all such pre-existing goals and measurements should be listed and evaluated. In general, careful and systematic co-ordination of various SPI activities as well as measurements used should be performed to avoid conflicts and duplication.

When analysing results, especially when processes are relatively immature and not effectively controlled, special care should be taken to avoid easily-reached conclusions and judgements. The development team should be involved in such analysis and conclusions should be validated with them, to avoid a risk of reduced buy-in and loss of credibility for the value of the AMI-related SPI activity.

Thus, in the context of PITA, some favourable changes in the indexes to date, have been analysed but, except for particular instances, no clear conclusions were found whether a positive trend was exhibited.

Still, numerical feedback proved useful in focusing attention on the pertinent issues and especially in these modules where there was significant variation from average. No clear conclusion was drawn even in such cases due to the scarcity of data and to avoid over-reaction to effects that could be attributed to random causes or to the combination of factors (e.g. due to complexity).

Long-term measurements based on late phase results such as testing or field performance, are important in providing the overall picture for both product and process and in validating the derived GQM plan. Such long-term measurements need to be balanced with short-term metrics, based on data and facts becoming available during the progress of the project. These later ones, provide timely feedback to support in-process decision making and enable more effective closed-loop management of project tasks.

Currently, only function test results are collected and give a superficial indication of improvement in some indexes (e.g. effectiveness of inspections) while system test and field use results (defects) are expected later in 1997.

Visibility of GQM results by the development team is very important. Results visibility supports participation and motivation of the development team regarding SPI work. Short-term goals should be set to provide software developers with feedback on SPI results and ensure their involvement and collaboration.

One particular problem in applying the AMI method, especially to long life-cycle projects, is the fact that towards the last phases where system test and operational data are becoming available, most of the project team members have been reassigned to other work (only maintenance/support kernel group is active) to effect meaningful feedback sessions. Still, there has been organisational provision to get as many as possible project members together again to a final evaluation and feedback session.

In the meantime, a permanent organisation being institutionalised under the SEPG and implemented within each application area unit (starting from the one the baseline project belonged), performs ongoing analysis to results collected and provides a continuity in experience, offering input and advice to new projects in order to apply AMI efficiently.

As regards the costs and ROI of the experiment, the general experience of the AMI community is validated by INTRACOM's experience to date, giving roughly a 4-5% overhead to development costs. This overhead concerns initial direct costs to the project, excluding initial training, performance of CMM assessment, institutionalisation of SEPG, establishment of on-line facilities to measurement collection, general evaluation, administrative and dissemination costs required for the experiment. These are costs related to line organisation's support of the experiment and would be amortised over a multitude of projects taking advantage of the experience.

If such costs would apply to the baseline project, then the overhead would rise to prohibitive levels (30% or even 40%).

On the other hand, it is expected that in new applications of AMI, the direct overheads will be reduced to around 2% of development effort, leaving room to effect other specific improvements at project level (which AMI will actually facilitate and support).

## **7. Conclusions and further activities**

As the PITA project was progressing, results from the baseline project were validated and analysed. Based on such analysis, specific corrections or further improvements to development processes were proposed (by the development team). Next, an evaluation of the derived GQM plan (model) was effected, in order to validate its correctness and to modify it, if and where necessary.

An overall evaluation of the effectiveness and benefits of the experiment were performed based on latest data. The experience was packaged (in the form of a generic GQM plan and implementation procedure, that can be tailored to the needs of individual projects), so that the approach together with all necessary improvements could be available for reuse within INTRACOM.

Goal tree analysis and metrication based on PITA experience are already being introduced to all new projects to support specific improvement goals like for instance, improving the effectiveness of software inspections and enhancing testing effectiveness.

A significant effort is devoted to dissemination activities, both internally to INTRACOM and externally, for interested parties both in the Greek and the European market of software developers.

Internally, PITA was organised and executed in such a way, so as to ensure maximum transferability of acquired experiences. One important provision was the performance of an organisation-wide CMM assessment, setting a context for improvements covering the whole software developing organisation.

The AMI application on the baseline project, was largely based on the outcome of that assessment and the derived GQM plan is expected therefore, to be easily adaptable for other projects.

Particular goals (e.g. inspections related) and measurements from the derived GQM plan could be adopted/standardised across INTRACOM's software developing organisation. This will allow internal bench-marking, in order to identify and promote best practices.

Other goals identified in various SPI activities are also to be analysed and measured with AMI. In particular, the AMI approach will be used to measure the progress and the effectiveness of another CEU-funded ESSI experiment called PIBOP ('Process Improvement through PSP'), introducing Watt Humphrey's Personal Software Process at INTRACOM's software development practices.

Externally, various dissemination activities to present PITA approach and results are scheduled, including participation in conferences, workshops, seminars, publication in magazines, newsletters and the Internet (WEB page that will be kept up to date). One particular target group involves Ericsson and its subsidiaries, that are seriously engaged in SPI based on CMM. INTRACOM collaborates with those companies, both on software development, and on process improvement.

As a general remark, the approach followed by PITA is quite generic and should be easily replicated or adapted to a wide range of software developing organisations. A particular forum of disseminating PITA experiences, is the AMI User Group, where INTRACOM's experiences and results are compared to similar ones from other organisations having applied AMI.

## 8. References

- [1] B. W. Boehm, "Software Risk management: principles and practices", *IEEE Software*, Jan. 1991, pp. 32-41.
- [2] F. P. Brooks, *The Mythical Man-Month*, Addison Wesley, 1995.
- [3] *METKIT Software Engineering Measurement Teaching Materials*. Information is available on Web page URL : <http://www.sbu.ac.uk/~csse/metkit.html>.
- [4] T. Matsubara, "Project Management in Japan", *American Programmer*, June 1991, pp. 41-50.
- [5] R. B. Grady, *Practical Software Metrics for Project Management and Process Improvement*, Prentice Hall, 1992, ISBN 0-13-720384 -5.
- [6] R.B. Grady, *Software Metrics: Establishing a company-wide program*, Prentice Hall, ISBN 0-13-821844-7, 1987
- [7] K. H. Moeuller and D. J. Paulish, *Software Metrics: A Practitioner's Guide to Improved Product Development*, Chapman & Hall, 1992.
- [8] V. R. Basili, G. Caldiera, H. D. Rombach, 'The Goal Question Metric Approach', *Encyclopedia of Software Engineering*, volume 1, John Wiley & Sons, 1994, pp. 528-532
- [9] C. Gresse, B. Hoisl, J. Wust, *A Process Model for GQM-based Measurement*, Kaiserslautern University (Postfach 3049 D-67653, Kaiserslautern, Germany), STTI Report STTI-95-04-E
- [10] K. Pulford, A. Kuntzmann-Combelles, S. Shirlaw, *A quantitative approach to Software Management - The ami Handbook*, Addison-Wesley Publishing Company, ISBN 0-201-87746-5, 1996.
- [11] M.C. Paulk, C.V. Weber, B. Curtis, M.B. Chriss, *The Capability Maturity Model: Guidelines for Improving the Software Process*, Addison-Wesley Publishing Company, 1995, ISBN 0-201-54664-7.

# Analysis of the Performance of Software Businesses with Artificial Neural Networks

V. H. Haase  
University of Technology  
Graz, Austria  
*vhaase@ist.tu-graz.ac.at*

## Overview

One of the key questions is "How is business performance relates to process improvement". The contribution of Dr Haase bases on a large set of data (collected and analysed with the Synquest method) and applies a most recent approach "neural networking" to obtain a relationship with the business performance.

## Abstract

Quality data about business processes in small software companies are analysed using neural network based tools. It is shown that this technology is powerful :

1. To identify "types" of businesses
2. To learn functions on overall performance dependent on specific quality parameters
3. To identify improvement steps which are most appropriate for achieving higher performance

Based on sample data of 51 business units it was found that thorough inspection of early software life cycle phases contributes most to high performance.

This method can be extended to other types of business data, especially to the Balanced Business Scorecard.

For obtaining the slides from this talk and participating in this work please address:

Prof. Dr Volkmar Haase  
Institute for Software Technology  
Münzgrabenstraße 11 / II  
A-8010 Graz  
Austria  
Tel. +43 316 873 5731  
Fax. +43 316 873 5706  
Email: *vhaase@ist.tu-graz.ac.at*

# Distance Training and Co-operative Work through Internet Video-Conferencing

G. Bazzana, E. Fagnoni, G. Rumi

*ONION Communications-Technologies-Consulting, Via L. Gussalli 11, 25131  
Brescia, Italy, email: [gb@onion.it](mailto:gb@onion.it), [ef@onion.it](mailto:ef@onion.it), [gr@onion.it](mailto:gr@onion.it)*

J. Boegh

*DELTA Danish Electronics, Light & Acoustics, Venlighedsvej 4, DK-2970  
Hørsholm, Denmark, email: [jb@delta.dk](mailto:jb@delta.dk)*

E. Van Veenendaal

*KEMA International B.V., P.O. Box 9635, 6800 ET Arnhem and Eindhoven  
University of Technology, Faculty Technology Management The Netherlands,  
email: [E.vanVeenendaal@kema.nl](mailto:E.vanVeenendaal@kema.nl)*

S. Geyres

*SMC International-Division PSTI Evaluation, INNOPOLIS - Voie 1 - Rue de la  
Decouverte -BP 394, F-31314 Toulouse Labège, France, email:  
[geyres@easyline.com](mailto:geyres@easyline.com)*

## Abstract

Software Process Improvement (SPI) concepts and benefits are still poorly known and understood by most software executives in Europe. In particular, SMEs need special attention. The EPIC Project has been launched by SMEs and with the goal to reach SMEs.

EPIC aims at proposing an innovative dissemination approach that is expected to eliminate problems such as: isolation of the companies within their specific application domains, very different development practices and business needs, locality of the market, limited attendance to international conferences or events and narrow-scoped local meetings.

The basic idea of EPIC to reach SMEs is to organise a set of local workshops, focused on precise SPI topics and with an international dimension. The international dimension is important to allow wide exchanges of experience, in order to avoid meetings that would carry a too much narrow view.

The usage of video-conferencing technologies is one of the communication services most envisaged by both large organisations and SMEs. Unfortunately so far such services are not enough widespread due to the high costs of professional solutions and to the low quality of service of naïve Internet-based solutions. While waiting for the guaranteed quality of service that ought to be offered by Internet-2, a

solution has to be found which optimises the cost-quality ratio. This paper describes the set-up and usage of video-conference solutions used by a group of European companies to perform multimedia based geographically distributed and simultaneous workshops about software process improvement.

More details can be found on the WWW at the URL: <http://net.onion.it/>

## Introduction

SPI is a concept that is still not well known and understood by many small or medium enterprises (SMEs). The EPIC Project [1] (European Commission Number: 23659) aims at disseminating pragmatic experiences about software process improvement, with a special focus on SMEs, and by means of multimedia advanced communication technology.

The Project is targeted at middle management and SMEs. It has planned – and started – to organise ten cluster meetings, half of which dedicated to specific application domains and the other half dedicated to specific topics.

Given this target approach, the idea of using video-conference technology has of course been considered by the project team. Video-conference brings all communication potential that is needed to turn traditional local meetings into really attractive pan-European exchange events. The usage of video-conference is envisaged by both large organisations (which see it as a way to minimise travel efforts and expenses) and by SMEs (which would like to take benefit of this to push their business opportunities abroad).

Today, widespread adoption of video-conferences facilities is hindered by the following factors:

- High costs of professional services, often based on leased communication lines and proprietary solutions; this constitute a barrier for SMEs and very often is a limit also for large enterprises: they are all refrained from making huge investments on a solution that could become obsolete very soon and which only allows them to do video-conferencing among a limited number of sites sharing the same technology

*They need to use an open solution suitable for communication with whoever in the world;*

- Low-quality of service of Internet-based solutions, due to the limited bandwidth available at European level; in fact the average bandwidth which most users can experience when connected to a professional Internet Access Provided is such that in principle it could be possible to use some video-conference packages that thanks to high compression algorithms and intelligent image pattern recognition claim to be usable with less than 3 k-bit/sec bandwidth. In practice there is no guarantee of the quality of service, which is almost always of such poor quality that it forbids its usage for professional purposes.

*They need a stable and sufficient bandwidth for smooth exchanges.*

The emerging Internet-2 protocols, together with the progressive strengthening of communication lines, are expected to introduce guaranteed bandwidth, which could possibly result in a solution to the problem. But such protocols will be available to the end-users only in several years and thus there is the need in the meanwhile to define solutions which optimise the cost- quality of service ratio.

This paper describes the usage of video-conference facilities for distance training and co-operative work across several European companies, in the context of a co-operative Dissemination Action performed under the auspices of the ESSI initiative of the European Commission.

The focus of the paper will be on the following aspects:

- Goals of the EPIC Project, focused on Process Improvement topics;
- Business needs and goals of the project;
- Technological solutions adopted;
- Application of such technology for the organisation of multimedia geographically distributed and simultaneous workshops
- Potential for replication and deployment.



## The EPIC Project

### EPIC in a nutshell

The EPIC Project (EC Number 23659) is intended to be very practical, focusing on business goals and needs derived from experiences in the field. It falls within the ESSI Programme (European Systems and Software Initiative), which hosts various projects of very different natures but with an key focus on Process Improvement experiments and dissemination.

The target audience is meant to be middle/high management and in particular Project Leaders, Department Leaders and R&D Managent, practitioners and SPI coordinators.

Special care will be devoted during the workshops to SMEs and non-IT companies (where most of the European software is developed and maintained).

In particular, the workshops are meant to be attended by people whose intention is:

- to discuss their process improvement experience with experts to make sure that the chosen way is the right one;
- to improve timeliness of their projects and the quality of the software products;
- to exchange know-how with ESSI supported process improvement experiments to get new ideas about how to make it better and what to avoid;
- to participate in an up-to-date workshop environment.

### European coverage

The EPIC Consortium includes seven partners: Onion – Italy (Prime); Delta - Denmark, ISCN - Ireland, KEMA – The Netherlands, LGAI - Spain, SMC International - France, University of Paderborn – Germany.

In addition to the countries of the partners, also Austria and Greece will be involved in the workshops, thanks to agreements with local companies. Henceforth the global European coverage of the EPIC Project is very significant, as shown by the following picture.

### Focus of the workshops

As already described, EPIC aims at circulating pragmatic experiences about Software Process Improvement through targeted events run throughout Europe and conducted with the support of the most advanced communication technologies.

The targeted events are subdivided into two sets:

- Cluster meetings focusing on the presentation of pragmatic experiences for specific application domains. Such events have been targeted both to IT and non-IT companies, without constraints on the company size. Considering the break-down of IT companies in Europe, the following five application domains will be disseminated:
  - Embedded software (telecommunications, avionics, devices, etc.);
  - Commercial software (banking, insurance, finance, etc.);
  - Industry (e.g.: manufacturing, engineering, process control, energy, etc.);
  - Public administration;
  - Software houses.

The application domains have been selected in accordance with the high interest shown by software Process Improvement Experiments (PIEs) for such sectors [2].

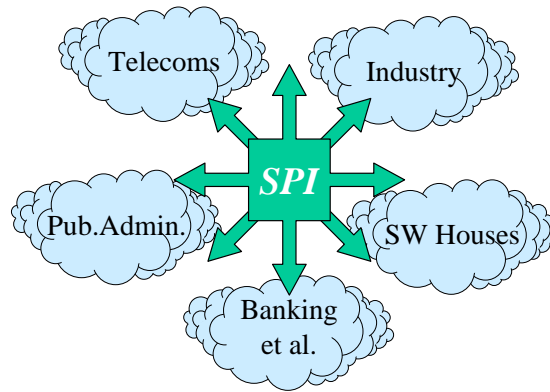


Fig.1 – Application domains covered by EPIC

➤ Cluster meetings on topics strictly related with Process Improvement. Such events will be targeted mainly to IT SMEs and will be oriented toward the impact of Process Improvement on specific issues having a direct influence on meeting business goals. Considering the most frequently asked questions about Process Improvement and the skills of EPIC Partners, the following five topics have been selected for dissemination:

- Process Improvement (PI) and Software Product quality and testing;
- PI and ISO 9000;
- PI Measures and Return On Investment;
- PI and security/ formal certification;
- PI and new technologies.

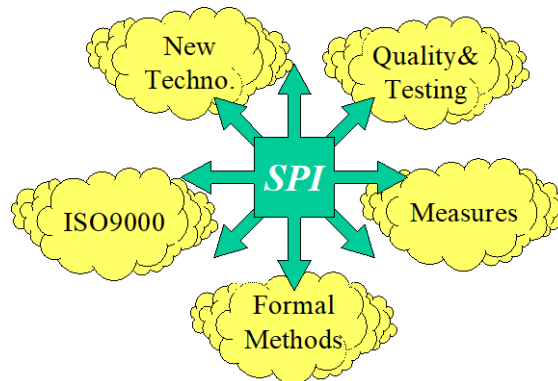


Fig.2 – Technical aspects covered by EPIC

The following table shows the planned dates and venues for the workshops.

Country	Application area / date									
	Industry	Public Admin.	Banking/ Insurance /Finance	Embedded Systems	Formal methods/ Security/ Safety	PI and Sw Quality and Testing	PI measures and ROI	SW Houses	PI and new technologies	PI and ISO 9000
	Oct-97	Nov-97	Dec-97	Feb-98	Mar-98	Apr-98	May-98	Jun-98	Sep-98	Oct-98
Austria			ø					ø		ø
Denmark	ø			ø	ø	ø	ø			
France		ø		ø	ø	ø			ø	
Germany	ø					ø		ø	ø	ø
Greece					ø			ø		
Holland	ø	ø	ø			ø	ø		ø	
Ireland					ø	ø				

Country	Application area / date									
	Industry	Public Admin.	Banking/ Insurance /Finance	Embedded Systems	Formal methods/ Security/ Safety	PI and Sw Quality and Testing	PI measures and ROI	SW Houses	PI and new technologies	PI and ISO 9000
Italy	ø		ø	ø			ø		ø	ø
Spain	ø	ø	ø				ø			ø

Table 1 – Dates and venues of the planned workshops

## Organisational mechanisms

In order to reach and attract the target audience, taking into account their requirements and habits, it is proposed to run the events with usage of Multimedia technologies, in adherence to the following scheme:

- a set of cluster meetings (one set for each selected topic) is organised at the site of selected EPIC partners;
- such meetings are run at the same time by all involved partners (e.g.: on November 4 1997, all cluster meetings on Public Administration will be run);
- at each site the cluster meetings are attended by around ten-fifteen companies, including companies that have run a Process Improvement Experiment and companies that want to know about;
- the material to be presented is put on WWW accessible by all sites some days in advance so that attendees can start looking at it;
- Internet based audio/video-conference facilities are set-up among the various sites;
- Internet based facilities are used to run the “Multimedia-based geographically-distributed dissemination events”.

From an organisational point of view, each event will be managed by a subset of EPIC partners, selected on the basis of target audience, dissemination topics and partners’ know-how.

## Dissemination material

The material to be disseminated is composed of two main parts:

- WWW multimedia material;
- Accompanying supports to the multimedia material.

Concerning WWW multimedia material, an appropriate guide-line has been defined in order to produce consistent material, covering aspects such as:

- structure of the WWW sites (that will have to be easy to browse also for people not attending the events);
- usage of features enabling the co-operative browsing of sites (e.g.: the co-ordinator of the event activates a hyper-textual link and immediately all other sites participating in the event have their site aligned to display the new location);
- careful usage of HTML features in order to ensure portability across the various browsers (e.g.: various versions of Netscape’s Mozilla and Microsoft Explorer);
- moderate usage of advanced interactive features (e.g.: Frames, Java, VRML) in order to have a balance between usability and performance and accurate usage of audio/ pictures in order not to have too heavy pages to download
- set-up of interactive distance learning techniques (e.g.: questionnaires interspersed with navigation; adaptation of navigation to user’s skill and goals, etc.);
- set-up of automatic default navigation path (to be used in a “slide show” approach, whenever needed);
- thorough testing of the prepared material [3].

The choice of the authoring tool (the tool suitable to create the multimedia packages) relied on the assumption that no know-how on HTML had to be required by content-provider of the training material for workshops.

Thus the following features had to be satisfied:

- user friendliness, to reduce the overhead and allow a good productivity;
- good integration and compatibility with MS-Office, to allow an exchange of information between word processor, spreadsheet and authoring programs;
- possibility to build a template, to speed-up the development phase and ensure a good coherence in look and feel across the multimedia packages;
- good conversion in HTML, that must be fast and easy, must keep correspondence with the original document, must guarantee a good quality of the results and must not generate “large” HTML pages.

At the end of a comparative analysis between several tools, Power Point’97 was selected since it matches the listed requirements and is very presentation oriented, supporting also a portable format (presentations can also distributed for off-line analysis). Unfortunately HTML conversion leads to loss of many fancy features of such package, particularly the dynamic ones: sound effects (that can be replaced by adding a background music in each slide, preferably in midi format), slide transition effects, objects animations (that can be replaced by animated GIF) and links (that have to be re-built e.g. by using a client side map).

## Measurement of success

The measurement of success for the EPIC Project will be measured through the following quantitative and qualitative parameters:

- Direct participation of European companies to events: the Project will be considered successful if the events will be attended in average by 10-12 attendees per site. Such figure will be computed as an average over all meetings at all sites;
- Number of PIEs presented: the dissemination events will be based as reference material on the results of the ESSI Process Improvement Experiments. The Project will be considered successful if it will package and present the detailed results of no less than 40 Process Improvement Projects, including no less than 20 ESSI PIEs.
- Availability of dissemination/ training material for the software engineering community at large: the material will be composed of WWW pages and associated textual description as well as consolidated conclusions from meeting minutes. The Project will be considered successful if it will deliver more than 300 WWW pages in a format suitable to be browsed/ understood by the software engineering community.
- Access to WWW Project information: the Project has set-up a WWW site for presenting the events and offering detailed information/guidance. At least 50.000 hits are foreseen on the EPIC WWW during project lifetime (from Jan 1997 to Dec 1998)
- Customers’ Satisfaction Index (CSI): the Project will set-up a mechanism for evaluating the success of events, covering aspects like: interest of subjects, usefulness of the events, effectiveness of the mechanisms used, quality of the domestic arrangements, etc

## Technical goals

The technical goals of the experience can be summarised as follows:

*“To set-up audio-video conference facilities featuring optimal cost-quality of service ratio allowing their usage for distance training and co-operative work”*

The following sub-goals can be derived:

- connection of distributed sites with audio/video hyper-media technology;
- hardware/ software procurement costs for setting-up the infrastructure at each site less with a negligible cost;

- usage of scalable solutions;
- adoption of de-facto standards, not based on proprietary solutions;
- mirroring of the scenarios in various parallel sites.

According to these technical goals the EPIC project has set up its infrastructure. The choice was made on the usage of this approach and of innovative multimedia technologies for the management of multi-site synchronous cluster meetings dedicated to software process improvement subjects.

The goal of the experiment was to organise workshops that would be more attractive to software companies than more traditional events usually are.

As a matter of fact, software development industry in Europe is characterised by a large number of small software producing units, focusing on different application domains and thus with different needs. This results in problems to reach effective technology transfer of pragmatic experiences about best practices. The analysis conducted before the start-up of the Project showed that two barriers exist for an effective technology transfer and dissemination of Process Improvement Success Stories:

- clustering of the companies in specific application domains with utterly different development practices and business needs (it is very seldom the case that a detailed practice that gave good results in a telecommunication company is successful, or even applicable, in e.g. a banking environment);
- locality of the market and limited attendance to international events (looking at the attendee list of international events on software engineering, the number of SMEs and non-IT companies is quite small, in particular foreign companies).

To overcome these problems, an alternative approach could be to organise cluster meetings close to the sites of attendees. This, though being suitable and often successful, suffers of the narrow view that there is a risk to get without confrontation with a higher number of experiences.

Since it is generally accepted that technology transfer of success stories is of great importance, the proposed action aims at removing such barriers while taking into account the requirements of the intended audience and proposing an innovative dissemination approach. This in turn should eliminate the problems of narrow-scoped local meetings.

The requirements to be addressed in order to get more involvement can thus be summarised as follows:

- to introduce new forms of dissemination besides the common tutorial approach applied mainly in conferences, where we too often see an information flow going only from experts to experts;
- to minimise the need for absence from the company headquarters;
- to provide information tailored to the specific company needs;
- to provide pragmatic results applicable to specific application domains;
- to allow room for discussion, exchange of views and comparisons.

This is thought to be accomplished by the organisation of the multimedia based geographically distributed and synchronous workshops, a sample scenario of which is briefly described in the following.

Each local workshop is planned for about an attendance of 15 people. A local moderator prepares all equipment and material on site and co-ordinates the local discussion as well as the discussion between the local site and the remote sites via the video/audio hyper-media connection.

In a cluster, 3 to 4 local workshops are connected together using real-time audio/video equipment allowing to switch communications between the workshops.

Based on the selected technology, each workshop consist of a balanced mixture of local and connected presentations and discussions concerning the above mentioned application domains and topics.

As a consequence, the benefits for the target audience can be summarised as follows:

- exposure to success stories matured in their specific application domain;
- possibility to benchmark various approaches and related outcomes;

- access to up-to-date assets in a very cost effective way;
- possibility to discuss specific issues without barriers; in fact the organisation of parallel events in various sites will allow to accommodate the dissemination approach and presentation style to the specifics of the location (cultural issues, language, etc.)
- possibility to have access to the dissemination material via WWW before and after the events, in order to prepare the participation and, afterwards, “to sell” the outcomes to the internal decision makers.

## The adopted solution

### System Architecture

The envisaged technology infrastructure must be able to reach two main goals:

- allow each company to participate to workshops from their own site;
- guarantee a good quality of service during each workshop.

In order to satisfy the first objective, the solution is to use Internet and de-facto standard tools to publish data. As far as objective two is concerned, due to the impossibility to guarantee a high quality of service using Internet, all involved companies have to set-up a private network, that is an Intranet, based on ISDN, by which the network availability and throughput can be ensured.

To complete the infrastructure, the Internet world and the “Intranet private network” must be connected.

Using such a scheme important benefits can be reached:

- independence of the connection from potential troublesome Internet servers;
- availability of high bandwidth to exchange multi media material (audio and video).

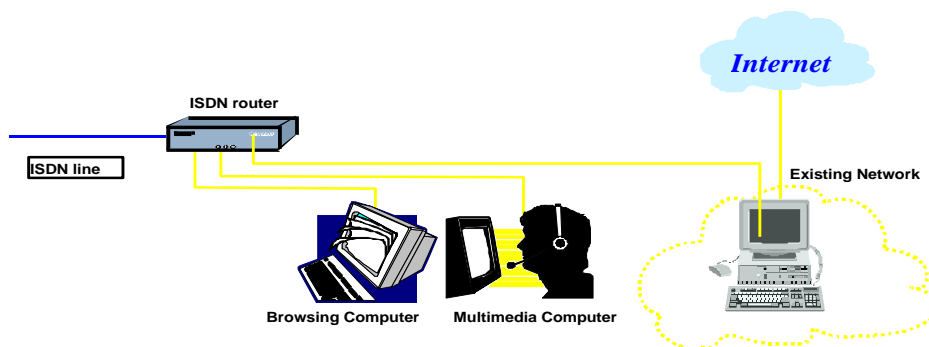


Fig.3 – An overview of the infrastructure needed at each company premises

Henceforth the network infrastructure solution is based on a private TCP/IP network using ISDN connection from each company to a central site (located at Onion's). In fact, all central servers reside at Onion which is permanently connected to the Internet.

Due to the fact that ISDN usage is expensive (companies need to do an International call in order to be connected to the central site) the private network will be used just during network set-up and testing (approximately ten hours) and during each workshops and in particular during interactive sessions (five hours for each workshop in average).

### Technological infrastructure

The following picture gives a more detailed overview of the defined technological infrastructure.

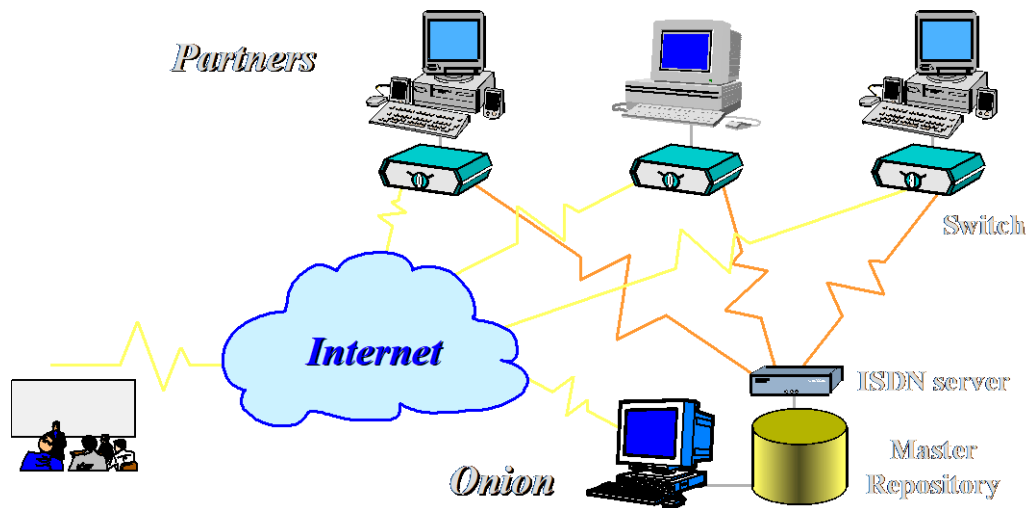


Fig. 4 - Technological infrastructure

The figure shows both the partner site and the Central site (that is ONION) and the connections between them.

At each workshop, all involved partners shall set-up a direct ISDN connection to the central site (making international phone calls) and at the same time use basic Internet for applications which are not bandwidth intensive (browsing, synchronous navigation, etc.).

The complexity of the technical infrastructure is “hidden”, due the fact that most services are directly provided by a Central Site.

In particular, the Central Site provides:

- audio/video “reflector” configuration and management that permit a co-operative videoconference, in contrast with the traditional one-to-one videoconference;
- synchronous browsing feature, that allows all Partners to be constantly synchronised on the same browser page;
- routing, that permit to each partner connected through the ISDN line to be connected to the Internet;
- accounting and security services.

The ISDN solution, combined with a direct access to Internet based on common lines, has been chosen because only such a solution can guarantee the needed bandwidth which allow to have a clear image and a noiseless voice transmission during multi-point video-conferencing (thus involving broadcasting aspects).

The following figure shows the details about the technological infrastructure foreseen for the Central site, which acts as technology manager.

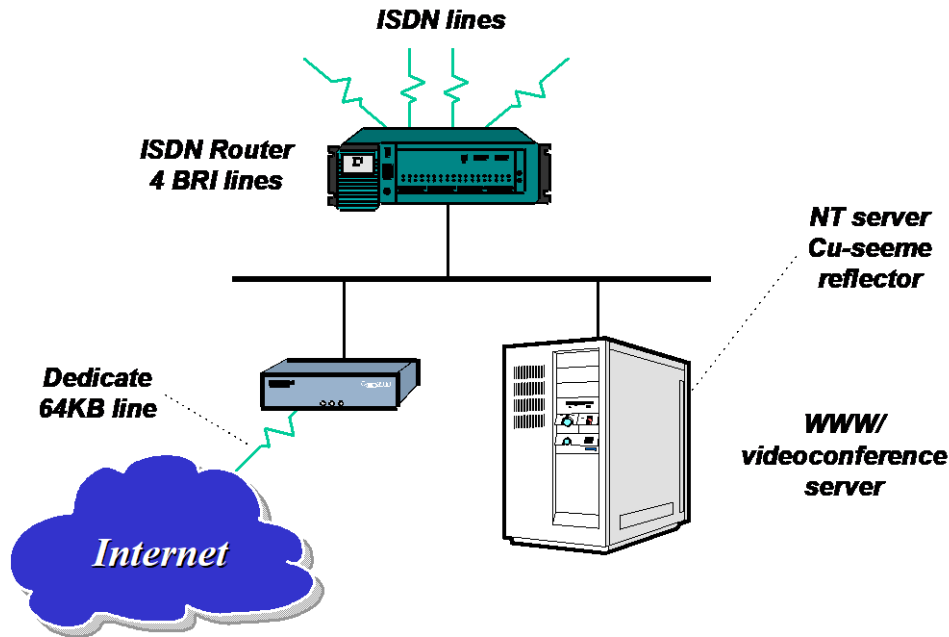


Fig. 5 - Technological infrastructure at Central site

The following picture shows the detailed structure of the technological infrastructure needed at each site willing to have an active role in the workshops.

The technological infrastructure requires the usage of Enhanced CU-Seeme [4] for the video/audio conference and MS-Explorer for the browsing aspects. For sites willing only to act as listeners, the technical infrastructure can be simplified, since there is no need for the Cam-Corder. Other partners will of course need the Cam-Corder for the interactive discussion sessions.

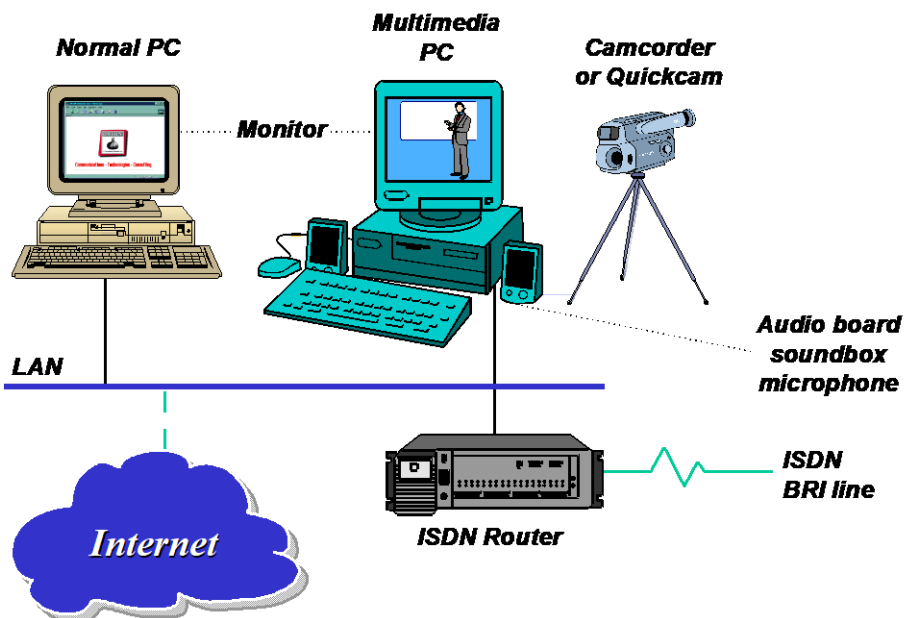


Fig. 6 - Technological infrastructure at each partner site

In order to better understand the correct usage of the EPIC technology infrastructure, the following picture shows the interactions between partners with respect the whole set of activities in which they can be involved during workshops. The figure also shows the position of the participants from outside the consortium.



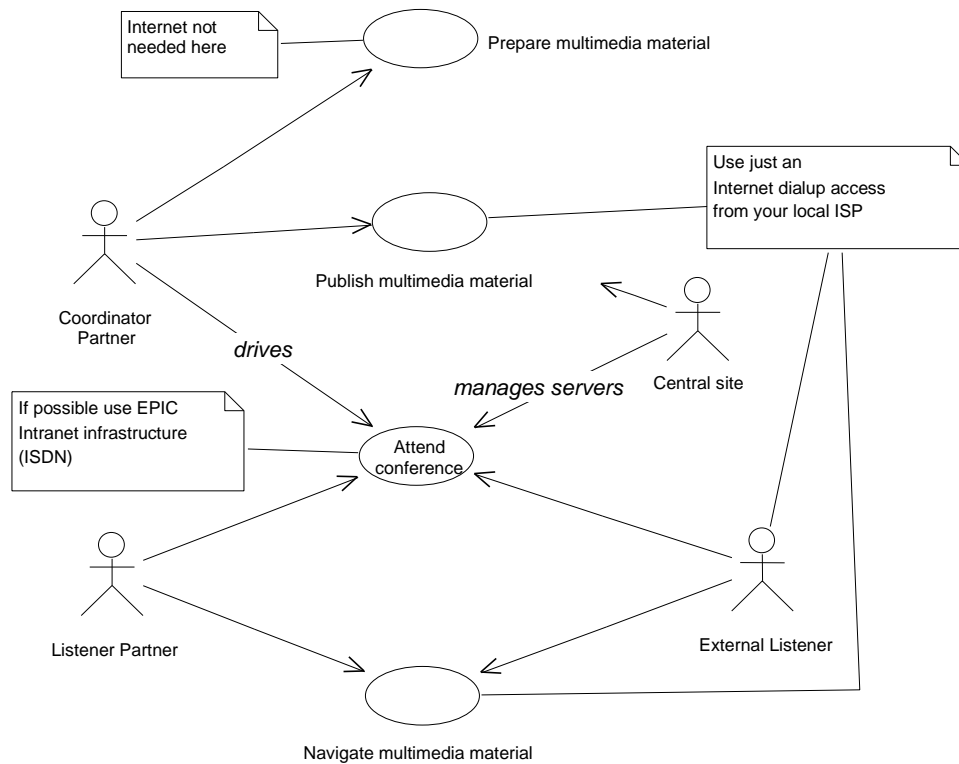


Fig. 7 – Usage of the EPIC technology infrastructure

Looking at the picture, four main use-cases can be identified:

- **preparing multi media material:** in these case neither access to the Internet nor to Intranet is needed because all authoring works can be done “off line” on a local PC. Of course Internet can constitute a valid support when exchanging data and ideas with other people during conference preparation
- **publishing multimedia material:** what is needed in this case is to transfer data from the local PC to the EPIC server. To accomplish this step it can be used a simple Internet connection to a local Internet Services Provider (minimum cost); it is also possible to connect to the EPIC Intranet achieving maximum performances but high costs.
- **attend the conference:** in this case, in order to minimize risks during conferences at partner premises, each EPIC Partner must connect to the EPIC Intranet. The connection will be active just when needed (for an estimated period of 5 hours per workshop). It is relevant to note that it is possible to participate to workshops also from Internet but, in particular for videoconference services, it is not expected to have high performances by using this kind of connection.
- **navigate all conference material:** in this case it must be possible to access multimedia material related to each workshop (both in advance and after). To do this, both Partners and people from their site do not need to access the EPIC Intranet, but can use their standard Internet access.

The most critical case is represented by the running of the workshops. The detailed view is shown in the following figure.

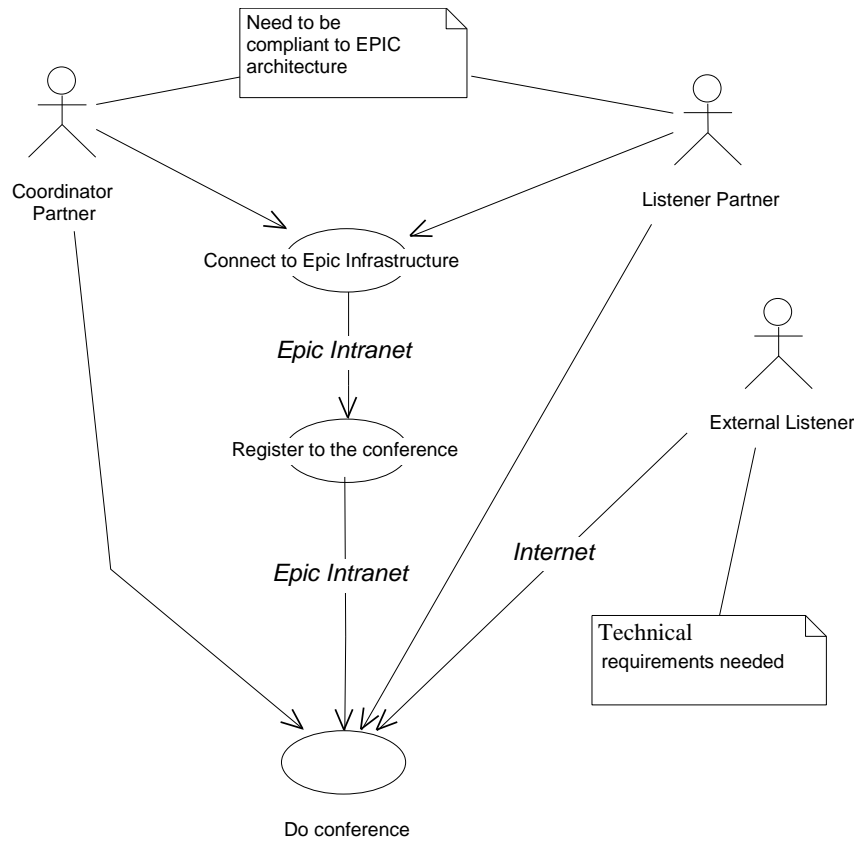


Fig. 8 – Running of workshops

According to what defined, each partner must enter the EPIC Intranet before taking part in a workshop.

Note that using "Dial on Demand Routing (DDR)" features, it is possible to minimize the time of ISDN connection, keeping the line up just when needed.

### Attendance to EPIC Workshops

Attendance is foreseen in one of the three following means:

- By attending the workshop at premises of one of the EPIC Partners;
- By being connected to the workshop from the company headquarters, using an infrastructure similar to the one used by EPIC Partners in the "listener" mode, thus with a guaranteed quality of service; this possibility is offered without any fee but requires to make agreements with the Project Co-ordinator in advance of the meeting, so that the communication facilities can be prepared accordingly to the needs;
- By being connected to the workshop from the company headquarters, using an Internet connection. In this case no special arrangements have to be made in advance (apart from the communication of the access rules to the URL from which the workshop starts) but of course quality of service cannot be fully guaranteed (especially for video transmission).

At the Project WWW (<http://epic.onion.it/>) such possibilities are explained in more details, together with subscription forms.

### Potential for replication and deployment

First of all, we have to stress the importance of the dissemination of best practice experiences to European SMEs. This ought to contribute to a positive influence on the start-up of Process Improvement Projects and thus on the reaching of higher capabilities, resulting in increased competitiveness of the European Industry.

From a technical point of view, the approach described in this paper is in no way restricted to the specific application domain chosen for the piloting.

As a matter of fact, the working scheme is content-independent and thus can be adopted for distance training and co-operative work on whatever subject of interest.

Moreover, the defined architecture is suitable for a number of additional use-cases, among which the following are worth mentioning:

- Outsourcing of communication services, in particular video-conference. In this case the infrastructure set-up in EPIC for audio/ video conferencing services can be offered to third parties for usage under a billing scheme; this is expected to be suitable for companies which do not need an intensive usage of video-conferencing services and thus can find very suitable the renting of “video-conferencing on demand”. From this point of view the ISDN/ Internet scheme which has been set-up in EPIC is very interesting since it does not put severe pre-conditions on the infrastructure of the companies using the service.
- Adoption of EPIC architecture for video-conferencing in the “virtual Enterprise”. In this case the infrastructure set-up in EPIC can be replicated at the headquarters of companies having an intensive need for video-conferencing services and thus willing to replicate at their own headquarters the EPIC Master architecture; this service is expected to be quite appealing considering the economic savings of the EPIC approach with respect to other video-conferencing services currently offered on the market. Notwithstanding, in order to be successful, this service will need a higher quality of service than the one currently planned in EPIC: this does not seem to be a problem since the defined architecture is modular and can take advantage from higher communication speed ensured by “bandwidth on demand”
- Provision of an efficient and flexible source of information dissemination to industry as a supplement to traditional courses and seminars; in this case it is expected that the technology could be used as part of consulting services provided. It derives that the experiences gained with the multi-media conferencing technology will be exploited for other workshops beyond the EPIC events, since the idea of EPIC to have distributed international events one can attend locally is very appealing and is in principle applicable for many other workshops
- Inclusion of video-conferencing service as part of general interest services, using the experiences made with the multi-media conferencing technology for new research in the field of multi-media and Internet systems. For instance a city-information system could combine various local information services among which a conferencing service allowing people to attend e.g. to public city hall sessions or art events etc.
- Usage of EPIC approach for reviews when software development takes place on more than one location (or even country)

## Acknowledgements

We have to thank all the people that directly contributed to the success of the EPIC Project, among which: A. Bollin, R. Bos, F. Van de Laar, G. Deler, J. Hrastnik, A. M. Krebs, R. Kuster, P. Lucas, J. Maniera, A. Mauro, R. Messnarz, A. Ollé, N. Robusté, W. Schaefer, R. Urban, F. Visentin, P. Watzke, A. Zuendorf.

We have to thank the European Commission for the financial support given to the EPIC Project (Number 23659), run under the ESSI Initiative as part of ESPRIT Framework IV Programme. We are especially indebted to the Project Officer, Mrs. M. Rohen, for her continuing support.

## References

- [1] ONION, *The EPIC World Wide Web*, <http://epic.onion.it/>
- [2] European Software Institute, *Software Best Practice Questionnaire*, ESI Technical Report, November 1996 (<http://www.esi.es/Publications/>)

- [3] G. Bazzana, E. Fagnoni, M. Piotti, G. Rumi, F. Visentin, *Testing in the Internet*, EuroStar 96 – Proceedings of 4<sup>th</sup> European Conference on Software Testing Analysis and Review, Amsterdam - December 1996
- [4] White Pine, *CU-Seeme*, <http://www.cuseeme.com/>

# A European Software Good Practice Repository

Giuseppe Satriani  
European Software Institute, Bilbao-Spain

*a software good practice is a software development, making best use of currently available methods and technologies in the most appropriate way, according to business needs. ESI considers that a best practice could be generated from a process improvement experiment defined in a particular company and applied in at least one concrete project that affects some company process. The main goal of the Good Practice Repository is to provide the necessary infrastructure for disseminating validated good practices and for providing services to facilitate their adoption.*

## Introduction

The European Software Institute (ESI) is a non-profit industry initiative with the aim of maturing, disseminating and promoting Software Best Practice. Taking advantage from the ESI method and vendors independence and from the amount of information available inside, the rationale is to create an integrated (similar Data Bases structure, searches and user interface) and consistent collection of Data Bases, called Good Practice Repository. ESI has already developed a first release of the Technology Shelf (T-Shelf) repository and is the prime contractor of VASIE 2 (Value Added Software Information for Europe), a project for the dissemination (through world-wide web) of ESSI (European System and Software Initiative) project results.

These two repositories are considered by ESI as the first building block for a European Good Practices Repository; the aim is to build the kernel for a European exchange of software improvement experience involving both users and suppliers.

## The European Good practice “Laboratory”

(<http://www.esi.es/Repositories/welcome.html>)

In 1993 the European Commission launched the European Systems and Software Initiative, ESSI. This initiative aims at improving the software development capability of European enterprises in all sectors, with special emphasis on quality. ESSI projects support transfer of technology from innovators and leading edge companies to other ‘early adopters’, and then to the early majority through a range of actions from assessments to process improvement experiments, and from dissemination to training.

This European software good practice “laboratory” is open to any approach an organization might want to chose for process improvement; it is independent of method, technique, tool, consultancy and training support, provided it can be established that defined business objectives will be supported.

Business improvement is the ultimate goal of acquiring good practices to improve software processes. Neither is this program solely directed towards software producing organizations.

Based on the fact that more than 70% of all software is developed in the non-IT sector any organization from any industrial of service domain may experiment in this laboratory. Hundreds of such projects have been or are running, and regular tenders for new, focused, actions ensure that experience gathered will be built upon.

The results of these projects have clearly shown that a wide range of benefits can be expected by the participating organizations directly through adoption of software “good practices”: software development, making best use of currently available methods and technologies, in the most appropriate way according to business needs.

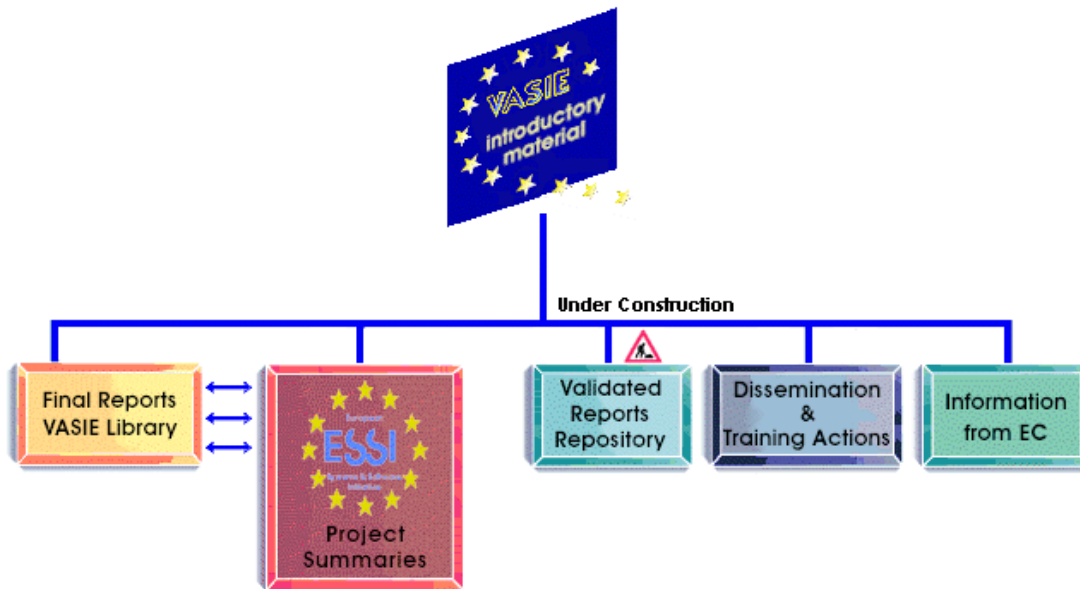
The benefits of an improvement action, varies depending on size of organization, on industrial sector and on application: airlines, banks, software houses, manufacturing companies - large and small. Improvements may address various parts and aspects of the software processes in the participating organizations, and these improvements are essentially driven by ‘user needs’.

### Towards a good practice repository: the baseline components

The most important aspect of such an open improvement “laboratory” is, however, that experience gained can be exchanged so that many more parties can benefit, and not only the organization performing the actions. To be possible on a continued basis a repository of experiences gained is necessary as a first step towards a good practice exchange.

#### VASIE (<http://www.esi.es/VASIE/>)

The European Software Institute (ESI) is a non-profit industry initiative with the aim of maturing, disseminating and promoting software good practice. In close collaboration with the European Commission, several project partners, and its own industrial members, ESI has established the kernel of a repository for results from the ESSI projects. This repository, and the corresponding dissemination project, is called VASIE (Value Added Software Information for Europe). All ESSI improvement projects are reviewed, categorized and added continuously to a world-wide web based information structure to guarantee widest possible access. The picture in the next page shows the way the information is structured on the Web.



In its current expansion phase VASIE will be enhanced concerning its technical infrastructure, but, more importantly, to enable “visitors” of the repository and potential adopters of such practices to interact with fellow improvement practitioners. The aim is to build the kernel for a European exchange of software improvement experience involving both users and suppliers.

Access to the information of the VASIE good practice library will be complemented to allow interactive experience exchange during and after a software process improvement experiment. Users

will be able to join fora related to specific projects, or specific issues, and will be able to query further details, share their own views and experiences, and receive comments from both other users and SPI experts. These fora are expected to develop discussions along certain lines or 'themes', such as, for example, the introduction of reuse as part of process improvement. These themes will be summarized at regular intervals - and again be added to the library.

Information destined for the library will be reviewed and edited by an international team of experts, appointed in agreement with the European Commission, based on their qualification and experience.

The partners of the current VASIE project are:

Etnoteam, Italy

MTA SZTAKI, Hungary

SIM-HSG, Switzerland

ZGDV, Germany

In particular, MTA-SZTAKI is responsible for fostering the VASIE repository in Hungary and in the East of Europe, by preparing case-studies on process improvement experiment conducted, by managing the mirror, by co-ordinating improvement experiment, by becoming a reference point of good practices for the east Europe.

### **TECHNOLOGY SHELF (<http://www.esi.es/Projects/Tshelf/>)**

As a step in this direction ESI has developed an information service called 'technology shelf', which collects information about tools, methods, training courses, events, consulting services and other practical information in the software process improvement domain. This service is already operating with information from a wide variety of suppliers. In the next step the information of VASIE and that of the technology shelf will be linked together, thus providing business evidence on service offerings through links to case studies; and information on how to proceed and whom to work for readers of the case studies.

There are 4 options to search in the Technology Shelf:

- search by basic process and service/tool  
*When searching by basic process, it is possible to select one basic process and one type of service or tool (training, consultancy, assessment service, audit service, tool or other). The result is a list of links to the service/tool pages of the selected type and related to the basic process given as input.*
- search by core competency  
When searching by core competency, it is possible to select one core competency. The result is a list of links to the company pages whose core competency is the selected one.
- search by assessment method  
When searching by assessment method, it is possible to select one assessment method. The result is a list of links to the company pages which provide the selected assessment service.
- search by audit method  
When searching by audit method, it is possible to select one audit method. The result is a list of links to the company pages which provide the selected audit service.

## **Conclusions**

Collaboration is, of course, open to any organization interested in sharing software good practices and experience in their application. Expressions of interest are welcome from individuals, projects and organizations. VASIE is also looking for qualified and experienced persons to join its growing international editorial and review board.

There are a lot of Good Practices repository projects, actions for the dissemination of Good Practices, training initiatives on Good Practices all around Europe, etc. However there is no coordination action for making all this initiative more effective and user oriented, and for trying to uniform the way the information is presented and agree on searching methods. An harmonization project could be an important key factor to structure the collection and dissemination of Good Practices.

For further information contact Giuseppe.Satriani@esi.es.

## References

- [1] Ruhe G., How to organise a repository of best practices in software engineering: a study for the European Software Institute, June 1994
- [2] Tully C., Improving total software performance in the context of total business performance, in: Proceedings of ISCN '96, pp. 223-232, Brighton, Metropole December 1996
- [3] Basili V.R., The experience factory and its relationship to other improvement paradigms, in: Proceedings of the 4th European Software Engineering Conference (ESEC) '93, Germany, September 1993
- [4] Basili V.R, Selby R.W, Hutchens D.H., Experimentation in software engineering, in: Transaction on Software Engineering, vol. SE-12, No. 7, July 1986
- [5] Mylopoulos J., Rose T., Software Repositories: Tutorial Notes, in: Fifteen International Conference on Software Engineering, 1993
- [6] Henderson J. Venkatraman N., Architecture of virtual organizing, in: paper on "Avoiding the Hollow: the building block of virtual organizing", Boston 1996
- [7] Hosking DM., Moreley I., A social psychology of organizing: people, processes and contexts, New York 1991.
- [8] Snow c., Miles R., Coleman H, Managing 21<sup>st</sup> century network organizations, in: Organizational Dynamics, Vol.20, Nr.3, 1992, S.5-20

## Appendix 1

### **European Software Institute (ESI) - <http://www.esi.es>**

(Tel. ++34-4-4209519 Fax ++34-4-4209420 e-mail:info@esi.es)

ESI is a major industrial initiative focusing on the improvement, dissemination and usage of processes critical for software intensive systems development, procurement, quality and maintenance to make these processes predictable in terms of cost and time to market

ESI is a non profit, industry led, membership-based organisation. It has support from public institutions, independent of commercial interests. ESI co-operates with key R&D institutes in this field and relies on regional partnerships for the exploitation of results and products for all proximity services that cannot be efficiently brought to the market by electronic means.

At the same time ESI has established institutional links with applied research organisations such as the Software Engineering Institute (SEI), the Applied Software Engineering Centre (ASEC), the European Software Process Improvement Foundation (ESPI), the Fraunhofer Institute for Experimental Software Engineering (IESE).

ESI best serves the European industry by pooling its own resources with those of its members and collaborative partners. Due to its neutrality and independence as well as to its European nature, ESI is also eligible to access the results of the European Commission programs in line with confidentiality rules, in order to enhance their use and make them widely available.



## Appendix 2

### ***Giuseppe Satriani's CV***

Giuseppe Satriani is presently a Project Leader at the European Software Institute. His main responsibilities are the definition and the implementation of the Best Practice Repository and the management of the EC initiative VASIE2. He is also involved in the Staged Model project that consider business goal as drivers for improvement programs.

He is SPICE qualified assessor, SEI Qualified Instructor for the CMM and a VASIE projects reviewer.

Giuseppe Satriani was previously an Engineer at the System-Test department at Olivetti SpA, afterwards he was Project Leader at the Quality department with the responsibility of quality system definition and application within the development department. Afterwards, he has been responsible for marketing activities (product promotion and suppliers management) at the Add-On Business department.

Giuseppe Satriani received his BS degree in computer science from Pisa University.

# NQA - Network based Quality Assurance system

Dr Richard Messnarz  
*ISCN Ltd., Dublin, Ireland*

Dr Robert Stubenrauch  
JOANNEUM RESEARCH, Graz, Austria  
The NQA Site  
<http://www.iscn.ie/projects/nqa/>

## **Introduction**

To stay competitive on the global market it is necessary to set up win-win based agreements in cost sharing projects in which partners from different countries share the risk and the effort and jointly exploit ideas, products, and services. Through effective distributed collaborations organisations can cut down the risk significantly (e.g. sharing the development cost with, for instance, 7 other partners) and can reach a much larger market (e.g. selling the product then in 7 regions of Europe).

However, the key problem is that distributed collaboration needs effective co-ordination of the work of the different partners. And old conservative means (direct supervision, local meetings, local and not distributed teams, etc.) no more work, unless they are supported by new and effective communication systems such as Internet based project archives, Internet based guidelines for project documentation, a virtual office on the net, as well as video workshop facilities for on-line team meetings.

While video workshop facilities are tried out in the ESSI project EPIC, the NQA system development focused on the virtual office on the Internet through which different partners can share a project archive, follow documentation guidelines, have workflow support in document flows, and have server functions for up- and download, including access rights control for security.

NQA is positioned as a system which facilitates the co-operation of a distributed development team within a software development project supporting 4 major work scenarios, such as planning, design, acceptance test, and maintenance. It can be as well used for EU supported projects (usually performed by consortia) where each work package is treated as a sub-project of the overall one.

Originally the NQA system came from a co-operation idea with partners from CEE who wanted to have a system in place for establishing an outsourcing support system which allows effective documentation and subcontracting to the East supported by an up-to-date communication solution for quality documentation.

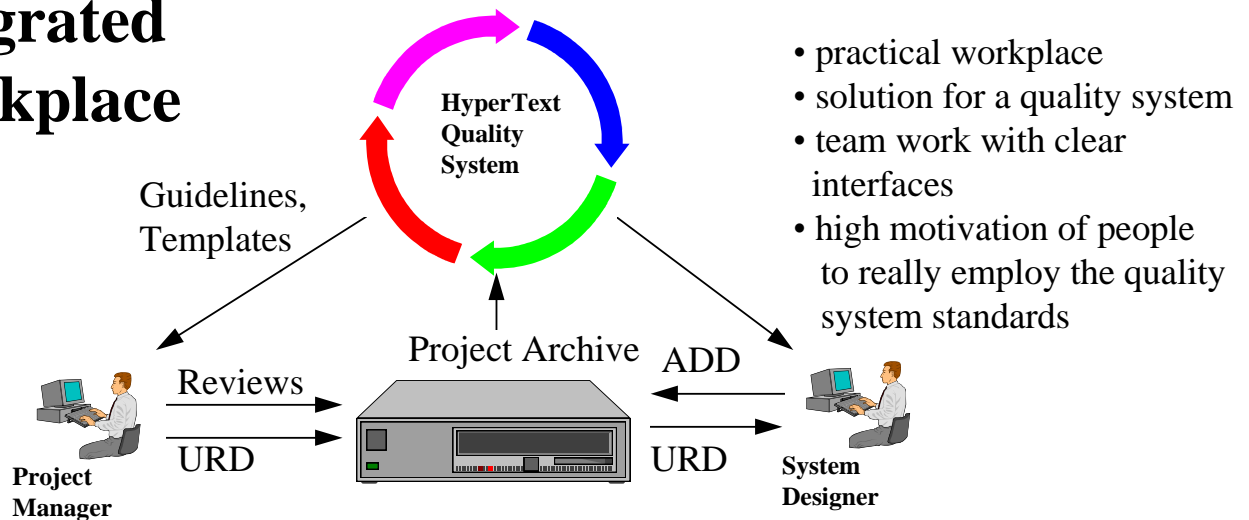
In the East there is an incredibly large resource (low cost, and high reliability if using the right procedures) available which by proper use can make the European software industry competitive against aggressive market strategies of the US and Japan.

## NQA's General Concept

The NQA system provides the following functions through a central Intra- or Internet server:

- an on-line quality manual
- work scenarios for planning, design, acceptance test, maintenance including
  - role plays and work instructions
  - document flows
  - activity flows
- a set of templates for planning, quality control, design, acceptance test, and maintenance
- a computer supported (CGI scripts in the current Beta-Release, later it is based on the Hyperwave server system) project administration and workflow facility with
  - project creation
  - selection of templates for use and computer supported transfer into the right place in the project archive according to the document type e.g. a plan is put into the planning cluster within the project archive
  - computer supported test status of documents (draft, reviewed, approved)
  - automatic distribution of documents to the distribution list by email
  - File Up- and Download utilities for transfer of documents into the right place in the project archive according to the document type
  - automatic contents list generation for documents
- a search index to find a term by alphabetical order and come to the right information and template

## Integrated Workplace



measured by ISCN: motivation increase from 17% to 57%, ISO Certification in 5 months

**Figure 5 :** NQA's General Concept

The acronyms in Figure 1 stand for

- URD ... User Requirements Document
- ADD ... Architectural Design Document

In addition to the HTTPD software solution there are other basic components:

<i>Component 1</i>	<i>Component 2</i>	<i>Component 3</i>	<i>Component 4</i>
NQA Paper Manual	NQA Winword Macros and Templates	NQA On-Line Manual and Templates	NQA HTTPD Solution
<ul style="list-style-type: none"> <li>• Role-plays</li> <li>• Work</li> </ul>	<ul style="list-style-type: none"> <li>• WinWord templates</li> </ul>	<ul style="list-style-type: none"> <li>• HTML Manual</li> <li>• HTML</li> </ul>	<ul style="list-style-type: none"> <li>• HTTP server support:</li> </ul>

<i>Component 1</i>	<i>Component 2</i>	<i>Component 3</i>	<i>Component 4</i>
NQA Paper Manual	NQA Winword Macros and Templates	NQA On-Line Manual and Templates	NQA HTTPD Solution
instructions • Document flows • Project administration • Quality evaluation • ISO 9001 mapping	English and German • Industry examples • Macro supported: Template menu Project Administration • works with: Windows NT Windows 95 Intranet	Templates English and German • Industry examples • Template Menu • Road map for readers	• Project administration • Online Up/Download • Document flows • Document status • Generation of index • includes Component 3

**Figure 6 :** NQA System Components

### ***NQA Solutions for SMEs***

These are small software organisations with size between 5 and 100 employees who plan to have an integrated workplace solution for Quality Assurance and Process Improvement using a joint experience and documentation base on a shared Windows 95 or Windows NT computer in an Intranet. Usually NQA for SMEs is used in a MicroSoft environment where NQA Macros organise a computer supported joint archive for all projects following documentation guidelines, and standard MicroSoft products such as Microsoft Visual SourceSafe for configuration management of the joint archive, and Microsoft Exchange for document flow management.

This includes the components 1,2,3 in the above Figure 2.

### ***NQA Solutions for Large Organisations or Consortia of SMEs***

The target group are medium and large software organisations who plan to have a distributed workplace solution for Quality Assurance and Process Improvement using a joint experience and documentation base on a Server running Unix on the Internet.

Usually NQA for medium and large organisations is used in a distributed environment where the NQA package is installed on a central server with a connection to the workplaces (usually via the Internet).

This includes the components 1,3,4 in the above Figure 2.

## **The NQA On-line Guidelines (and Printed Manual)**

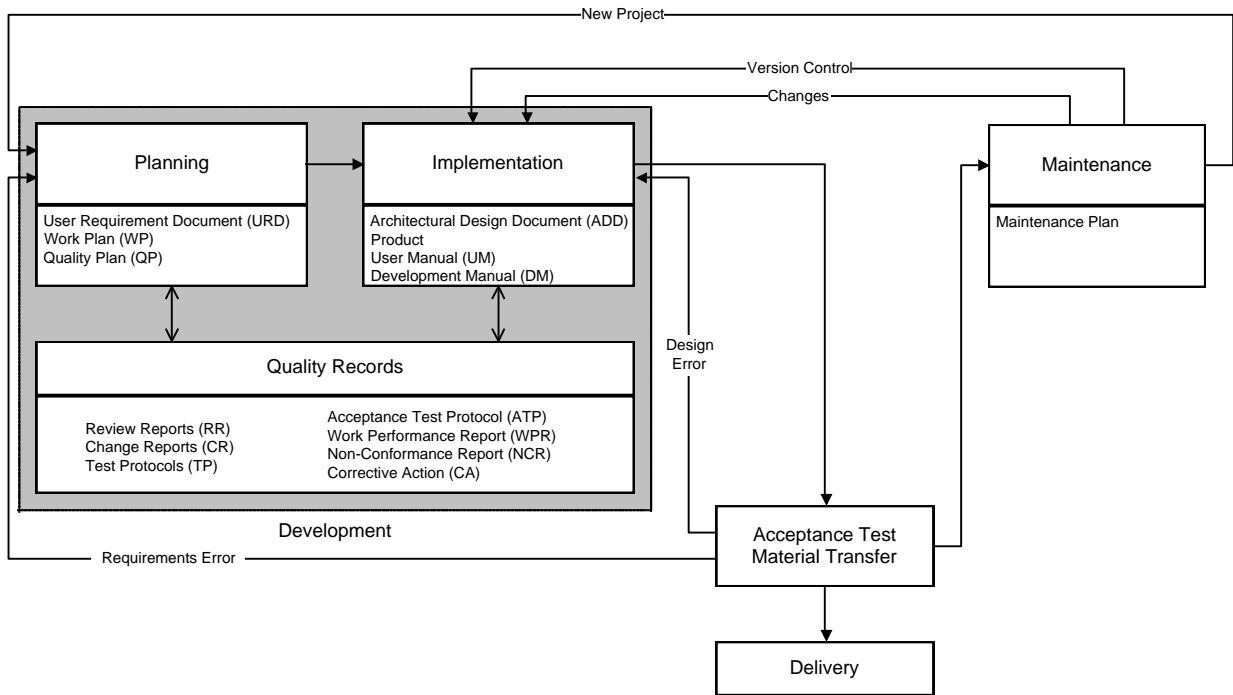
The NQA manual provides guidelines from three different viewpoints.

1. Roles, and communication between the roles: How to work in a team in a certain work scenario.
2. Work Flows : Sequence of work steps.
3. Document Overview: Which documents to be produced with which content.

For all documents in NQA the IEEE standards and the ESA PSS 05 software engineering standard have been taken into account. In addition those guidelines have been cut down into a lean documentation approach which was tried out at 3 leading companies. These use cases are described in a later section of this paper.

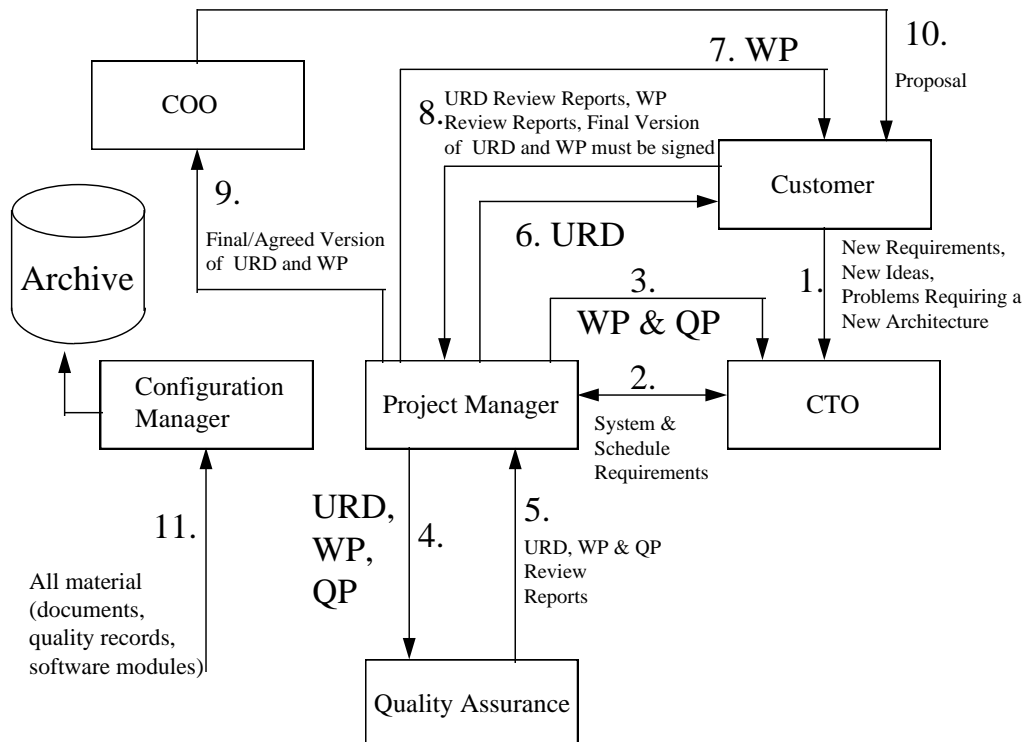
The below figure shows the work scenarios proposed by NQA for a software development team. .

## Global View Software Development



**Figure 7 : Development Related Work Scenarios Proposed by NQA**

Behind each work scenario there is a set of roles to be played by the team members. Each role is described by responsibilities and by its interfaces to other roles (played by team members). This way the initiation of a project becomes the assignment of people to defined roles and interactions between the roles. The below figure shows the proposed role play for the planning scenario.



### **Figure 8: Role Play for the Planning Scenario Proposed by NQA**

The CTO (Chief Technical Officer, in some companies the CTO responsibilities are fulfilled by the project managers) manages the customer contacts and derives information about new requirements, problems, and the need for new architectures. He formulates this information within a draft requirements document and instantiates a project forwarding the needs, requirements, and problems to the project manager.

The project manager refines the User Requirements Document (URD) and establishes a Work Plan (WP). Both documents are reviewed first by the quality assurance ( a status reviewed is achieved) and are secondly reviewed and signed by the customer (a status accepted is achieved). To ensure proper quality of the services and the product the project manager also establishes a Quality Plan (QP) which is reviewed by both, the quality assurance as well as by the CTO.

The agreed work plan with cost estimates and the requirements for the basis for the COO (Chief Operating Officer, Business Executive) to write a proposal and agree on a contract with clear goals, responsibilities, and cost parameters.

The configuration manager is responsible for archiving all versions of the documents and sources and quality records (e.g. review reports from the WP, QP, and URD reviews)

Work Instructions for the Planning Scenario (see numbers of communication flows in the above Figure 4)

1. The CTO receives new requirements, new ideas, and problems requiring a new architecture (thus the creation of a new project is needed).
2. The CTO forwards these system and schedule requirements to a responsible Project Manager either in form of a draft User Requirements Document (URD) or in form of a Review Report (RR) done together with the Customer.
3. The Project Manager refines the user requirements and establishes a draft Work Plan (WP) and Quality Plan (QP) which both are reviewed by the CTO.
4. The project manager provides the Quality Assurance (QA) with the draft URD, WP, and QP.
5. The Quality Assurance (QA) reviews the URD, WP and QP and documents all findings in Review Reports (RR). The Project Manager refines the URD, WP and QP according to QA's Review Reports and achieves a test status Reviewed for the planning documents.
6. The Project Manager forwards the URD with test status Reviewed to the Customer.
7. The Project Manager forwards the WP with status Reviewed to the Customer
8. The Customer reviews the URD and WP and documents findings in Review Reports which are sent back to the Project Manager. Based on this feedback the Project Manager refines the URD and WP until acceptance by the Customer is achieved.
9. The Project Manager provides the URD and WP which have been accepted by the Customer to the COO who then makes a review of these agreed versions. He checks the project from the cost and contractual point of view and can initiate step 2. again.
10. If the COO accepts the URD and WP, all planning documents achieve a test status Approved and the COO is writing a proposal (as basis for a contract) to the Customer.
11. The Configuration Manager (CM) is responsible for creating a project archive under a project identifier PJYY\_### (e.g. PJ97\_003, in "how to start a project" in the manual) with directories for planning documents, development documents, quality records, and maintenance documents, and to archive all versions of all documents.

Beside this role play, there is an activity and a document overview for the planning phase, with links to the document descriptions and templates to be used.

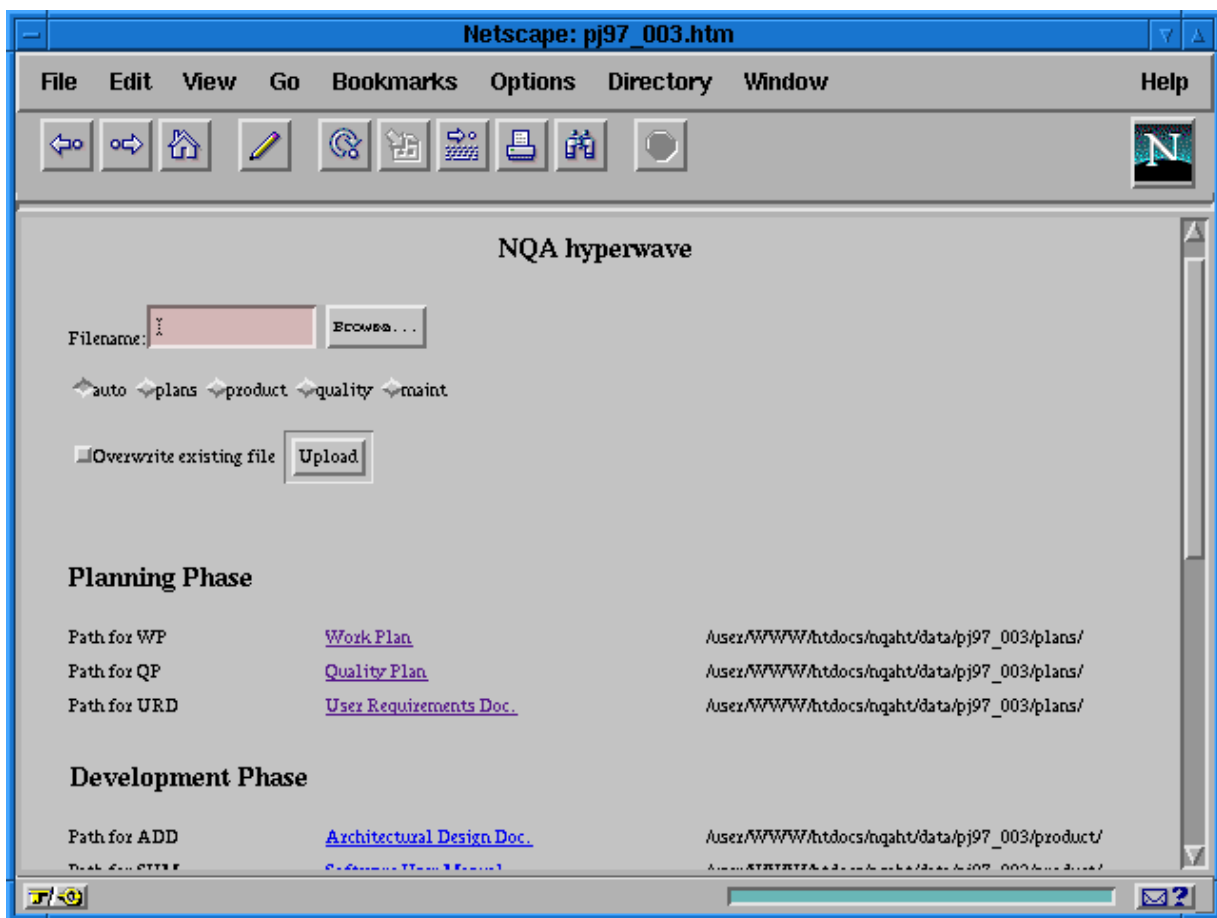
The manual and Word Templates also include administrative support (based o ISO 9001) for

- project administration
- contract management with standardised forms for contracts, agreements, and proposals
- etc.

## The NQA Virtual Office

By just using Netscape the team members (from home, from any work place, etc.) can access the NQA home page. From the home page a project administration function can be started offering the creation (or deletion or find) of a new project with a unique number and an acronym. All projects are kept in an administration table. In this administration table a link to the specific project administration sheet can be chosen and the user comes to a project specific sheet which contains:

- a link to the planning related documents
- a link to the design related documents
- a link to the quality related documents
- a link to the maintenance related documents
- a file up- and download utility
- file up- and download looks at the document type (stored within the document and set by a template) and stores the document automatically on the right place in the project archive



**Figure 9: Sample Picture of a Project Specific Administration Sheet**

The software support includes

- a menu of document templates from which a template can be selected and opened.
- A Save-For-Edit Function which by selecting a certain projects stores the template for editing into the right place in the project archive (a planning related document automatically under plans). This is done by using type-variables within the templates.
- Selecting a document in the project archive and download it for editing purposes. An upload of the document into the archive storing the template into the right place in the project archive (a planning

related document automatically under plans). This is based on type-variables which come from the originally selected template.

- A workflow function. This is done with an email distribution list per document and with a SUBMIT an email is automatically created which sends the document's http address , information about the document type and content to all addresses in the distribution list.
- A document status function which allows to set a status of draft, reviewed, and approved.
- An automatic contents list generation of produced documents in the archive.

Everything bases on standard HTML and Netscape. It also works on Explorer but the file up- and download function only works for the newest Explorer version.

## **Experience with NQA**

NQA is a brand new system. It has been finished in a complete version at the beginning of this year and the httpd solution as a Beta release has run through an acceptance test in August 1997.

So far NQA was acquired and field tested in 3 SMEs one at the size of 300, and two at a size of approx. 50. Based on this experience (from end of 1996 to July 1997) the NQA system has been refined to become more user friendly and stable.

In general NQA was used to facilitate an ISO 9001 certification. Therefore there is a mapping between ISO 9001 attributes and documents and work steps in NQA. However, NQA was not used as an ISO 9001 quality system in general but as part of an ISO 9001 quality system for the software unit within a company.

What were the major advantages

- engineers usually are confronted with a manual and much paper work when an ISO 9001 system is installed. NQA, however, is a work place solution where the quality assurance procedures form part of the normal work place and people do not feel that it is too much additional paper. In one of the above referenced SMEs they already had an ISO 9001 certificate but had problems in the software department, only 17% of the people were using it. After NQA above 60% of the people started to actually use the procedures.
- NQA document templates contain industry examples. Especially the requirements document structure was helpful in making the planning more complete. There was a worst-case project with no requirements and a customer asking for more and more within the same budget, which helped to convince people that a complete requirements documentation and planning is needed. In fact the new project with NQA had clearly defined requirements and plans and for deviations it was clear why they were created (actually in this case by the customer himself not delivering some basic data in time).
- In general software developers like electronic solutions and do not like to read books and large manuals So they certainly prefer to have on-line access. Especially the Search Index helps people to easily find the proper information by selection of a key word in the index.

What were the major problems

- So far NQA does not contain a configuration management tool so that the project administration puts documents automatically in the right place but for version control an additional system (beside NQA) is needed. This will be solved by the Hyperwave platform (see later sections about Hyperwave).
- The Word Solution largely bases on existing Microsoft products which are sometimes regarded as not stable, especially with the change from Word 7 to 8 and the data compatibility problem.



## NQA Future Development on Hyperwave

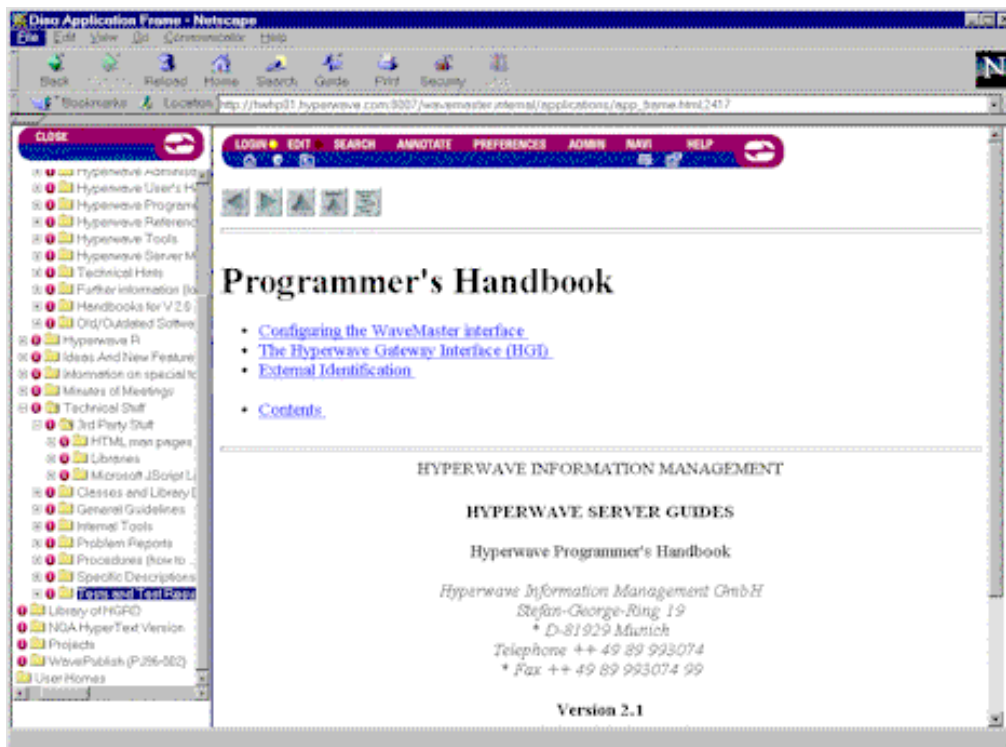
Currently an NQA customisation is being developed based on the Hyperwave Information Server system which will result in a range of additional benefits for NQA users. The Hyperwave NQA solution will fully exploit the advantages offered by this Web compatible document management system which is targeted at large and complex Intranet applications.

### *The Hyperwave Server System*

In a nutshell, as its main advantages Hyperwave

- cuts administration costs dramatically
- reduces complexity by avoiding structural links
- guarantees easy location of any information on large sites
- offers navigation through personalised, structured views
- provides collaborative and remote authoring.

For detailed technical and commercial information on Hyperwave see <http://www.hyperwave.com>.



**Figure 10 : Hyperwave is an on-line information system and powerful database**

### *Future View on an NQA on a Hyperwave Platform*

Hyperwave has been chosen as a platform for NQA because in this particular context it offers features that make it a very attractive platform. On a Hyperwave basis an NQA system is seamlessly integrated in a general document management system of arbitrary size. Thus a comprehensive internal archive can be comfortably accessed during the process of generating and maintaining the project documentation. The archive and the NQA documents can be maintained through the same user interface where the availability of features depends on the access rights of the user. User groups in Hyperwave NQA can be structured hierarchically with inheriting the corresponding rights.

Hyperwave offers an inherent hierarchical information structure (a kind of folder called „collections“) that allows for dynamic generation of structural navigation features such as going „up“ and „down“ the folder hierarchy, automatic sorting and sequencing etc.

In the context of quality management the following Hyperwave features are particularly useful:

- **Personal collections:** Folders to which any document that the user may access can be copied;
- **Copying documents:** Documents from the archive can be logically or physically copied to any folder by means of „drag and drop“;
- **Search:** Fulltext, title and attribute search, restrictable to certain folders or „current folder“;
- **Open and expire date:** Documents automatically „appear“ or „disappear“ at specific dates;
- **Meta-data:** Documents can be enriched by arbitrary additional searchable information;
- **Annotation:** Folders and documents may be annotated in the form of text attachments to particular text selections, or as images or links;
- **Arbitrary document formats:** Aside from the common formats such as HTML and PDF, arbitrary other formats can be held by a Hyperwave system; most common file formats can be indexed by the search engine.

For NQA, above general features of Hyperwave have been customized to meet the particular needs of quality management. Additional facilities are provided by particular workflow capabilities. These modules basically provide the facilities of the non-Hyperwave NQA solution but integrate them seamlessly in the Hyperwave environment and draw on specific features of that server system.

## References

Maurer H., Necessary Ingredients of Integrated Network Based Learning Environments; *Proc. ED-MEDIA '97*, Calgary, AACE (1997), 709-716.

Maurer H., What We Want from WWW as Distributed Multimedia System; *Proc. VSMM'97*, Geneva. IEEE, 148-155.

Messnarz R., Mac an Airchinnigh M., Biro M., Tully C., ISCN - An International Software Collaborative Network, in: *Proceedings of the ISCN'96 Conference on Practical Improvement of Software Processes and Products in December 1996 in Brighton/London*, ISCN Ltd. Dublin, Ireland, 1996

Messnarz R., Kugler H.J., BOOTSTRAP and ISO 9000: From the Software Process to Software Quality, in: *Proceedings of the APSEC '94 Conference*, Comput. Soc. Press of the IEEE, Tokyo, Japan 1994

Messnarz R. Tully. C. (eds.), The PICO - Book : *Software Process Improvement for Business Benefit - Principles and Experience*, The PICO Consortium, October 1997, <http://www.iscn.ie/publications>

in publication

Maurer H., Using the WWW System Hyperwave as Basis for Sophisticated Networked Teaching and Learning; *CIT (Journal of Computing and Information Technology)* (special issue)

Maurer H., Dietinger T., How Modern WWW Systems Support Teaching and Training; *Proc. of ICCE'97*, Dec. 2-6, 1997, Kuching, Malaysia

# Contract Management with Euromethod

Alfred E. Helmerich

Forschungsinstitut für Angewandte Software-Technologie  
(FAST) e.V. Arabellastr. 17, D-81925 München, Germany  
ahe@fast.de

## Abstract

Acquiring an information system to meet new business needs is not a trivial task. It includes deriving the acquisition goal, developing a strategy for its implementation, contracting for parts of the acquisition goal, integrating the parts into the complete information system and into the business processes of the acquiring organisation.

Properly addressing these issues during acquisition significantly increases the likelihood of a successful outcome. Effective acquisition of an information system and related services requires clear descriptions of the desired final state and the current situation. It is important that the customer and supplier have the same understanding of the current situation and the information system and related services to be achieved.

Euromethod has been designed to help organisations with the acquisition of effective information systems and related services in a variety of situations. It encourages customers and suppliers to control costs and timescale, to manage risks, to improve mutual understanding and to reach a fair contract.

Through the achievement of these objectives, the European Commission aims to encourage the opening of the information system (IS) market, to improve the mobility of people internationally, to ease the organisation of international projects by a flexible contract management.

## Introduction

One of the principles of the European Union is the completion of a single market. Therefore the EC requires for their administrations that the call for tenders be open throughout Europe to allow competition to take place and refer to product descriptions based on standards rather than on brand names.

### ***Why should the tenders be open?***

There are good reasons for any customer to favour open tenders, even though they might require more work in the beginning and result in starting dates of projects to be shifted in time:

- competition will generally lead to more cost-effective solutions;
- competition increases the variety of solutions and generally will lead to better solutions;
- standards will prevent a lock-in to one supplier and secure the investment for future

- updates;
- better planning of the acquisition leads to better control and results.

However, not only the customers but also the suppliers profit from an increased market. Indeed many standards are rooted in initiatives by suppliers to make their products interchangeable and secure their investment in research.

### ***Why are the Council Directives not enough?***

The EC has regulated the public procurements of various types, e.g. the procurement of:

- construction work (Works Directive [1])
- IS-products (Supplies Directive [2]),
- IS-services, e.g. for those processes that represent an economic value not related to the production of material goods (Service Directive [3]),
- for IS-services in special branches, e.g. telecommunications (Utility Directive [4]).

As can be seen by the number of directives, the various types of procurements require special treatments. Whereas in the procurement of products standards can be defined and off-the-shelf products therefore exist, it is impossible to do likewise with defining services that entail creative work and problem solutions.

Moreover, information has become a cornerstone of modern organisations. And the procurement of an information system (IS), whether in the private or the public sector, be it paper-based, partially or fully computerised, often is a key to the success – and often the survival – of an organisation. Also the complexity of a computerised system may range from a single PC to a distributed heterogeneous system containing many complex and interacting applications.

Uncertainty will depend on the type of application, organisational aspects, technology to be used, etc. Applications that are common across organisations and are well understood present little risk. Novel and/or specific applications have the potential to provide competitive advantage but may be more risky.

However, neither standards nor the Council Directives alone can guarantee a good quality of the solution. A method - EUROMETHOD - is needed to install the directives in practice.

### Why should EUROMETHOD be applied?

#### The method "EUROMETHOD"

Euromethod was designed to support the definition, planning, and execution of the effective acquisition of information systems and related services.

It is used to assess and determine:

- the problem situation and the associated risks
- the goal of the acquisition
- the strategy for the acquisition, for the IS-adaptation and service provision
- the delivery plan showing the customer-supplier relationships at contractual level including the exchange of deliverables.

Euromethod does not address legal aspects. Neither is it an IS-development method.

#### The framework "EUROMETHOD"

Euromethod provides a framework i.e. a set of concepts and a terminology:



- to improve the customer-supplier relationship;
- to harmonise methods;
- to standardise the procedure;
- to provide standardised templates.

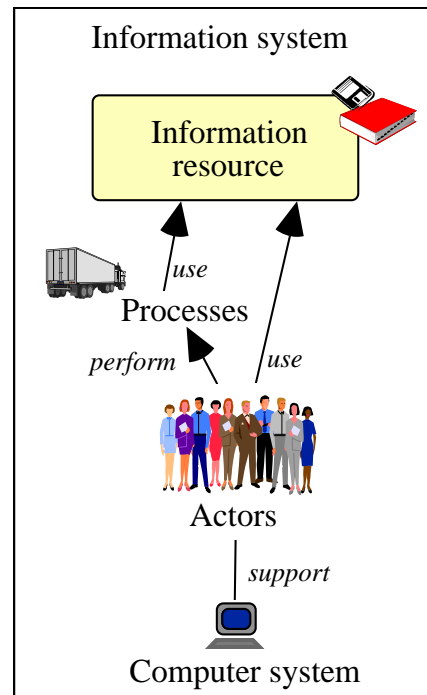
One of the main obstacles in achieving mutual understanding is the variety of methods using different concepts and terminology. These methods often use a vocabulary that stems from software engineering and may be difficult to understand by IS users, procurers and contract managers.

Euromethod addresses this problem by considering adaptations and service provisions from an acquisition point of view rather than an engineering point of view. A bridging dictionary enables people to understand the types of deliverables proposed by a method without having been trained in that method. Bridging dictionaries already exist between some methods (like SSADM, SPICE, Merise, MEiN, Dafne) and Euromethod.

A standardised procedure and templates reduce the efforts for the next tender and generally lead to more transparent and fairer contracts.

#### The project "EUROMETHOD"

- 1 In a first phase in early 1989 the European Union Member states agreed on the needs and requirements for a EUROMETHOD.
- 2 In a second phase from May 1990 - Feb. 1991 the CEC DG XIII/PPG funded a Feasibility



**Fig. EM.1:** Information system  
The aspect of the organisation that provides, uses and distributes that information, together with the associated organisational resources such as human and technical, is called an Information System (IS).

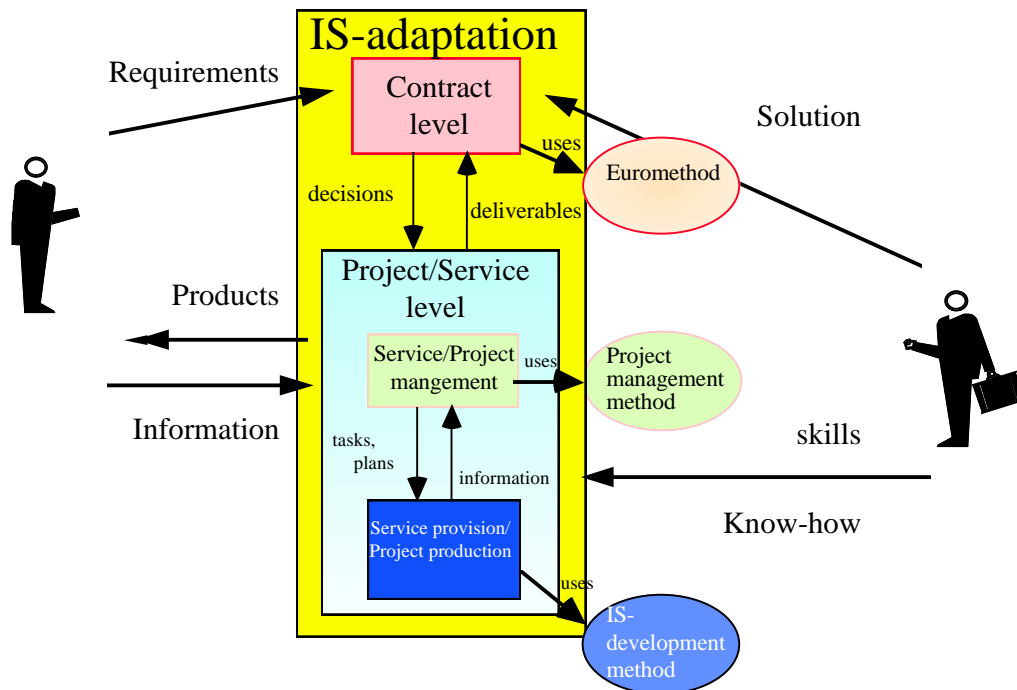
- Study performed by an pan-European consortium (Eurogroup).
- 3  From May 1992 - April 1994 Version 0 of EUROMETHOD was developed and put to trail in the next phase.
- 4  From July 1994 - July 1996 EUROMETHOD was applied within seven projects throughout Europe, including a PHARE project in Hungary (see Tab. EM.1)
- 5  The results were used to develop the current version 1 of EUROMETHOD.

<b>Name</b>	<b>Description</b>
<b>Albacom,</b> by Mark Gibbons BT	<b><i>Joint Venture between BT Italia and Banca Nazionale de Caroro</i></b> <ul style="list-style-type: none"> <li>• 80 employees</li> <li>• 3 business areas: Networks, Operational Support System, Billing</li> </ul> key factors: <ul style="list-style-type: none"> <li>• focus on delivery</li> <li>• show business benefits on a daily basis</li> <li>• situational factor analysis for each subsystem, but with jargon adapted to IS-adaptation (justify each value)</li> <li>• complex risk management 62 decision points</li> </ul> results <ul style="list-style-type: none"> <li>• Euromethod was well received as a structured approach and</li> <li>• a means for manager to ask questions and judge deliverables.</li> <li>• It increased the level of confidence of the overall project manager.</li> </ul>
<b>Bank Contract</b> by Victor Van Swede Cap Volmac	<b><i>How can Euromethod better the internal contract management of a bank?</i></b> result: <ul style="list-style-type: none"> <li>• deliverable profiles proofed that requirements were not fulfilled</li> <li>• Euromethod is systematic way of working</li> <li>• standard method products often provide unneeded inf. for decision</li> <li>• method bridging was not needed.</li> </ul>
<b>Oursource,</b> by Victor Van Swede Cap Volmac	<b><i>Government used Euromethod to outsource services!</i></b> result: <ul style="list-style-type: none"> <li>• for each service (help desk, configuration management..) a situational factors analysis had to be performed.</li> <li>• A strategy had been developed for each service (Sub-System)</li> <li>• New concepts like SERVICE - TASK - SERVICE_PROPERTIES had to be added using the quasi-standard ITIL from CCTA.</li> <li>• Initial states and final states had to be supplemented with SERVICE-Level.</li> <li>• Services are part of Euromethod Version 1.</li> </ul>
<b>ESAS II ,</b> by Niels Anderson Data Centralen	<b><i>Requirements study for the Government before a call for tender</i></b> The call for tender was to procure <ul style="list-style-type: none"> <li>• a system</li> <li>• maintenance service afterwards in one go.</li> </ul> To calculate the costs for maintenance was not possible, but one could agree on a costing structure and basis. Result: <ul style="list-style-type: none"> <li>• To procure a multi-step project the Danish procurement law had to be rewritten.</li> </ul>
<b>Logo,</b> by Marcel Franckson SEMA	<b><i>Ministry of Environment used Euromethod to select supplier</i></b> Result: <ul style="list-style-type: none"> <li>• Euromethod helped define the expectations of the Ministry</li> <li>• a model supplier section was developed as a case study</li> </ul>
<b>Semantic,</b> by Marcel Franckson SEMA	<b><i>Semantic Model of all French Government information</i></b> The situational factor analysis of Euromethode was employed to proof that the project should be aborted. Result: <ul style="list-style-type: none"> <li>• Euromethod preventing the Government from greater harm afterwards.</li> </ul>

Name	Description
<p><b>Hungary,</b> by Alfred Helmerich and Mark Gibbons</p>	<p><b><i>Modernisation of consular procedures</i></b></p> <ul style="list-style-type: none"> <li>• The technical annex for a call for tender was produced, consisting of a fully developed delivery plan.</li> <li>• Risk management was employed to reach at a suitable strategy and decision point sequence.</li> <li>• During the consultancy the initial, final state and the situational factors were dramatically improved.</li> </ul> <p>Result:</p> <ul style="list-style-type: none"> <li>• The use of Euromethod was heartily welcomed.</li> </ul>
<p><b>BayIM,</b> by Alfred Helmerich FAST e.V.</p>	<p><b><i>Procurement of X.400 for all Bavarian ministries</i></b></p> <p>Euromethod was used to simulate the procurement and to analyse pro and cons of using Euromethod.</p> <p>A number of important issues were raised:</p> <ul style="list-style-type: none"> <li>• abnormal contract closure needs to be refined</li> <li>• necessity of a procurement goal was recognised</li> <li>• configuration management , due to evolving Euromethod deliverables is needed.</li> </ul> <p>result:</p> <ul style="list-style-type: none"> <li>• the German Procurement Handbook became the skeleton of Euromethod Version 1</li> </ul>
<p><b>SNI-CCP,</b> by Alfred Helmerich FAST e.V.</p>	<p><b><i>Euromethod to define a project</i></b></p> <p>Euromethod was used very informally to plan an internal project of the Culture Change Program.</p> <p>Result:</p> <ul style="list-style-type: none"> <li>• It proofed extremely valuable to switch from the project view up to the contract view to define dates, strategies and the final state.</li> <li>• The focus on deliveries and decision points sped up discussions.</li> </ul> <p>The effect of Euromethod was monitored throughout the project.</p>

**Table EM.1:** Pilot applications of Euromethod Version 0

**How is the contract relationship defined within EUROMETHOD?**



**Fig. EM.2:** Structure of an acquisition organisation

One can distinguish three levels. The contract, management and production/provision level. Euromethod addresses the first level, whereas most IS-development methods address the second and/or third level.

The customer/supplier relationship takes place on three distinct levels.

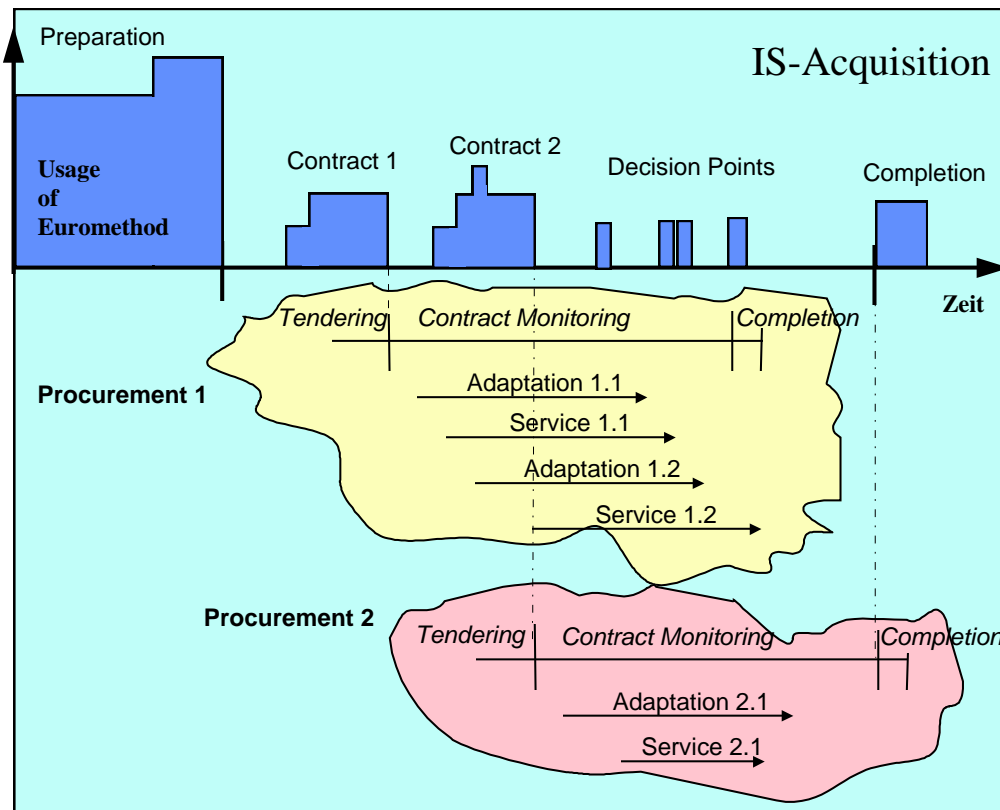
- The service provision or project production function provides the required service or systems for the customer, e.g. business process engineering, computer system operation, network maintenance, software development. For this purpose it uses resources (skills, knowledge, products, etc.) from the supplier and sometimes from the customer.
- The service or project management function plans and monitors the service or the production. It organises the team, allocates resources to the tasks, and makes sure that the required quality is achieved within timescale and budget.
- The acquisition management function controls the acquisition and its various contracts. It is responsible for the service and system requirements that are documented in the request for proposals, tender responses and contracts. It controls whether requirements are met by the services and system and takes the appropriate measures when they are not. In complex acquisitions, this level may be split into acquisition management and contract management and the various contracts may involve different people in their management.

**Making a contract**

Due to the complexity of information systems acquisitions are usually split into several procurements regulated by contracts allowing for smaller lots and increased competition. The offside is an increase in preparation and administration especially at the beginning.



A very common example of a stepwise acquisition is a pre-study performed internally or externally, before the actual procurement with a call for tender is launched.



**Fig. EM.3:** The acquisition process model

Complex acquisitions may involve more than one supplier, each one being responsible for a subset of systems and services. Suppliers in their turn may have sub-contractors providing them with some services and systems.

### ***The initiation of the acquisition***

The acquisition process (or acquisition for short) is the process of obtaining a system or a service, or any combination thereof. Its necessity usually arises from some business needs.

The acquisition goal is used to drive the acquisition process, which starts with the formulation of an acquisition strategy determining the number and the kinds of adaptations, service provisions, and contracts, that are needed to reach the acquisition goal.

The planning of the acquisition process on the other hand usually results in a further refinement of the acquisition goal, in terms of

- target domain affected;
- systems and services requirements;
- cost/benefit analysis;
- stakes and stake holders.

The acquisition planning will start by determining the overall adaptations and service provisions plan scenarios, then analysing the risks and designing an acquisition strategy within a risk management framework; setting up the acquisition organisation; and finally planning the main decision points of the acquisition:

- Decide to change some situational factors;
- Decide to change or refine requirements prior to tendering;
- Decide to use external assistance in the acquisition management;
- Decide the types of suppliers: internal or external;
- Determine the types of tendering (open, restricted, negotiated);
- Determine the interaction with suppliers (single-phase, multi-phase tendering);
- Determine the flexibility of contracts (capability to modify or refine);
- Decide the strategy regarding standards;
- Identify contracts and sequencing constraints (one or several contracts);
- Decide to buy or develop;
- Determine requirements to adaptation strategy;
- Determine the type of service arrangement;
- Determine requirements to service provision strategy.

Each step is called a procurement of the systems and services that are defined within its contract. It usually consists of a sequence of three processes:

- tendering process,
- contract monitoring and
- contract completion process.

### ***Mode of tendering***

The EC directives ask that all call for tenders with a value above a certain limit (GATT-limit) are officially and openly announced in the European Journal or electronically in Tenders Electronically Daily (TED; telnet: echo.lu). The directives regulate the structure and scope of the announcements; EUROMETHOD prescribes the structure and supports the preparation of the accompanying detailed technical information.

A standardised structure of the tender information and tender response already reduces the work load of both suppliers and customers. It is only natural that in EUROMETHOD the tender information as well as the tender response are already in the same format as the technical annex of the final contract.

According to the EC directives there are four modes of tendering described:

- Open call for tender  
The default for any procurement allowing unlimited participation of suppliers.
- Restricted call for tender  
In special cases the call for tender can be given to a short list of suppliers only. Generally this is admissible when the short listed suppliers effectively are all possible suppliers for the specific procurement. The above condition can be verified by a market study, a previous open call for tender or by a previous open call for application.
- Negotiated call for tender  
Allows the customer to make a contract with one supplier, if an open competition is proofed to be without success, not possible or not justified by the procurement. Mostly it is used to contract some additional work (less than 20% of the original contract) or if there is only one supplier.

- Open call for application

Describes a two phase tender process, where the customer first calls openly for suppliers to claim their interest. In a second step the call for tender is given to those suppliers only that have applied in the first step.

In England customers use the open call for application mode to generate a short list of possible suppliers. That is, they first call for mini-proposals, select admissible suppliers and then call the short listed suppliers for their full proposals.

It is conceivable that the customer uses the mini-proposals to generate options for solutions that are then discussed with all short listed suppliers on a round table prior to the call for a full proposal for one selected option.

## What contains a contract?

A contract is a binding agreement between two parties especially enforceable by law or a similar internal agreement wholly within an organisation, for the supply of services or systems. Several contracts may be required for the acquisition of the systems and services needed by an organisation.

It is the main goal of any contract to describe clearly the deliverables that are to be exchanged between customer and supplier. Deliverables can be products or services and are described

- by their goals, constraints and quality characteristics (e.g. deliver a certain product to a customer within a certain time and cost and to the customer's satisfaction);
- by their results (e.g. delivered product);
- by their activities (or sub-process) (e.g. the delivery process will consist in getting the product out of stock, checking its characteristics, selecting the transportation means).

## Description of Results

The description of results is easiest if the products are already standardised as described in the European Procurement Handbook of Open Systems (EPHOS).

If the results are information systems, they can be described in EUROMETHOD by the concepts of initial and final states.

## The levels of abstraction

In analogy of the different levels of customer/supplier relationship, products of different level of abstraction are passed on between the levels.

As Euromethod supports the contractual relationship, it only provides templates or profiles to characterise descriptive items. These profiles do not contain a summary of the content of the descriptive items, they rather classify the scope, the quality and functional properties of the descriptive item.

For that reason the profiles are very flexible and can be

- used to describe information at the contract abstraction level for decision making;
- used as acceptance criteria for

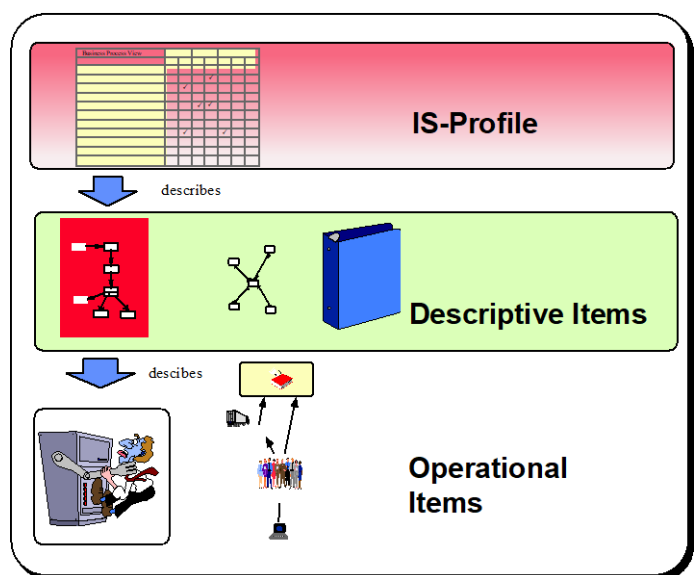


Fig. EM.4: Different levels of abstraction

controlling the contract;

- adapted to the situation by defining the granularity of the grid to suit its objective.

The latter can be used to describe the necessary deliverables for decision points. The common decisions are the selection of suppliers in the tendering phase, decisions about system design, future investments, and system acceptance in the contract monitoring and completion phases.

### The types of deliverables

When characterising descriptive items by profiles, Euromethod recognises three main types of deliverables, for which different default profiles are provided.

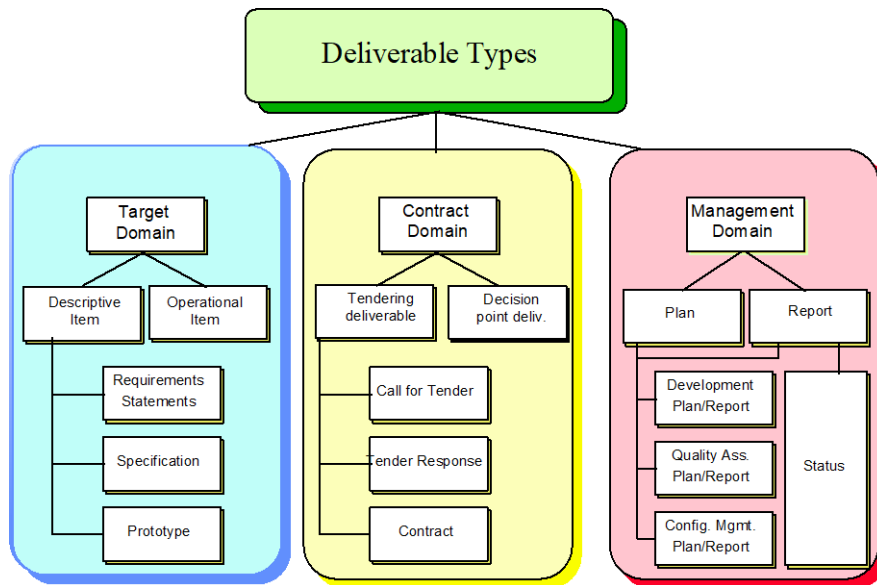
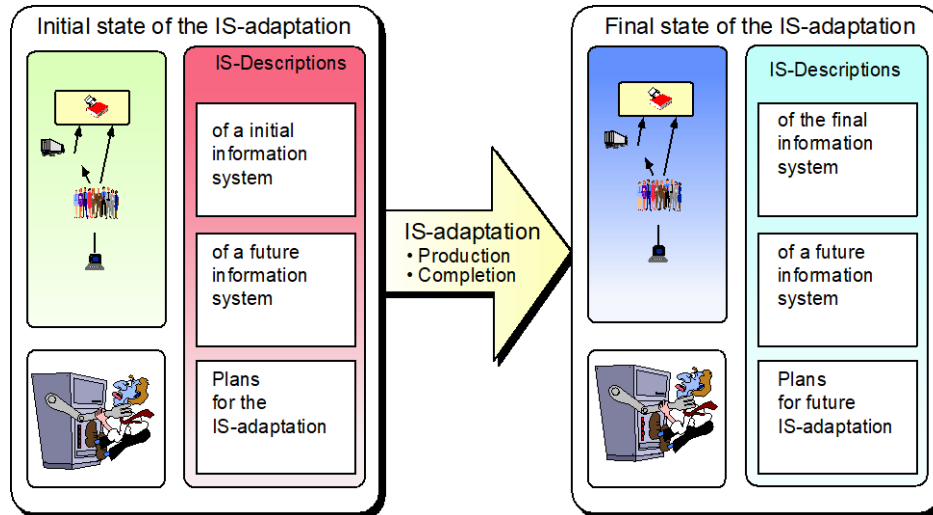


Fig. EM.5: Deliverable types in Euromethod

## Initial and Final States

The profiles of all deliverables available at the starting point of the IS-adaptation are called the initial state profile. Likewise is the set of profiles at the expected end point called the final state profile. The two are used to illustrate the transition the information system is meant to undergo during the IS-adaptation.



**Fig. EM.6:** Initial & Final State of an IS-adaptation

An IS-adaptation is defined by its initial and final state. System development methods help create documents that describe the IS (IS-descriptions). Euromethod in addition helps to create profiles that characterise and describe the IS-descriptions

## Description of Tasks

Task descriptions are used to steady state processes outsourced to some supplier supporting the day-to-day functioning of the organisation. They are usually continuous and they contain activities that are repeated regularly during the life of the organisation. They remain in the same steady state, or incur only slight changes, for long periods.

Task descriptions are also used to manage contracts and allow for the flexibility of contracts needed during the adaptation of an organisation to its changing environment. Each adaptation is a specific process that has a beginning and an end and that executes a state transition in the organisation, i.e. it moves the organisation from an initial state to a final state in a certain elapsed time. The adaptation process can be adjusted to the problem situation and monitored to guarantee success by the following activities:

- Risk Analysis;
- Strategy Selection;
- Decision Point Planning.

## Risk Analysis

Euromethod provides a list of situational factors that need to be analysed as to their potential to cause risks, e.g. their likelihood of happening and the severity of consequences.

For each situational factor, Euromethod proposes heuristics to diminish the inherent risk. Some actions are rather local and address one situational factor only, others are more global and affect the strategy selected for the IS-adaptation, like the splitting of the project into various steps or the evolutionary development.

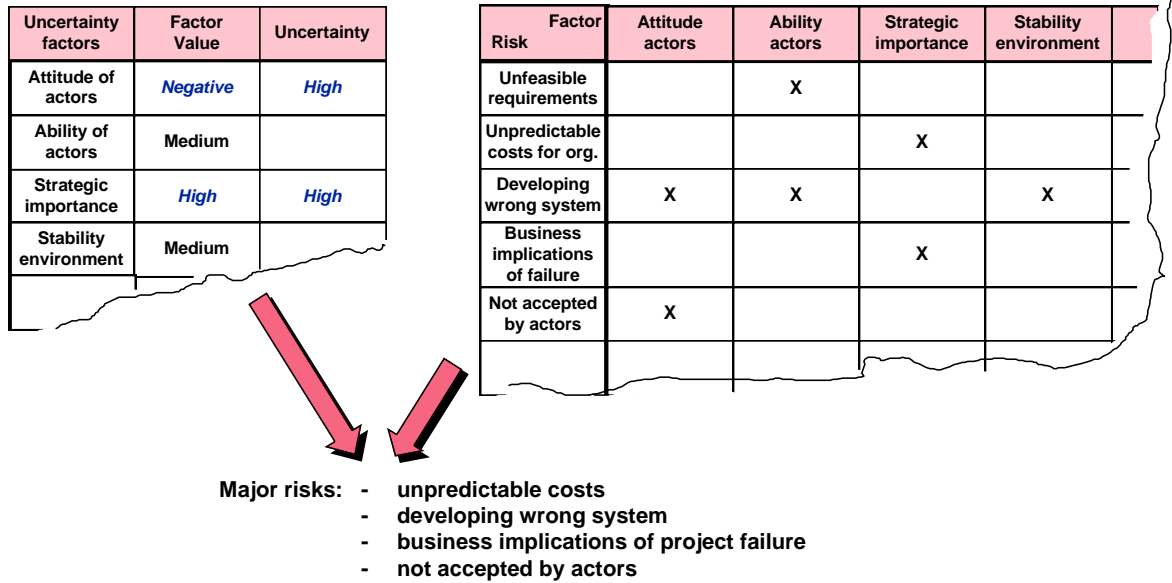


Fig. EM.7: Usage of the situational factor table in risk analysis

### Strategy Selection

The following table lists the strategy options among which one can choose in EUROMETHOD. The choice is determined by the situational factors as explained in the previous chapter. Risks that are not covered by the chosen strategy have to be specially treated and monitored by the project control.

Adaptation approach	Strategy options
Cognitive description approach	<ol style="list-style-type: none"> <li>1. Analytical: <ul style="list-style-type: none"> <li>• use of abstractions and specifications</li> </ul> </li> <li>2. Experimental: <ul style="list-style-type: none"> <li>• use of experiments and prototypes</li> </ul> </li> </ol>
Social Description approach	<ol style="list-style-type: none"> <li>1. Expert-driven: <ul style="list-style-type: none"> <li>• production and assessment separated</li> </ul> </li> <li>2. Participatory: <ul style="list-style-type: none"> <li>• joint production and assessment</li> </ul> </li> </ol>
Construction approach	<ol style="list-style-type: none"> <li>1. One shot construction: <ul style="list-style-type: none"> <li>• a single version constructed and tested in one step</li> </ul> </li> <li>2. Incremental construction: <ul style="list-style-type: none"> <li>• parts constructed and tested in a sequence of steps</li> <li>• no change of descriptions after first construction</li> </ul> </li> <li>3. Evolutionary construction: <ul style="list-style-type: none"> <li>• versions constructed and tested in a sequence of steps</li> <li>• changes of descriptions are possible after learning from test</li> </ul> </li> </ol>
Installation approach – system coverage	<ol style="list-style-type: none"> <li>1. One shot installation: <ul style="list-style-type: none"> <li>• a single version installed in one step</li> </ul> </li> <li>2. Incremental installation: <ul style="list-style-type: none"> <li>• parts installed in a sequence of steps</li> <li>• no change of descriptions after first installation</li> </ul> </li> <li>3. Evolutionary installation: <ul style="list-style-type: none"> <li>• versions installed in a sequence of steps</li> <li>• changes of descriptions are possible after learning from usage</li> </ul> </li> </ol>
Installation approach – geographical coverage	<ol style="list-style-type: none"> <li>1. Global installation <ul style="list-style-type: none"> <li>• installation in all locations in one step</li> </ul> </li> <li>2. Regional installation <ul style="list-style-type: none"> <li>• stepwise with more and more locations</li> </ul> </li> </ol>

**Table EM.2:** Strategy options for an IS-adaptation

### Decision Point Planing

A key element of flexible contracts are decision points. They allow the customer in co-operation with the supplier to make intelligent decisions based on the deliverables produced.

Although the outcome of a decision can not be planned, it is possible to plan the decision and to specify the necessary deliverables.

Actually, during the tendering process a refinement of that planning takes place and finally leads to the formulation of a delivery plan as a basis of the contract.

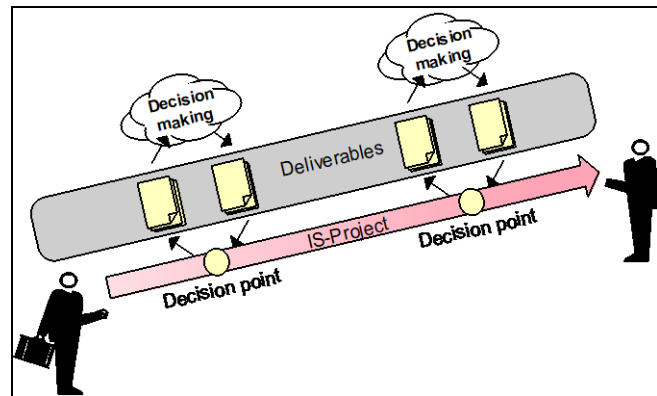


Fig. EM.8: Decision points

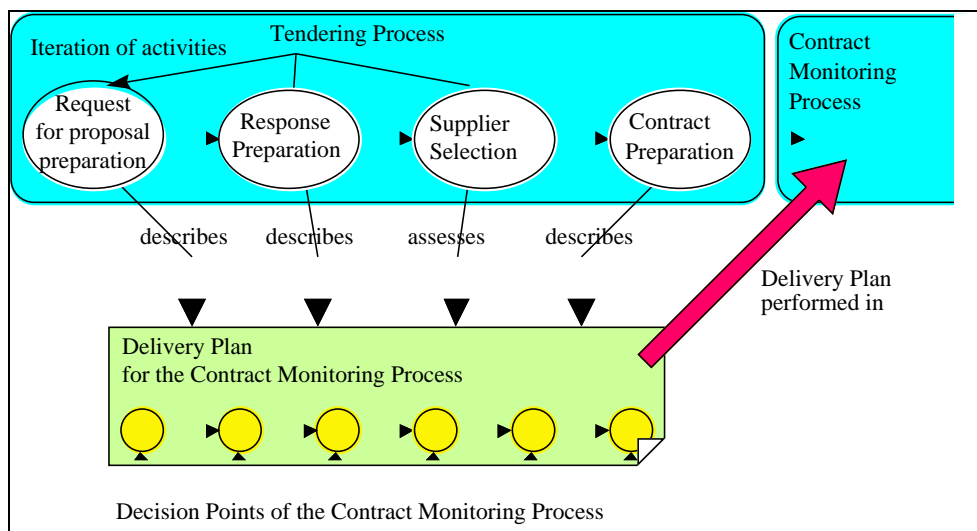


Fig. EM.9: The connection of delivery planning and contract monitoring

### Description of Goals

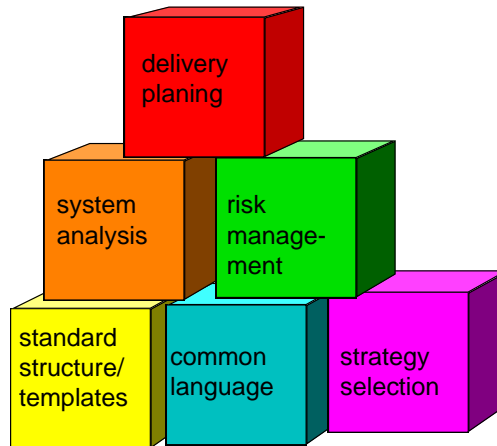
The goals of an acquisition are described in terms of a business strategy with market survey and estimates for costs and benefits. The acquisition goal is needed to co-ordinate all subsequent procurements and to guarantee the overall success.

The goals of a procurement are described by the final state of the IS-adaptation or the service level that has to be achieved.



## Conclusion

In Euromethod, a contract is not used as a legal means to pull the wool over the partner but as an instrument to come to a fair and clearly understood agreement that can be tailored to the problem situation and is flexible enough to adopt if needed.



**Fig. EM10:** Benefits of using Euromethod

<b>Benefits to customers</b>	<b>Benefits to suppliers</b>
Clearer expression of requirements	Better understanding of customer's needs
Improvement of risk management	A clearer view of the customer's IS
Guidance in choosing the appropriate acquisition approach for a specific problem situation	Easier to obtain a clear endorsement from customers of the key design decisions
Better understanding of suppliers' proposals	Determination of the appropriate approach for a project or service provision
Easier evaluation of suppliers' proposals	Selection of the appropriate methods, techniques and tools
Easier system and service acceptance, through better requirements definitions and planning	Easier system and services acceptance, through better requirements definitions and planning
Improved decision process relating to deliverables and services	Enhanced management of risks involved in a project and/or a service provision
Avoidance of lock-in to a supplier	Better information to control ambitions and costs
Avoidance of lock-in to a specific method	
Better information to control costs	
Easier control of ambitions	

## References:

- 1□ 93/37/EEC: Works Directive concerning the co-ordination of procedures for the award of public construction contract valued at 5.000.000 ECUs or more, in: *Works Directive*, 14.06.1993
- 2□ 92/50/EEC: Service Directive concerning the co-ordination of procedures for the award of public service contracts valued 200.000 ECUs or more, in: *Service Directive*, 18.06.1992
- 3□ 93/36/EEC: Supplies Directive co-ordinating procedures for the award of public supply contracts valued 200.000 ECUs or more. This is an amendment of 77/62/EEC, 80/767/EEC and 88/295/EEC, in: *Supplies Directive*, 14.06.1993
- 4□ 93/38/EEC: Utilities Directive concerning the co-ordination of procedures for the award of public service in the area of water, energy and transport (>400.000 ECUs) or telecommunications (>600.000 ECUs) in: *Utilities Directive*, 14.06.1993
- 5□ Eurogroup, *Euromethod Version 1*, 1996
- 6□ Eurogroup, *Euromethode V0*, 1994
- 7□ EPHOS: *European Procurement Handbook for Open Systems*, EC

## Annex

### About the Author



Dr. Alfred Helmerich

The author studied Physics and has developed an evaluation tool set. After his thesis he became a trainer for software developers within Siemens Nixdorf. In 1994 he was outsourced to the research institute FAST e.V. where he took over the German role in developing Euromethod.

### About the research institute of applied technology (FAST)

**F ▶ A ▶ S ▶ T**  
Forschungsinstitut für  
Angewandte Software-  
Technologie (FAST) e.V.  
Arabellastr. 17  
D-81925 München  
Germany  
Phone: ++49 (89) 920047-0  
Fax: ++49 (89) 920047-18  
E-Mail: info@fast.de

Proposed by the scientific-technological Advisory Board of the Bavarian Prime Minister, the Research Institute for Applied Software Technology (FAST) was founded in 1993 by a group of leading business enterprises. Among its sponsors are:

- the Bavarian Science Foundation,
- the Bavarian Ministry of Economics and Transport,
- the Bayerische Landesbank,
- BMW,
- IXTRA
- Siemens,
- Softlab.

End of 1994 FAST had 20 employees.

The most important objective of FAST is to promote the use and application of software technology by Bavarian companies. FAST acts as a catalyst between users, providers, and universities in the area of software technology.

- FAST offers provider-independent and competent decision support for the assessment, selection, introduction, and use of software technology.
- FAST supports technology transfer through research and development projects, jointly carried out by providers, users, and universities.
- FAST promotes co-operation in the area of information technology and achieves synergy through bundling and concentrating a variety of activities.

# Process Improvement Experiment at MemoLuX

Miklós Biró Dr.  
MTA SZTAKI  
H-1111 Budapest  
Lágymányosi u. 11.  
[miklos.biro@sztaki.hu](mailto:miklos.biro@sztaki.hu)

Éva Feuer  
MTA SZTAKI  
H-1111 Budapest  
Lágymányosi u. 11.  
[eva@ilab.sztaki.hu](mailto:eva@ilab.sztaki.hu)

János Iványos  
MemoLuX Ltd.  
H-1143 Budapest  
Zászlós u. 18.  
[iva@memolux.datanet.hu](mailto:iva@memolux.datanet.hu)

## **Introduction**

Fast, better and cheaper, these three words describe the desires and demands of IT organizations not only in Central and Eastern Europe. Developing quality applications in a time- and cost efficient manner and deliver valuable predictable products is really difficult. The knowledge of recent technologies is not lacking completely in Central and Eastern Europe, but usually the software development process is not controlled, modeled, defined well and there aren't any tools to help the process. It is hard to produce applications in an industry that is in a constant state of flux; methodology, technology and business processes are changing and evolving constantly. So instead of producing software products faster, better, cheaper, the software development projects are actually going over budget, over time and in poor quality.

Applying old solutions to new problems isn't going to work. The solution to improve the business of IT is to improve the process.

## Project History

Process Improvement Experiment started at MemoLuX with the first Bootstrap assessment in Hungary, carried out in 1993 by MTA SZTAKI, the only Hungarian licensed assessor in order to clarify the strength and weaknesses in the software development process with the following conclusion:

	<b>Strength</b>	<b>Weaknesses</b>
<b>Organization</b>	human policy	quality system, quality manual
<b>Methodology</b>	configuration & change management; management of subcontractors;	risk management quality management
	documentation and registration of user and software requirements	architectural design documentation and testing of the dev. Process
<b>Technology</b>		management tools

According to the first BOOTSTRAP assessment, the maturity level of the software producing unit was initial with a value of 1.5 and 1.75 for the selected project. Over 50 % of the attributes were at level 2 while some attributes reached level 3 and 4. Technological support was at 20 %. These values were typical in the IT business in Hungary till the middle of nineties.

In 1995 MemoLuX launched new projects under the name of PASS (Payroll Accounting and Settlement System). For that time MemoLuX became one of the biggest bookkeeping and payroll accounting firms in Hungary with over 70 employees. The company had the potential of developing an integrated modular nation-wide networked software system for payroll accounting, which is part of the public accountancy service. The demand for complex systems of this type was present already at companies belonging to the productive sector of the Hungarian economy and had clients from EU countries.

In the past, MemoLuX had rather short term projects for which the application of high level development methods and tools would have been impractical and costly. The company's profile, however, was changing, the need for undertaking large projects was emerging, which implied the introduction and application of SQA methods.

In June 1995 MemoLuX made two successful proposals in response of the Hungarian and European calls for supporting R&D activities. One proposal was accepted by the Hungarian National Committee for Technological Development (OMFB), which helped to improve the software development process during the contracted period from November 1995 to March 1997. MemoLuX had followed the activity plan, and had taken improvement actions, the software development process improved significantly by the work of MemoLuX own staff, and the results were proved by the second BOOTSTRAP assessment done again by MTA SZTAKI utilizing the financial fund from OMFB. The maturity level of MemoLuX has raised to repeatable process level. The success of the OMFB supported project led MemoLuX to have better starting position when the EU Software Best Practice started.

On March 16, 1995 The IT Program (Esprit) published its sixth call for proposals under the Fourth Framework Program. The changing policy of the European Commission brought the

opportunity for CEEC participants to apply directly to the Commission with proposals and under the Process Improvement Experiments (PIE) Tasks MemoLuX produced an accepted proposal. The unique of this proposal was that the Prime User and Coordinator firstly became from non EU member state and the conditions allowed to formulate a consortium from one country which caused legal difficulties during the contracting period. The legal issue solved by the involvement of ISCN as associated partner of the project and finally the contract came into force by the end of June 1997.

The new features of process improvement under the ESSI PIE project are:

- MemoLuX role as the prime user of the software development activities (Payroll Accounting and Settlement System chosen as baseline development project for process improvement)
- Starting the practice from higher level of maturity model (CMM score is above 2.5)
- Consulting work on quality issues is given by MTA SZTAKI as subcontractor of the project
- Utilizing EU funds for investing in technology (LBMS Process Engineer Tools)
- Implementation of measurement procedures in the project
- Connection to EU dissemination activities by ISCN from the very beginning of the project

### ***Description of the companies, business, product***

#### ***MemoLuX Ltd.***

MemoLuX Ltd., a Hungarian private company with professional experience is a service provider in finance and public accountancy, management organization, software development and information system engineering.

#### ***MTA SZTAKI (Computer and Automation Institute of the Hungarian Academy of Sciences)***

MTA SZTAKI is the largest research institute of the Hungarian Academy of Sciences. Contract-based target research, development, training and expert support for domestic and foreign industrial, governmental and other partners have been key activities of MTA SZTAKI since its year of foundation (1972).

#### ***ISCN The International Software Consulting Network***

ISCN offers professional services in the fields of process analysis, process modelling, process and product measurement, and practical experience with the installation and performance of improvement projects.

Hereby we would like to thank the valuable help and advice of our ISCN consultant: Richard Messnarz.

Detailed description of the companies can be found in Appendix II.

#### ***The PASS project***

It is the first Central and Eastern European ESSI PIE named PASS (Pay Roll Accounting and Settlement System) project directly supported by the European Commission. The PASS project is carried out under the ESSI initiative with the financial support of the Commission

of the European Communities under the ESPRIT Programme EP21223.



Fig.EF.1.: The PASS LOGO

The PASS (Pay-roll Accounting and Settlement System) project started as a new business project of MemoLuX. Its business purpose is to develop a modular, platform independent, integrated networked software system satisfying functional requirements of EU standards in public accountancy and applicable for the Hungarian as well as for the international market. The system provides direct service among Employers, Employees and Banks. The PASS project is the baseline project for the Process Improvement Experiment.

### **Starting scenario**

In PASS project the quality of MemoLuX's development process is largely enhanced to become well defined and predictable, and in a dissemination effort (supported by ISCN) this PIE is used as a master example to adapt Eastern European processes to the high quality norms of Western Europe, this way facilitating the integration of Eastern Europe into a joint EU in the long term.

Starting the PIE project we had to make five very important decisions:

- ? • Roles of the participating organizations
- ? • Team structures
- ? • How we work together
- ? • How we communicate
- ? • Software standards used

### ***The roles of the participating organizations***

MemoLuX Ltd. is the coordinator / prime user, develops the baseline project.

ISCN is the associated partner, provides requirements for control & measurement of results, prepares all results as best practice reports, makes project monitoring and dissemination of results for EU.

MTA SZTAKI is the subcontractor, provides consultation, training, implementation of the measurement plan, makes BOOTSTRAP assessment.

The following figure represents the roles of the organizations.

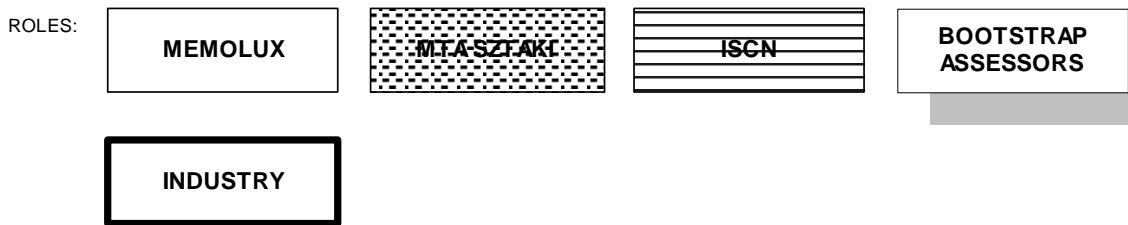
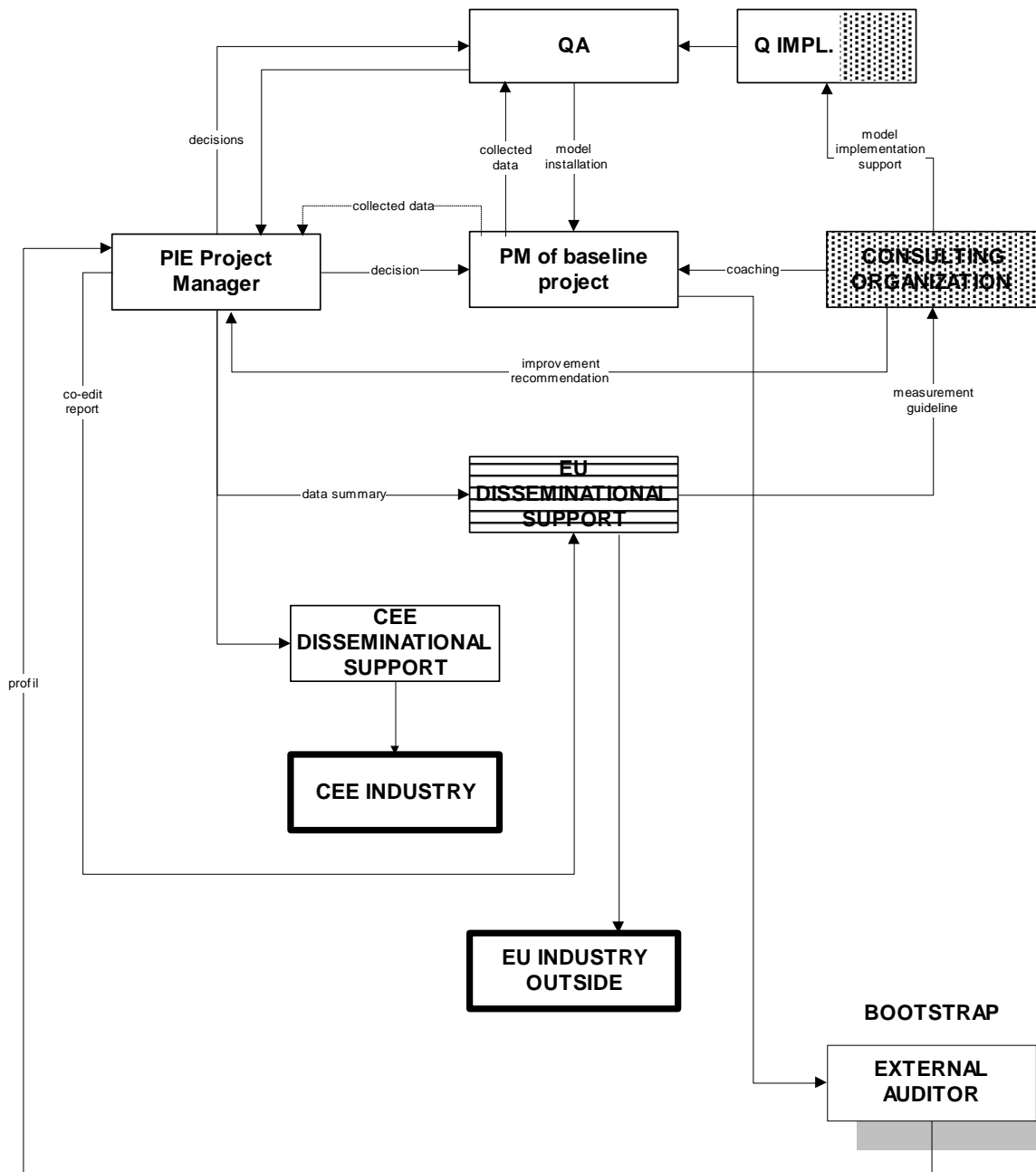


Fig.EF.2.: The roles of the participating organizations

**The team structures**

For completing the phases of the PIE and the baseline project new organizations had been defined. These were the project organization for PIE, the project organization for the baseline



project, Steering Committee, Project Board, QMU (Quality Management Unit) setup for MemoLuX, establishment of the risk management team, etc. The next step was to define the roles, the persons and the organizations involved for each workphase.

### ***How we work together and how we communicate***

Since the problem of who does what was solved by defining the roles and the associated responsible persons, we had to make decisions on how we are going to put our thoughts together. MemoLuX and MTA SZTAKI are both in Budapest but on different banks of the river Danube, but the location of the ISCN expert is constantly moving, sometimes the ocean separates us. Luckily we all have e-mail and computer network, so it was decided to have a commonly used account on a server. At the premises of MemoLuX an FTP server was set up and all the released and work documentation is stored there. The server is always on-line. The system administrator distributes the e-mails from here to everybody on the list.

### ***Software standards used***

The useable software standards were fixed for scheduling and documentation, common templates were made, reviewed, accepted and distributed.

### **Plans and expected outcome**

MemoLuX Ltd. is steadily growing and is managing more and more larger projects. While introducing best practices the company will effectively manage a large number of projects in future by reusing lessons learned.

Utilization and trading of the PASS system belongs to the business policy of MemoLuX. Software process improvement will streamline work processes and align them to ISO 9001 standards.

According to our opinion, any company in a situation similar to MemoLuX could benefit from the PASS process improvement results which show how to start improvement programs in those sectors that are of critical importance in businesses between Eastern and Western Europe.

MemoLuX is able to establish a stable and predictable development process, MTA SZTAKI supports MemoLuX to effectively implement and use the new quality system, and ISCN ensures a European wide dissemination of results plus consulting about how to measure and control process improvement.

ISCN ensures that the results will be discussed in a broader community in the EU via WWW and conferences to make the Hungarian efforts and results visible as well as to enable a feedback loop between EU PIEs and this Hungarian PIE.

### **Implementation of the improvement actions**

As described in the previous parts of this article MemoLuX had decided to improve its development process. The process improvement experiment gave MemoLuX a good opportunity to experiment with and evaluate new methods, procedures and tools in a real life environment to model their processes and to implement a quality system.

Nowadays the formal modeling of processes is gaining increasing interest in the field of analyzing organizations with respect to the quality of their products, productivity and efficiency. The process models are the basis for improvement actions and comparing.

First MemoLuX made an evaluation process to select the most suitable process model tool. The LBMS Process Engineer tool is a system for developing systems, consists of a set of processes for planning, managing, and developing Information Systems and a technology automating the use. LBMS offers a product that not only provides an extensive library of best practices, but can help an IT organization to capture its own brains as organizations best practices. The components of LBMS Process Engineer ordered by MemoLuX are the Process Manager with Process Library and the Project Manager. The Process Library can store best

practices which can be made into MemoLuX's repeatable processes, making a standard for development and raising the expertise of the entire organization. Process Management is the method to capture, deploy, execute and improve best practices for continuous improvement. Applying Process Management new processes can be authored, or the best practices from the Process Library can be customized to satisfy MemoLuX's need to become organization standards. The Project Manager provides the ability to generate detailed project plans based on the processes, define and store information on progress deliverables, roles and resources and apply metrics and estimating models.

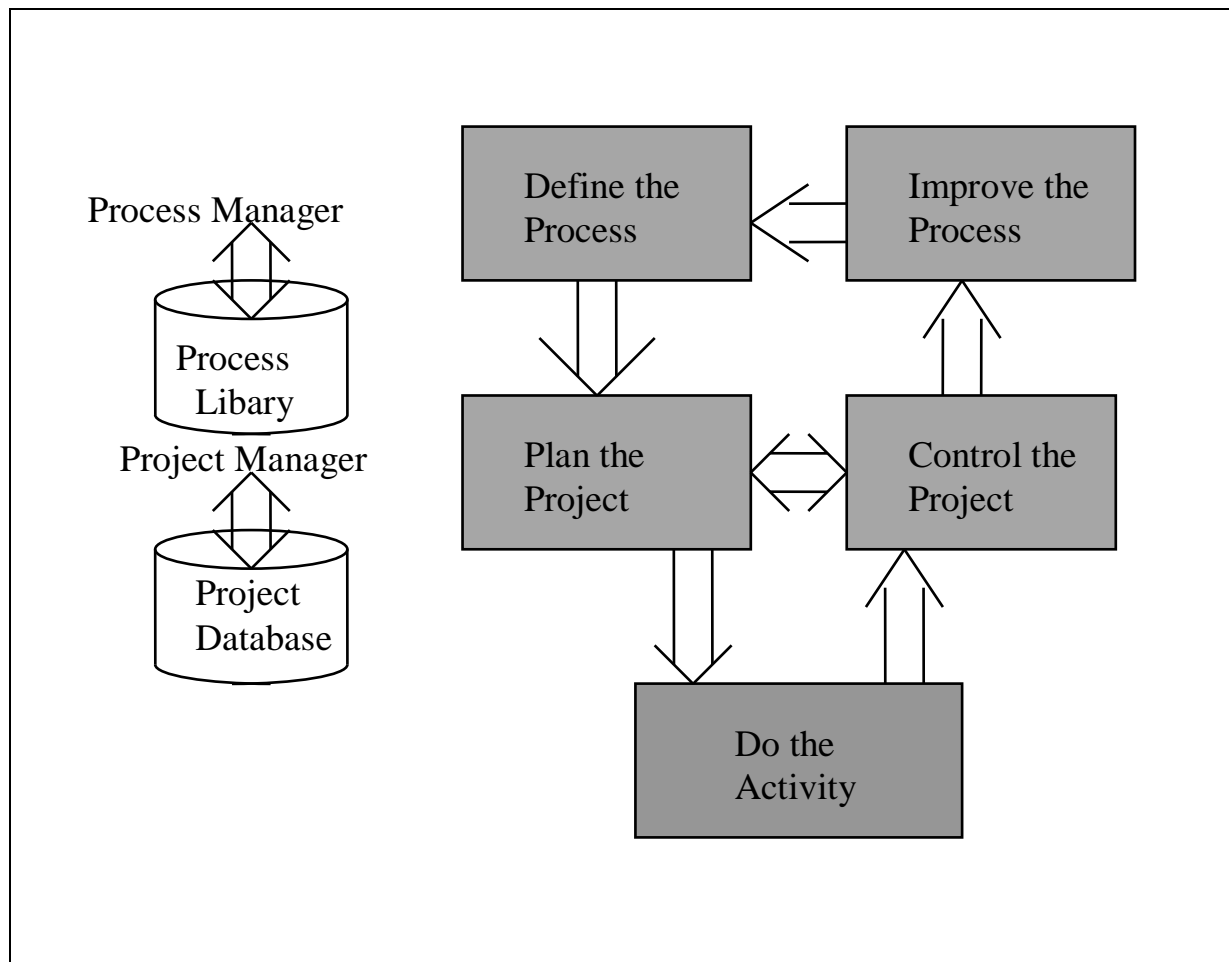


Fig.EF.3.: LBMS Process Management

### Selecting scenarios

The development of a process model is itself a process. This process had to be adapted for the specific needs of MemoLuX. The level of detail to be specified depends on the level of the development process (see BOOTSTRAP assessments), the available computer system configuration, skills and experience of the personnel, and the size of the organization. It was decided to define a simple workflow for software development.

Modeling the workflow it is suggested to identify the following system constituents:

- views
- roles

- activities
- steps
- work
- resources
- objects
- deliverables
- milestones

We selected processes for trying how to build a workflow and using these experiences the modeling process can be completed. The selection criteria was, that the maturity level of selected attributes should be over 2.5 (measured by the last BOOTSTRAP assessment), i.e. the process is a standard, well documented process. Four scenarios were selected:

- Project Plan Scenario
- Review Model Scenario
- Change and Configuration Management Scenario
- Testing scenario

## Measured results and the lessons learned

### *First experience with Process Engineer*

The tasks of the selected scenarios were defined, each consists of five steps:

- Workflow definition
- Implementation
- Coaching and Training
- Using
- Data collection

Starting with the Planning Scenario we have found a predefined process among LBMS processes, which is quite similar to our image of the planning workflow. In this paper we would like to put attention to the scheduling and estimating activities which are supported by a subprocess of the predefined process.

When it was possible we have applied in this first Project Planning Scenario experiment the standard values, because we have found that if something extra is added, the software doesn't seem to support to put it automatically to every location where it is needed. We have tried to make a project schedule according to our plans. This means that the schedule was produced by Microsoft Project first separately and then with the subprocess: Project Schedule and Budgeting.

Naturally our activities and resources have been used as input data and we have applied the export feature to make a Microsoft Project schedule from the data collected by the Process Engineer. To predict the resources that will be required to complete the project an estimating technique was used. Applying this technique effort estimates are developed for the activities in a work breakdown structure. Estimates are used in scheduling to determine the number of resources required for a project and to set milestone and completion dates for a project. Cost estimates are used in cost/benefit analysis to determine the overall viability of a project. It was interesting to compare our effort estimates to the built in ones and to real life. We are going to use this comparison in the baseline project and to improve our development processes.

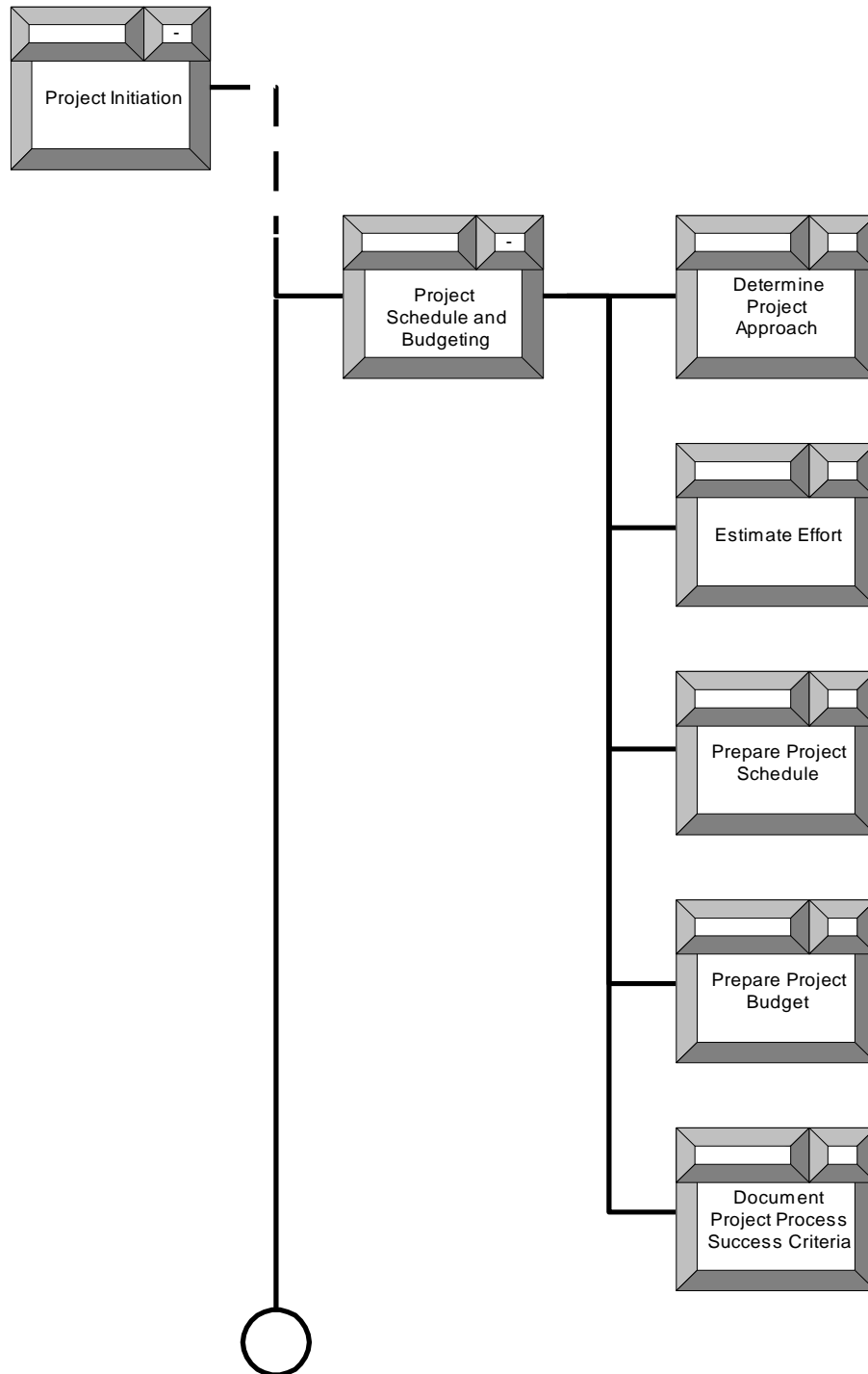


Fig.EF.4.: Project Schedule and Budgeting.

***The architecture of the measurement plan and first results concerning measurement for the planning scenario***

Software metrics were selected to measure the effort and the quality of the process improvement experiment. The architecture of the measurement plan is described in the following:

- Measurement Requirements for the Planning Scenario
- Measurement Requirements for the Review Scenario
- Measurement Requirements for the Testing Scenario

- Measurement Requirements for the Configuration Management Scenario

Implementing the Planning Scenario we have applied the following metrics:

### Measurement Requirements for the Planning Scenario

- the actual schedule (weeks)
- the actual resources consumed (who/days)
- a comparison of the actual schedule with the planned one
- a comparison of the actual resource consumption with the planned one
- a deviation analysis showing the estimated differences between the actual and the planned data

During the workflow definition and implementation of the Planning Scenario we have found problems in applying the predefined roles and resources to our real life situation. Due to this and to our inexperience using the built in processes these steps didn't keep the schedule our efforts were underestimated.

### References

- [1] PASS Annex I, Version 5.1, 23.4.97
- [2] PASS Measurement Plan, Version rev01, 10.9.97.
- [3] Walk K., Messnarz R., Object Oriented Modelling of Work Processes, in: *ISCN'96 Proceedings*, pp. 264-287

### Appendix I

**Éva Feuer** is the deputy head of Quality Management of the Computer and Automation Institute of the Hungarian Academy of Sciences (MTA SZTAKI). She graduated in mathematics at the József Attila University in Hungary.

She is an expert in the field software quality, she deals with software development process quality and also with software product quality.

She is a certified TÜV auditor for ISO 9000 series and BOOTSTRAP assessor.

She is a member of the Statistics Section of the Hungarian Quality Board.

**Dr. Miklós Biró** is the head of Quality Management of the Computer and Automation Institute of the Hungarian Academy of Sciences (MTA SZTAKI). He has a Ph.D. in mathematics from the Loránd Eötvös University in Budapest and an Executive MBA (Master of Business Administration) degree from the International Management Center at École Supérieure de Commerce de Rouen, France in association with the Krannert Graduate School of Management at Purdue University, USA. He has numerous publications in international scientific journals and conference proceedings in the fields of combinatorial optimization, human-computer interaction, decision and negotiation support, and software quality management. He initiated and is managing the membership and participation of MTA SZTAKI in several European organizations and projects. He is member of the editorial board of the journal on Software Process Improvement and Practice published by Wiley, and president of the professional division for Software Quality Management of the John von Neumann Computer Society. He is member of several professional bodies and societies.

**János Ivanyos** is the managing director of MemoLuX and responsible for the IT business and administration work of the company. He graduated at the University of Economics in Budapest and started to work for the National Planning Office in 1984. He established the

MemoLuX company together with his two partners in 1989. He is the PIE project manager within the PASS project.

## **Appendix II**

### ***MemoLuX Ltd.***

MemoLuX Ltd., (URL: <http://www.memolux.hu>) a Hungarian private company with professional experience, is a service provider in finance and public accountancy, management organization, software development and information system engineering. In Hungary, MemoLuX is ranked after the "Big Six", the six greater advisory firms in public accountancy.

MemoLuX Ltd. was established in 1989 as the successor of a computer technique institute having been founded at the early seventies. Since its foundation MemoLuX has dynamically increased its sales and doubled the number of employees. At present, MemoLuX has the potential of developing an integrated modular nation-wide networked software system for pay-roll accounting, which is part of the public accountancy service, the business aim of project PASS. The demand for complex systems of this type is present already at companies belonging to the productive sector of the Hungarian economy and having clients from EU countries.

In the past, MemoLuX Ltd. had rather short term projects for which the application of high level development methods and tools would have been impractical and costly. The company's profile, however, has been changing, the need for undertaking large projects is emerging, which implies the introduction and application of SQA methods:

### ***MTA SZTAKI (Computer and Automation Institute of the Hungarian Academy of Sciences)***

MTA SZTAKI (URL: <http://www.sztaki.hu>) is the largest research institute of the Hungarian Academy of Sciences. Contract-based target research, development, training and expert support for domestic and foreign industrial, governmental and other partners have been key activities of MTA SZTAKI since its year of foundation (1972).

The present staff of MTA SZTAKI consists of about 400 employees, out of which about 250 are university graduates, 60 people have intermediate education, and an assistant staff of about 40 is also available. Graduate employees are highly qualified engineers and mathematicians with ample expertise and practice in advanced information sciences (computers, communication and control) as well as in information technology.

MTA SZTAKI has played a key role in establishing and operating the Hungarian backbone to the Internet.

MTA SZTAKI has experience not only in information technology consulting, systems integration, and software development but in software quality theory and management as well. As an important development, it became the first Central and Eastern European member of the BOOTSTRAP Institute.

The Informatics and Systems and Control Divisions of MTA SZTAKI recently acquired TÜV CERT certification of their ISO 9001 compliant quality system.

MTA SZTAKI has initiated a wide range of activities for the promotion of Software Quality Management (SQM) in Hungary. SZTAKI is participating in the leading Hungarian organizations for quality improvement: the Technology Transfer Office of the Hungarian Committee for Technological Development, the Hungarian Quality Society, and SZTAKI is one of the founding members of the recently established Hungarian Quality Consultants' Association. SZTAKI became one of the first Central and Eastern European corporate members of ESI (European Software Institute). SZTAKI is also the first Central and Eastern European member of ERCIM (European Research Consortium for Informatics and Mathematics).

MTA SZTAKI is partner in several software best practice related European Union projects including ESSI/VASIE (Value Added Software Information for Europe), the first Central and Eastern European ESSI PIE named PASS (Pay Roll Accounting and Settlement System) project directly supported by the European Commission.

**ISCN (International Software Consulting Network Ltd.)**

ISCN (URL: <http://www.iscn.ie>) started in 1994 with a first conference in May 1994 in Dublin dealing with "Practical Improvement of Software Processes and Products". This conference series continued with ESI-ISCN 1995 in Vienna, ISCN'96/SP'96 in Brighton/London, and with ESI & ISCN 1997 in Budapest on Nov. 10 - 12.

In 1995 a group of top experts who met at the conferences formed a company representing a small office for co-ordinating joint consulting and development activities. In 1996 this company started to focus on collaborative cost sharing projects of different partners, all of them focusing on the development of process improvement products, training, and services.

The ISCN office, established as Ltd. organisation in Dublin, is led by 3 international experts, 1 WWW administrator, and 1 co-ordinating office and conference manager. This small office co-ordinates the activities of about 40 associated experts who work in cost sharing projects in which ISCN gets parts of the budget for co-ordination and dissemination.

*ISCN is partner in :*

EPIC is an ESSI dissemination action in which best practice know how (from PIEs) is discussed in up-to-date video workshop environments connecting distributed European workshops. ISCN plans to re-use the EPIC technology experience to connect the different partners of the ESBNET for a long lasting efficient communication and collaboration.

PICO (Process Improvement Combined approach) is an EU Leonardo project (started in 1995 under the EU life long learning programme). It developed a configurable set of training courses, plus a book, and a framework tool. PICO takes into account most recent improvement methodologies and is like an introduction to the different best practices covering process improvement from analysis to success. The book was written by 25 authors from 10 EU countries with contributions from Europe's leading industry. Please find further information at <http://www.iscn.ie/projects/pico/>

ISCN co-ordinates the development of NQA (Network based Quality Assurance environment) which is an Intranet based quality assurance system providing quality documentation guidelines, an on-line quality manual, computer supported project administration, templates with industry examples, and role plays for software development. The major advantage is that it runs on any WWW server and can be used in transnational co-operation, like it is used in ISCN network development cooperations. At the moment the development is co-financed by ISCN and three additional partners, including Hyperwave Ltd. Hyperwave is a hypermedia database system which can store any kind of music, videos, text, etc. and NQA can be used together with Hyperwave as the basic underlying database for Internet based archiving and access control. Please find further information at <http://www.iscn.ie/projects/nqa/>

ISCN partners developed ESD (Expert and company Skill Database) which is a configurable database storing company service and expert skill profiles and providing an expert system functionality to select proper experts and companies based on skill and service data and on restrictions (such as salary, languages spoken,...). Please find further information at <http://www.iscn.ie/projects/esd/>

# Configuration Management for safe Delivery of Software Systems

Martin Brett

Robert Bosch GmbH, Germany, FV/PLI3

## Introduction

“Configuration Management for safe Delivery of Software Systems“ is the title of an Esprit Project (24205) started in May 1997 in one of the development departments of Robert Bosch GmbH.

Today deliveries of software systems sometimes fail due to inconsistent configuration, resulting in increasing costs and late and repetitive deliveries of software. Therefore there are commercial losses and dissatisfaction for both the customer and the supplier. An increasing number of software variants for deliveries expected at the end of 1997 will add to the problem substantially.

## Implementing Software Configuration Management

To improve the current situation a suitable software delivery procedure is to be installed based upon a suitable configuration management system.

This will have an impact on the general method of software development. During the initial phase a method will be developed which will identify all relevant entities, their relation to one another, their occurrence during the software life cycle and rules for organising and handling this information consistently in order to support the software development process.

## Workplan

The project will be included in the ongoing process of improving the software development and will be carried out in three parts (Initial Phase, Implementation Phase, Dissemination).

- Initial Phase (steps 1 and 2, figure 1): covers a detailed analysis of the problem and the definition of a method to solve the problem. The method will be based on the answers to a number of questions to identify the configuration elements relevant for the organisation.
- Implementation Phase (steps 3 and 4, figure 1): includes the application of the method previously defined during a number of software deliveries, finding a suitable tool which fulfils the requirements defined in the Initial Phase and finally the further delivery of software systems using the new method and tool.



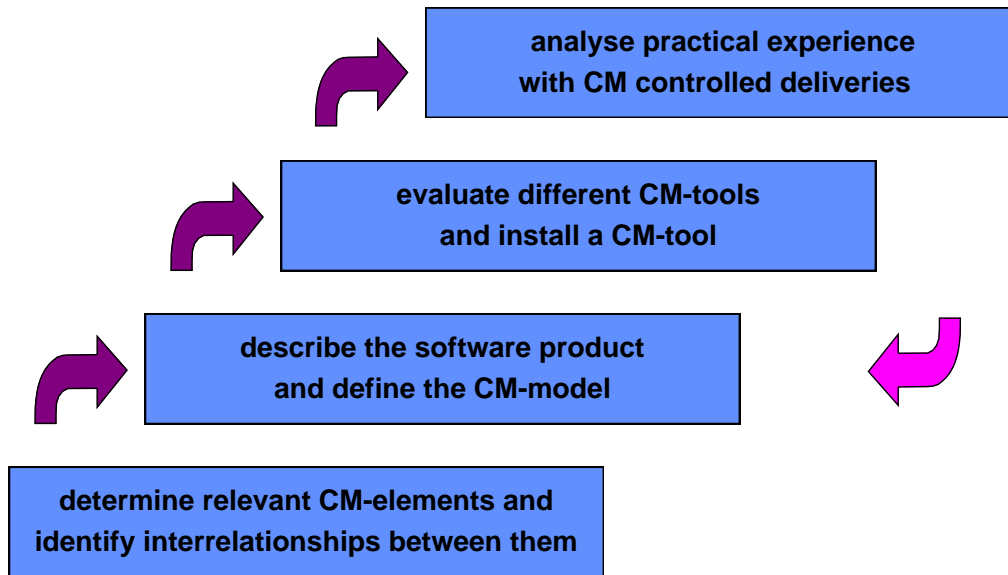


Fig. 1: Introductory steps for implementing Software Configuration Management

## Appendix

### *Author*

For the last three years the author has been working on software improvement processes starting with bootstrap assessments and setting up a number of im-provement projects.

Martin Brett  
Corporate Research and Development  
Robert Bosch GmbH

Martin.Brett@pcm.bosch.de

### *Company*

The BOSCH group has 156,000 employees world-wide (1/1/1996) and is renowned not only as a mayor supplier of automotive equipment, but also for the vast range of products it offers in areas such as public and private telecommunication, electrical powertools, domestic appliances, packaging machinery and automation technology. BOSCH is the worlds biggest independent manufacturer of automotive equipment.

## References

- [1] Implementing Configuration Management; Hardware, Software, and Firmware  
Fletcher J. Buckley, IEEE Press Order Number: PC0332-7, IEEE Computer Society Press Order Number: 3192-21, 1993, ISBN 0-7803-0435-7
- [2] Software Configuration Management Guidebook  
Mordechai Ben-Menachem, McGraw-Hill Book Company, 1994, ISBN 0-07-709013-6
- [3] Configurtation Management: Principles, Tools and Practice; Proceedings

UNICOM Seminars Limited 1997, Brunel Science Park, Cleveland Road, Uxbridge, Middlesex, UB8 3PH, UK

[4] Ovum Evaluates: Configuration Management Tools

Ovum Limited, 1 Mortimer Street, London W1N 7RH

**ESI & ISCN 1997 conference, 10-12 November, Budapest, Hungary**

**Industrial Experience**

**Configuration management  
for safe delivery of software-systems (CMSDS)**

**Implementing software configuration management**

## **Contents**

- **Esprit Project (24205)**
- **Software engineering process in the automobile industry**
- **Installation of configuration management (CM)**

**“Wrong Way“**

**“Better Way“**

**Baseline and the concept of the four steps**

- **Summary**

- **Esprit Project (24205)**

**ESSI PIE Project Program started in May 1997**

**Goals: Reducing the necessary effort required for software deliveries**

**Improving predictability of software deliveries**

**Current status**

**SPI: Bootstrap Assessment 1995 / 97 → Improvement process**

**Engineering: Variety of modules / increased number of variants**

- **Software engineering process in the automobile industry**

### **Characteristics**

- **Department of 80 engineers developing hard and software**
- **Embedded control systems**
- **Assembler and C programming language**
- **Module size approx. 20-1000 / 200-2000 LOC**  
**100 - 200 Modules per program**
- **Reaction to late customers requirements**
- **2 - 5 “Make File“ based deliveries per month**

- **Installation of configuration management (CM)**

**“Wrong Way“**

- **Problems develop with managing source files**
- **Consider possible tools for solving these problems**
- **Evaluate CM-tools on paper or with demo programs**
- **Buy and install one of these CM-tools**

**→ wondering why the tool is not readily accepted**

## **“Better Way“**

### **Concept of standardised method for implementation of CM**

- a) Baseline: All developers must have good knowledge of CM**
- b) Four steps to implement CM:**
  - ⇒ determine relevant CM-elements and identify interrelationships between them**
  - ⇒ Describe the software product and define the CM-model**
  - ⇒ Evaluate different CM-tools and install a CM-tool**
  - ⇒ Analyse practical experience with CM controled deliveries**



## **Baseline**

- **Management commitment**
- **Installation of a CM-team**  
(management, project leader, quality assurance, software and hardware engineers, external experts)
- **Definition of a common aim**
- **Visualisation of a road map for implementing CM**
- **Agreement on common terminology and technology of CM**

**→ to know what CM is !**

## **Four steps to implement CM:**

### **1. Determine relevant CM-elements and identify interrelationships between them**

- Collect all CM-elements in the organization (system, hard-, software)**
- Classify the CM-elements**
- Define the relevant CM-elements (all elements to deliver)**
  - Identify direct and indirect relationships between the CM-elements**

**→ to know which elements you have to manage**

## **2. Describe the software product and define the CM-model**

- Define the status of the elements (life cycle of the elements)**
- Represent the software development process in the CM-model**
- Define rules for the delivery process**
- Define the roles (rights) in the CM-model**
- Describe the CM-model**
- Discuss and improve the CM-model**

**→ to know what it means to work with CM**

## **Workplan for the next phases**

### **3. Evaluate different CM-tools and install a CM-tool**

- Demonstration Presentation of different CM-tools**
- Evaluation against the defined rules / the CM-model**
- Installation of (2) CM-tools**
- Selection of the CM-tool in collaboration with the users**

### **4. Analyse practical experience with CM controled deliveries**

#### **Measurement of results**

- Number of incorrect software releases**
- Delivery time / intervals**
- Degree of automation**

**- Number of late deliveries**

- **Summary**

**Experience after completing half of the project (two steps)**

- **Knowing what CM is**

- **Good understanding of configuration methods**

→ **Recognizing the advantages and disadvantages of working to rules**

- **Knowing that the difficulties in CM are not caused by a tool !**

- **Being well prepared for evaluating a CM-tool**

→ **No false ideas about working with CM**

**Books**

- **Implementing Configuration Management  
Hardware, Software, and Firmware  
Fletcher J. Buckley ISBN 0-07803-0435-7**
  
- **Software Configuration Management Guidebook  
Mordechai Ben-Menachem ISBN 0-07709013-6**
  
- **Ovum Evaluates: Configuration Management Tools  
Ovum Limited, 1 Mortimer Street, London W1N 7RH**

**Internet <http://www.cordis.lu/esprit/src/stessi.htm>**

**Robert Bosch GmbH**

**FV/PLI3; P.O.Box 300240; D-70442 Stuttgart; Germany  
Telefax: +49/711 811-3960; E-mail: [Martin.Brett@pcm.bosch.de](mailto:Martin.Brett@pcm.bosch.de)**

# SPI: an experience report from GSM development

S. Di Muro, S. Humml, A. Lora, S. Scotto di Vettimo  
Italtel SpA, Business Unit Reti Mobili, SS11, Km. 158 – 25060 Cassina de  
Pecchi (Milano)-Italy, e-mail: [lora@italtel.it](mailto:lora@italtel.it), [scotto@italtel.it](mailto:scotto@italtel.it)

G. Bazzana, G. Rumi  
ONION Communication-Technologies-Consulting, Via Gussalli 11-25131  
Brescia-Italy email: [gb@onion.it](mailto:gb@onion.it), [gr@onion.it](mailto:gr@onion.it)

## Introduction

The paper describes the experiences gained in a major Software Process Improvement (SPI) program undergoing at Italtel BURM (Business Unit Reti Mobili) as a support to the development of telecommunications systems for mobile telephony in accordance with the GSM (Global System for Mobile Communications) standard.

The following aspects are dealt with:

- the business motivation in terms of company profile, business needs, the reference development process and an insight of the GSM product line;
- the Process Improvement program in terms of the starting scenario, the Improvement Plan and its organisational issues;
- the improvement actions deployed in the first wave (July 1995 to December 1996) as well as the ones undergoing in the second wave (1997);
- the introduction of formal specifications languages and tools, supported by the European Commission with the SPECS Project, in the context of the ESSI Program;
- the impacts and the experiences gained.

## Business motivation

### **Italtel Company Profile**

Italtel is one of Europe leading full-line manufacturers in the telecommunications field, with over 11.600 employees and an income of over 4.000 Billions Italian Lire.

Italtel designs, manufactures, markets and installs systems and equipment for public and private applications. In Italy and abroad it implements systems and networks on a turnkey basis. The company is active, full-line, in every field of telecommunications, ranging from public switching to transmission, mobile telephony systems, private telecommunications, electronic interconnections, integrated IT-based systems for traffic and environmental monitoring, defense communications systems, modular metal structures and electric panels.

Research and development is crucial to Italtel's competitiveness on the world markets; in the last years Italtel has invested in excess of 15% of sales in R&D in the most advanced sectors of telecommunications.

Italtel is since January 1996 controlled by two leader companies in the telecommunications market : Stet and Siemens AG.

For more details on the company, the interested reader is referred to the Italtel WWW at the URL <http://www.italtel.it>

Figure 1 shows the presence of Italtel companies world-wide



Fig. 1 – Italtel World-wide

Figure 2 shows the location of Italtel sites in Italy, detailing also the activities managed in the various sites.



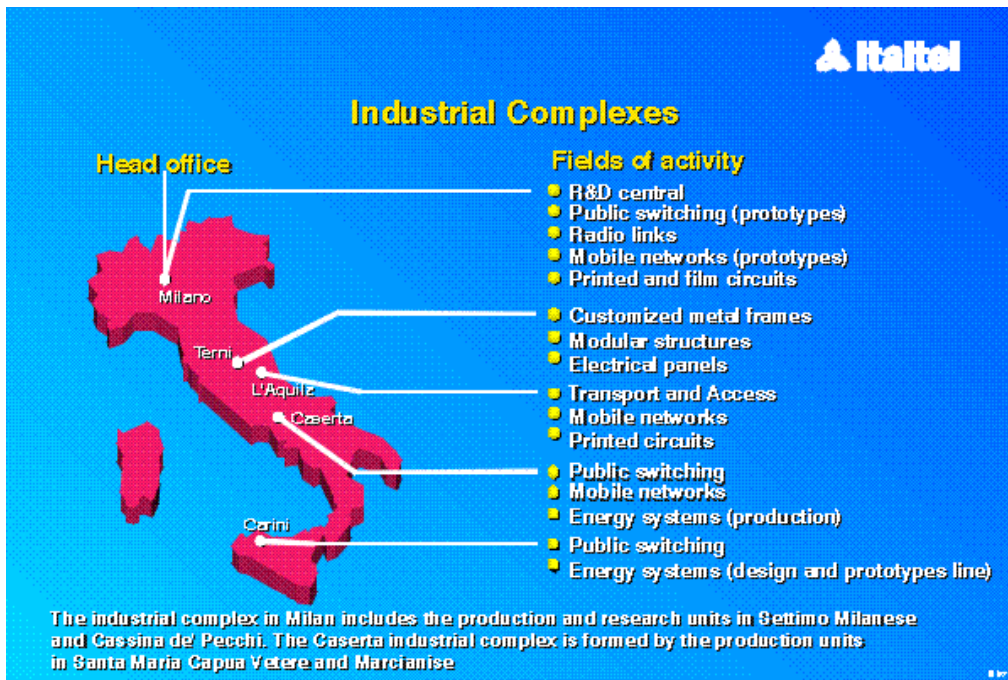


Fig. 2 – Italtel sites in Italy

Figure 3 shows the sales revenues (in Billions Italian Lire) for the last four years. The figure shows also the percentage of income from export, giving evidence of a progressive internationalization of the activities.

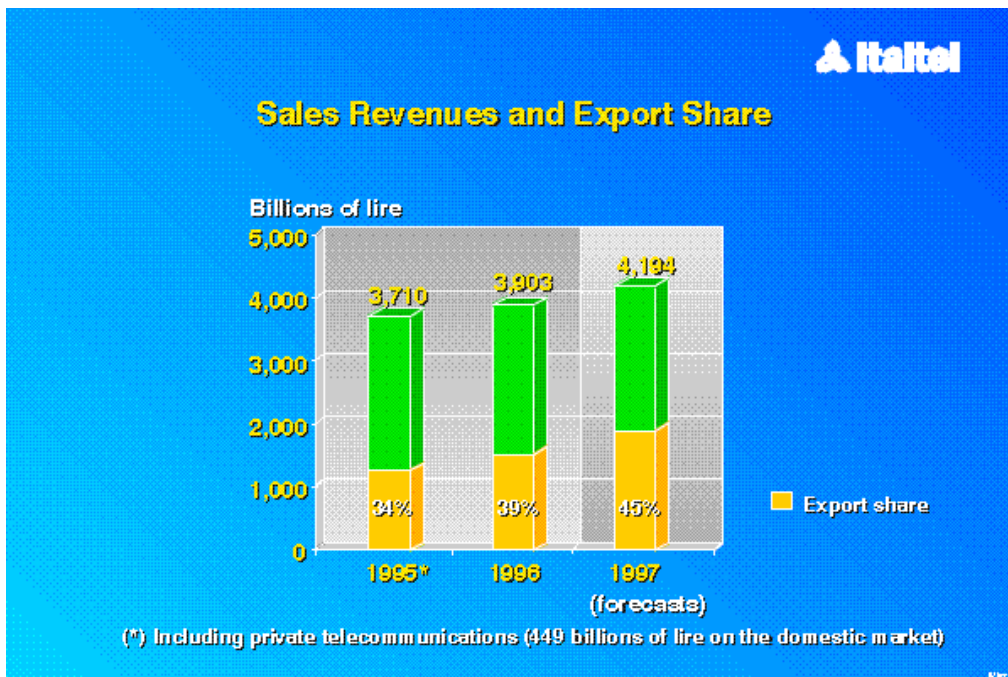


Fig. 3 – Italtel sales revenues

### **Business needs**

Concerning software process improvement, the following directions have been set at company level:

- software process improvement is deemed as a strategic asset for continuously increasing the company capabilities in an extremely competitive world-wide market;
- for all Business Units, the following directions are valid:
  - SPI shall be based on/ combined with the attainment of ISO 9001 certification;
  - SPI shall be based on an initial analysis of strengths and improvement opportunities (process assessment);
  - SPI shall be based on a reference model allowing the quantitative appraisal of achievements gained;
  - quantitative measures have to be supplied on a regular basis to the central R&D for integration and reporting to the high management; to this end, a common set of basic metrics has been defined, to which all business units have to adhere;
  - SPI shall be run under the responsibility of the R&D of the individual business units, in order to be as much as possible in touch with the needs and peculiarities of the addressed markets;
  - the central R&D and Quality shall act as catalyst for supporting the SPI initiatives of the business units, by means of organizing/ conducting training, providing expert's advise, maintaining awareness on the international trends (Italtel actively participates in the SPICE initiatives and in many other international research projects), circulating the success stories at company level, reporting the quantitative results in a coherent way and, last but not least, giving pragmatic support in specific phases of SPI (for instance: process assessment, modeling of new processes, alignment to ISO 9001 requirements, analysis/ interpretation of measurement data, root cause analysis, etc.).

This “distributed” approach (which implies the absence of a centralized SEPG - Software Engineering Process Group) has been felt so far successful in that it guarantees focus and effectiveness of the local initiatives, while keeping a sufficient control level on directives and results. The “distributed” approach implies also that SPI programs run at different Business Units can adopt different technical/ organisational choices, while keeping the same philosophy and approach.

The process improvement activities described hereafter are undergoing at Italtel BURM (Business Unit Reti Mobili), in the site of Cassina de' Pecchi, in the Milan surroundings.

This Business Unit is committed to the development of global solutions for mobile communications in the world-wide market. A great deal of effort is currently invested in the GSM application domain (GSM 900, DCS 1800, PCS 1900, etc.) owing to the rapid market take-up, which is experiencing an unprecedented widespread growth rate.

Italtel BURM is strongly committed to software process improvement to increase the company capabilities. This is motivated by the high world-wide competitiveness in the target domain, by the increasing complexity of the software embedded in the delivered systems and by the fact that projects are developed on an international multi-site basis.

Also customers are more and more demanding on software process maturity and stability. As an example, it is worth underlining the fact that one of the major customers took an active place in the process improvement program, by asking for process audits and by providing indications on the key process areas to be improved from its point of view. The “voice of the

customer” proved to be a key factor in reinforcing the commitment and driving the improvement actions towards a greater effectiveness.

### The reference development process

The software development life cycle can be summarized as follows:

- 1 **Analysis:** the goal of this phase is to analyze a given set of system requirements in order to provide the best software development within the system architecture; mapping of functions into the defined software architecture is also performed in this phase.
- 2 **Design:** the goal of this phase is to identify the complete software behavior for each subsystem and the functions to be provided by each component. This level of refinement includes enough details in order to allow the subsequent coding phase.
- 3 **Implementation:** the implementation phase is structured in the three following activities:
  - 3.1 Coding and debugging, with the goal to translate design information into source code files using the defined computer language, and to check the syntax and the semantic correctness of each source file;
  - 3.2 Test design and development, with the goal to define the testing strategy, to design the tests and to prepare the environment for the relevant testing phases
  - 3.3 Off-line testing, with the goal to execute module testing in a simulated environment for the new/ modified functions and to carry out non-regression testing for the unchanged functions with respect to the previous release.
- 4 **Integration testing:** the phase is structured in the three following activities:
  - 4.1 White-box testing with the goal to carry out white-box functional testing in the target environment;
  - 4.2 Black-box testing with the goal to carry out black-box functional testing on target;
  - 4.3 SBS integration testing with the goal to test the old and new functions with all network elements connected.
- 5 **System test:** the goal of the phase is to perform Independent Verification and Validation.
- 6 **Maintenance:** the goal of the phase is to perform the software product modifications due to correction of defects arisen after the product delivery or due to change requests.

The phases of the software life cycle are reported in Figure 4, together with the baselines (BX00) that quantitatively track the project progress.

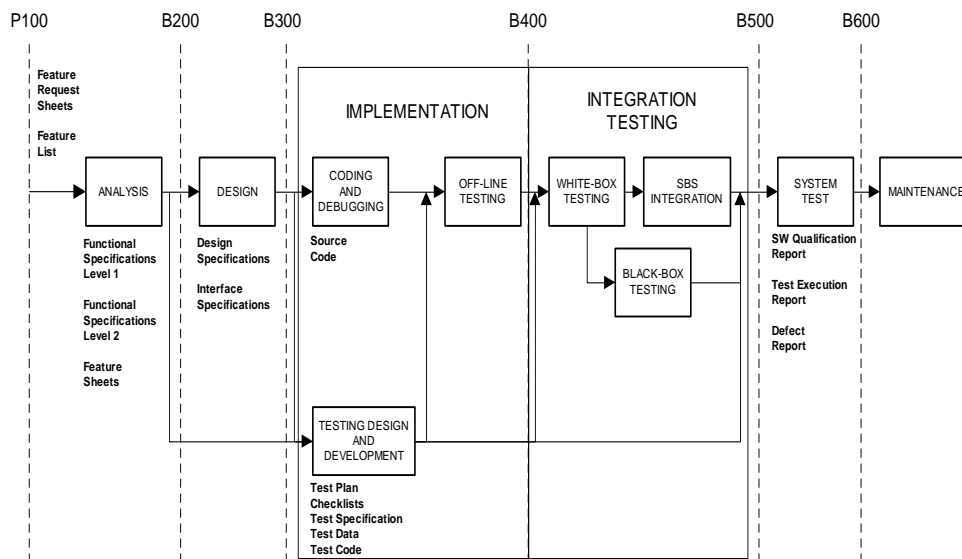


Fig. 4 - The software development life cycle

### The GSM product line

GSM is an European born family of standards for the mobile telecommunication Digital Cellular Systems [1], allowing telephony services through mobile phones. Mobile telecommunications is not a very recent technology, but it is a rapidly evolving one. The main differences with wire-line telecommunication access are:

- mobility management: as a consequence of the fact that subscribers can continuously change their point of access to the network, routing of calls involves new concepts like: location management, handover (automatic transfer of a call in progress from one cell to another without speech disturbance) and roaming (free circulation of mobile stations across networks handled by different operators);
- radio resource management: the link between the subscribers and the fixed infrastructure is not permanent and wave propagation limits and spectrum scarcity have to be taken into account.

From the architectural point of view, a GSM system is quite a complex object, since it has to deal with multi-services and with the peculiarities of cellular networks. Looking at the system from the outside, GSM is in direct contact with users, with other telecommunications networks and with the personnel of the service providers.

The internal GSM architecture distinguishes three parts: the BSS (the Base Station Sub-System), that is in charge of providing and managing transmission paths, the NSS (Network and Switching Sub-System), that is in charge of managing the communications and the OMC (Operation and Maintenance Centre) which provides the interface to the system for the network operator.

Getting into details of the BSS, we can find the following Network Elements:

- a transmission equipment (the BTS - Base Transceiver Station);
- a managing equipment (BSC - Base Station Controller);
- a speech encoding/ decoding equipment (TRAU - Transcoder and Rate Adapter Unit).

A simplified view of the architecture of a GSM system is provided in the following picture.

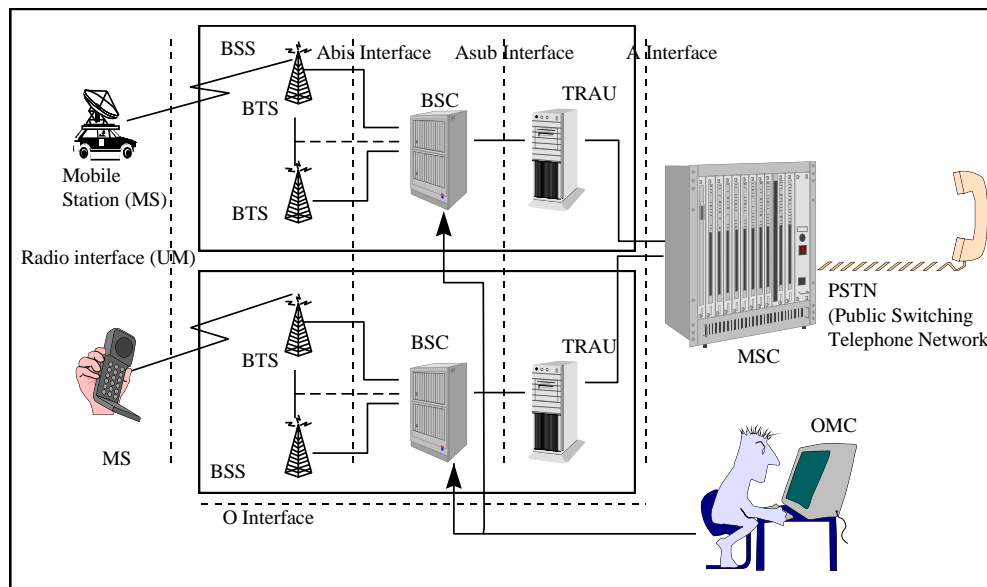


Fig. 5 - Typical architecture of GSM systems

The goal of the BURM projects referenced in this paper is to develop the OMC, BSC and TRAU Network Elements, together with a Local Maintenance Terminal (named: LMT) for controlling the Network Elements.

## The Process Improvement Program

### **Starting scenario**

Before the start-up of a formal Improvement Program, SPI at Italtel BURM was already an established practice, with a number of actions run between 1992 and 1995, namely:

- enforcement of Quality Assurance (QA) practices;
- definition of guidelines for all the phases of the development life-cycle;
- progressive strengthening of Configuration Management methods and tools;
- enhancement of planning and tracking practices, with strengthening of Project Plans and Gantt charts;
- enhancement of test design and administration practices, with introduction of a separate team for testing within the development structure;
- enhancement of tool support for test execution;
- adoption of a Qualification Report accompanying load release to system test;
- introduction of tools for source code static analysis;
- introduction of a measurement system and development of tools for data collection (Lines of Code - LOC, effort, etc.).

Such improvement actions allowed the Business Unit to attain very important results, both from a technical point of view and from the point of view of the Quality Management System that was certified as ISO 9001 compliant. At that time it was felt that the improvement efforts had to be kept as a continuous activity, in order to cope with an extremely competitive market; for this reason, software process assessment (SPA) activities were undertaken in order to focus the resources onto those areas that should provide the highest added value.

The Improvement Program started with a Software Process Assessment conducted by Siemens Central Research, Application Centre Software [2]. In May-June 1995, some of the most important projects in the GSM area were subjected to a formal process assessment; the activity was very broad and detailed, with interviews conducted with more than 35 people from the involved projects.

The assessment highlighted a good maturity level for the software producing unit, singling out also some improvement opportunities.

The results of the assessment can be summarized as follows:

- with reference to the CMM approach, the working practices were found to be in between the repeatable and the defined level, depending on the projects. Even if such result was felt as appropriate for the Business Unit at that time, the management understood the need for continuous improvement and strengthening of capabilities in order to face the growing world-wide competition and to increase customers' satisfaction.
- Very positive aspects found were the co-operative attitude of the staff and the alignment between rules and practices. Such aspects were felt as very promising in the light of a SPI Project that would have impacted the existing behaviors.
- Considering the scope, relevance and size of the impacted projects, it was decided to manage the improvements in a systematic way, setting-up an ad-hoc project to which an effort of about 5% of the R&D structure had to be devoted (including SPI Teams, but excluding the deployment of actions with a large effort associated).
- The assessment resulted in a catalogue of about 100 detailed recommendations accompanied by a portfolio analysis of the action clusters, highlighting the areas that should be attacked first.

The top ten recommendations for bettering cycle time and product quality are listed in the following (the order does not imply any priority):

1. technology innovation for the development environment;
2. strengthening of communication facilities across distributed development teams;
3. improvement in change request procedure;
4. enforcement in inter-working tests;
5. automation of target tests;
6. planning and managing of system test;
7. enforcement of handling of interdependencies across projects;
8. improvement of pre-analysis activities;
9. analysis of metrics and root cause analysis;
10. improvement in design and interface specifications.

### ***The Improvement Plan***

The Process Improvement Plan was designed to meet the business goal of the business unit rather than to get to a pre-defined maturity level.

This is especially important for an organization like Italtel BURM that has already reached both ISO 9000 registration and a satisfactory maturity level and that has to deal with a competitive market requiring challenging capabilities in terms of delivery timeliness, product quality and customer support.

Driven by such considerations, the Italtel BURM management decided to start-up a challenging Process Improvement Project, whose high-level objectives can be summarized as follows:

- to optimize the predictability of schedules and the reaching of timeliness goals;
- to further enhance product quality;
- to raise the availability and usability of documentation (both technical and user-oriented);
- to better the tool support to development activities;
- to keep productivity levels at the current levels, while reaching the goals mentioned before.

The analysis made brought to the setting-up of an eighteen months Process Improvement Project, focused on the following key aspects:

- planning, tracking and oversight;
- requirements engineering;
- integration and system testing;
- software development technology.

The choice of a limited number of topics was made with the aim of avoiding the errors reported by other SPI programs that tried to solve all issues too rapidly.

For each of the selected topics, a Working Group was established in order to propose, experiment, validate and apply those improvement actions showing the best return on investment. For each of the selected areas, the following rules were taken into account:

- one person shall be put in charge of the improvement area (what is normally referred as “the process owner”);
- the Plan-Do-Check-Act paradigm shall be followed;
- a core set of basic metrics shall be defined at the beginning in order to collect data able to track and quantify the impact of the experienced improvements;

- two lines of action shall be planned: the short term implementation of those actions that require small overhead/ elapsed time and the medium-long term planning of those actions involving a large amount of resources. This split is needed in order to focus both on quick solutions (that are very good for feeding and caring the enthusiasm) and on longer ones (that are likely to have a greater impact);
- the activities should be synchronized with project milestones, trying to define solutions just-in-time for their application within a (sub)-project: this is extremely beneficial both for having timely feedback and for focusing on pragmatic issues and feasible solutions.

This last issue is particularly important considering the fact that it was dangerous to activate the biggest effort of all four areas in parallel: in this case the projects would have been overwhelmed and probably many improvement actions would have been rejected simply because of lack of time. As a consequence, the high level plan was defined in such a way that the improvement actions followed a just-in-time strategy and the parallelism of the ‘Do’ step of the long-term actions was not too high.

### Process Improvement organisational issues

The SPI Program is intended as a continuous effort, handled with a management-by-objective approach with milestones and quantitative results.

In order to ensure its success, the SPI Project has been organized as follows:

- a SPI Steering Committee (referred in the following as “PISC”), chaired by the R&D Director and including all the managers reporting to him. The aim of this board (that meets regularly on a monthly basis) is to define priorities, assign resources, solve problems and track the success of the initiative;
- a SPI Project Office (called “PIPO” and equivalent to a SEPG), composed of a few experts, having the goal of planning/ tracking the project, giving technical guidance and harmonizing/ deploying the outcomes of the Working Groups; the PIPO has also the duty to organize the so-called “accompanying actions”, namely: training, dissemination and quantitative measurement.
- A number of Working Groups, composed of technical representatives from the various projects involved and dealing with improvement actions.

The following picture gives an overview of the adopted organisational structure.

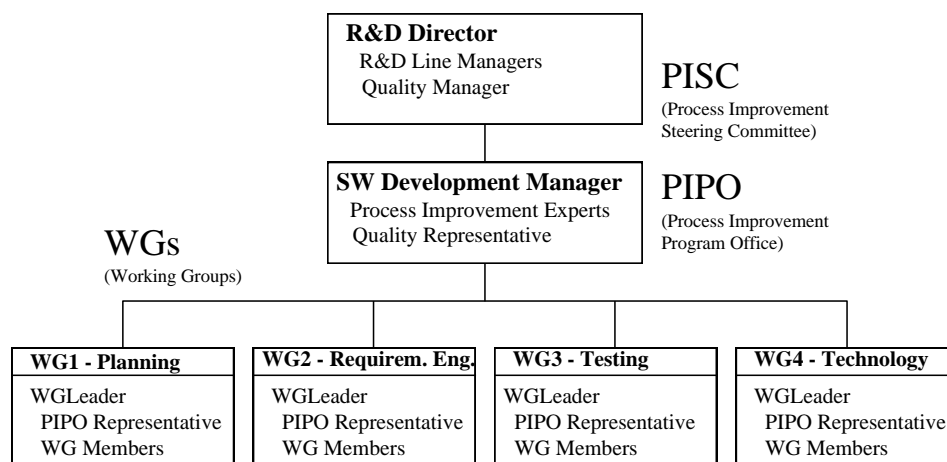


Fig 6 - Process Improvement Organisational Structure

## Improvement actions deployed

### **SPI first wave**

In the following, details are given of the improvement actions deployed in the Italtel BURM SPI first wave, that is to say from July 1995 to December 1996. Description is given in accordance to the defined Working Groups. For more details the reader is referred to [3].

### **Planning/Tracking and oversight**

Planning/ tracking/ oversight of big projects organized following a multi-site development scheme is an inherently challenging job. In order to master the complexity of projects, planning is managed by a separate organisational unit that provides high level plans, checks project progress on a regular basis, reports the status to the management and performs risk assessment and analysis, following up any decided action item.

Each phase is marked by a baseline, whose achievement has to be formally declared in accordance with pre-specified quality criteria, concerning the completeness of documentation for the specific phase, the level of coverage for documentation of the following phase, test coverage, fault density, absence of operational restrictions, etc. Phases are in turn subdivided into activities associated with well defined milestones.

The Software Process Assessment singled out the following strengths concerning Planning/ Tracking/ Oversight: presence of accurate plans agreed at all relevant levels, regular project tracking activities, collection of quantitative measurement, circulation of information about such data.

At the same time, the following improvement opportunities were identified: co-ordinated project management of parallel projects, extension of the usage of Root Cause Analysis (RCA) and adoption of planning/ tracking tools purported for large projects.

As a consequence of such recommendations and of subsequent technical analysis, the following actions were identified and deployed:

- enforcement of handling of interdependencies across projects;
- systematic metrics collection and root cause analysis;
- strengthening of oversight mechanisms through Progress Trend Analysis (PTA);
- collection and analysis of estimates and planning data from previous and current projects;
- adoption of more powerful planning/tracking tools.

### **Requirements engineering**

The Analysis and Design phases of the reference development projects are performed following a waterfall life cycle (with heavy concurrent engineering) that starts after the completion of Pre-Analysis activities. The development teams are put in charge of analyzing the impacts on the existing system caused by the additional features; as an output, designers produce "Feature Sheet" documents, detailing the impacted subsystems, the technical activities to be performed and the effort estimates. After the review/ approval of Feature Sheets, analysis activities bring to the drafting of Functional Specifications documents (one for each functional area); such documents are the starting point for testing activities as well as for Customer Documentation drafting. After the review/ approval of Functional Specifications, design activities bring to the drafting of DSD - Design Specifications Documents (one for each subsystem) as well as Interface Specifications documents. Such documents are the starting point for coding.

Any modification to approved specifications is handled through formal Change Requests. All documents are written in English language in accordance with documentation standards.

The Software Process Assessment singled out the following strengths concerning Requirements Engineering: availability of well-defined standard operating procedures and guidelines; impact analysis; formalized interface documents.



At the same time, the following improvement opportunities were identified: enforcement of details of DSDs and interface specifications; more controlled update activities in order to keep the documentation aligned with the implementation; adoption of the SDL specifications language and related CASE tools (in order to increase the quality of documentation, perform consistency and completeness checks and generate code frames).

As a consequence of such recommendations and of subsequent technical analysis, the following actions were identified and deployed:

- improvement in change request procedure;
- enforcement of pre-analysis activities;
- improvement in design and interface specifications;
- strengthening of traceability mechanisms;
- tuning of technical documentation guidelines;
- alignment of technical documentation to new guidelines for running projects in order to cover the whole documentation tree;
- re-enforcement of code review procedures, combining tool driven checking with human inspections.

### **Testing**

Besides review activities performed during the development phases, the validation of GSM systems involves several complex and effort-intensive tasks, that can be summarized as follows:

- unit testing: verification and validation (V&V) of the single software module in the development environment;
- off-line (host) testing: V&V of different software modules in a simulated environment;
- white-box testing: V&V of complete features in the target environment;
- black-box testing: V&V of a complete Network Element at the external interfaces in the target environment;
- SBS integration testing: V&V of interconnected Network Elements in fully equipped configuration;
- system test: V&V of the global system in the final environment, with an end-user perspective;
- acceptance testing: V&V of the system with the user in field environment.

At each step, regression activities have to be performed with respect to features delivered in previous releases, features delivered in previous loads of the release under development, stability of the system after fixing of faults and/ or implementation of Change Requests, changes in hardware/ firmware/ operating system/ configurations etc. As a consequence, regression testing has to be thorough, requiring considerable staffing. Moreover regression testing is subjected to severe deadline pressures. Besides the usual problem posed by big systems, the following aspects had to be taken into account:

- the equipment is quite complex;
- the numberless possible configurations of the system cause an exponential growth of situations to be considered;
- the test beds (environment and tools) need to be prepared ad-hoc;
- the wide range of items to validate (firmware, operating systems, transmission protocols, application SW dealing with call processing, application SW dealing

with operation and maintenance, etc.) requires utterly different approaches in test design and execution;

- the many-to-many relationships across physical objects (processors and executable processes) and features requires an accurate planning of deliveries and synchronization points among the various development teams.

Early error detection and anticipation of test activities in a more “friendly” environment are therefore very important for both productivity and product quality.

The Software Process Assessment singled out the following strengths concerning testing: test design activities performed starting from analysis/ design documents in parallel with development activities; existence of a well defined test life cycle, with associated documentation and responsibilities; root cause analysis activities and collection of quantitative data about fault density, test effectiveness, etc.

At the same time, the following improvement opportunities were identified: strengthening of the testing across Network Elements, quality management of system test activities (planning/tracking, reviews, configuration management), adoption of test automation facilities, strengthening of the host testing phases in the development environment.

As a consequence of such recommendations and of subsequent technical analysis, the following actions were identified and deployed:

- enforcement in inter-working tests;
- planning and managing of system test;
- strengthening of host (off-line) testing;
- consolidation of regression testing suites;
- improvement in object patch management.

### **Technology innovation**

Technology innovation is fundamental to support the introduction of enhanced methods and procedures: as a matter of fact, if the new practices are not substantiated by gains in the daily routine work, it is very likely that they get abandoned very soon and the whole SPI tends to be considered a “bureaucratic” exercise.

The Software Process Assessment singled out the following strengths concerning Technology: robust configuration management environment, strong testing tools, safe evaluation and procurement procedures.

At the same time, the following improvement opportunities were identified: procedures for technology innovation, communication means, CASE tools for the initial phases of the life cycle.

As a consequence of such recommendations and of subsequent technical analysis, the following actions were identified and deployed:

- Technology innovation for the development environment;
- Strengthening of communication facilities across distributed development teams;
- Tool support for cross-checking between plans and configuration management environments;
- Market survey for formal specifications languages and tools.

### **Impacts from SPI first wave**

#### ***Coverage of initial goals***

The outcomes of the first wave of Italtel BURM SPI can be summarized as follows:

- Excellent results have been gained on:
  - strengthened planning/tracking practices and tooling;

- documentation methodology and guidelines;
  - testing methodology;
  - software factory evolution;
  - harmonization across projects;
  - training dissemination;
- Good results have been gained on:
- deployment of practices in projects;
  - interface management;
  - coding guidelines and reviews;
  - test tooling;
  - communication facilities;
  - evolution of the Configuration Management environment.
- Improvement opportunities are still evident for:
- estimation practices;
  - systematic analysis of quantitative data.

The following additional aspects must be mentioned:

- training and internal dissemination have been carried out on all the deployed topics in order to make all members of the technical staff able to take profit of the innovations;
- external dissemination has been performed presenting various papers at different international events [3],[4], [5], [6];
- Measurements have been used to keep control of the Process Improvement Program;
- last but not least, management has played a central role in the overall initiative through a continuous and intensive commitment.

With respect to the goals set at the beginning of the SPI initiative, it is possible to say that the SPI first wave has been completed on time and within budget.

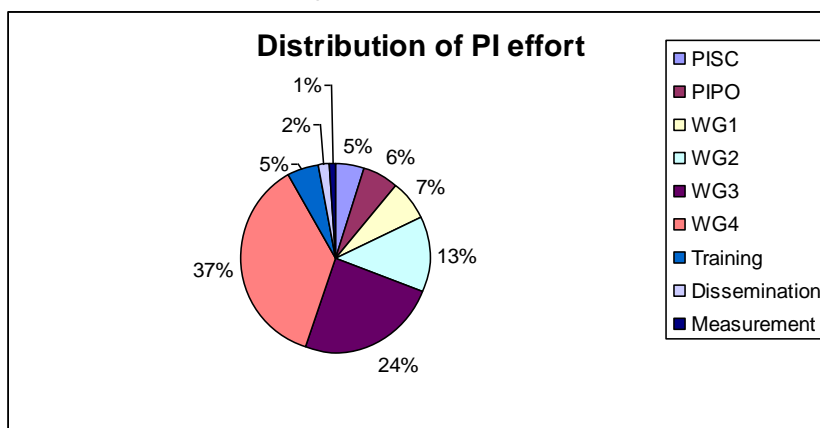
### ***Effort analysis***

The effort devoted to the SPI program has been significant, totaling an average of 9 Full Time Equivalent over the reference period, that is to say about 4% of the impacted organisational units.

Effort includes SPI management, WG activities, accompanying actions as well as SPI-driven projects (which sum up to more than half of the overall effort), but excludes deployment in projects.

Figure 7 shows the distribution of the SPI effort according to the main activities identified.

Fig. 7 – Distribution of SPI effort



## **SPI Wave 2**

### **Foundations for SPI in 1997**

At the end of the SPI first wave, the management derived the judgement that a good proportion of the initially foreseen goals had been successfully attained but that the deployment of results still needed additional work; considering also that projects are always confronting with new challenges BURM management gave the sign-off for new activities, having as foundations the following corner-stones:

- the definition of new enhancements should be kept to a minimum in order to focus on the adoption of enhanced practices in the daily routine work of all projects;
- “long-runner” activities involving a big effort shall be managed as ad-hoc projects;
- technology watch and innovation shall continue, combining process improvement with product improvements;
- analysis of quantitative data shall be made more systematic.

From an organisational point of view, the following considerations were derived:

- the management layers PISC-PIPO-WG seems appropriate;
- planning/ tracking of SPI is essential and thus PIPO staffing shall be kept;
- specific groups must be created for the various technical topics, resulting in more focused and smaller WGs (whose number has thus increased);
- management commitment ought not to diminish.

### **Improvement topics in 1997**

The SPI wave 2 topics can be subdivided in three categories:

- deployment in projects;
- long runners;
- new topics.

Each of the above is briefly explained in the following.

#### **1. Deployment in projects**

The goal of this activity is to fully deploy the enhancements defined in the daily routine work of projects. To this end, it is essential to agree with the middle management a deployment plan for each department in which pragmatic and detailed activities for the next release are listed. Such a plan (which is part of the more general “Quality Plan” for a project) will have to include details such as: which new practices will be deployed in which software area/ feature package; for each development team, the schedule for the deployment, including the provision of the required training and new hardware/ software tools; an impact analysis, for each affected area, of the introduction of enhanced practices on the overall project staffing/ schedule. This plan has to be agreed with the supervisors, Quality Management staff and senior developers.

#### **2. Long runners**

This stream of actions covers those improvement opportunities that were identified and sketched in the SPI First Wave but that require a significant amount of effort and time and thus will be managed as ad-hoc projects in the SPI second wave. They are briefly detailed in the following.

- Introduction of formal specifications languages and tools; this activity has the aim of introducing the SDL language and related CASE tools in the development activities. It is covered by the SPECS Project which is described in more details afterwards.
- Test automation; this activity, which is one of the most effort intensive of the whole SPI Program, has the following challenging goals: automated regression testing of embedded software on host environment (deploying experiences that showed to be best, through the harmonization of practices across the various groups); automated regression testing of embedded software on target environment (defining the best approach, setting-up the

environment and piloting test automation on a representative excerpt of test cases, in order to validate the approach and define the steps for test automation in the large); automated regression testing of GUI software (deploying the usage of commercial CAST tools, taking advantages of their advanced facilities for test automation); automated regression testing for fully equipped lines (defining the reference test-bed and the integration mechanisms across the tools as well as the mechanisms for ensuring reproducibility); automated regression testing for load/ stress (using commercial CAST tools as well as developing proprietary solutions and defining technical means for automatic checking of results).

- Development environment and infrastructure; the goal of this activity is the constant evolution of the development environment and infrastructure, with particular emphasis on: network infrastructure, new development environments, enhancements to configuration management environment, Intranet services and software factory evolution.
- Quantitative measurement; the goal of this activity is to strengthen quantitative measurement practices by means of: tuning of quality indicators and their application to all projects, strengthening of tools for data collection, piloting of project estimation tools (based on COCOMO, Function Points and Putnam models) and setting-up of a repository of historical data based on dynamic WWW structures.

### 3. New topics

- Customer documentation; this activity has the aim to precisely define the processes and interfaces between development teams and customer documentation teams in order to produce customer documentation.
- O-O development; this activity has the aim of defining methodological approaches and supporting tools for the extensive adoption of O-O practices in the software development projects.
- System Improvements; this activity has the aim of setting up a mechanism for channeling the experiences made by system test/ field support staff in order to benefit of their “hands-on” experience to identify and prioritize improvement opportunities at product/ system level that could result in competitive advantages with respect to the competitors.
- Planning guidelines; the goal of this activity is to package all experiences gained in SPI First Wave concerning planning, tracking and oversight in the form of a guideline to be part of the Quality Management System.

## **The SPECS Project**

The SPECS Project (SPecification Enhancements through Case and SDL) is an ESSI Project funded by the European Commission, whose main goal is the introduction of SDL based CASE tools as a support to the analysis/ design/ coding activities, thus covering one of the most important streams of Wave 2.

This Process Improvement Experiment (PIE) is felt to directly impact the following aspects:

- timeliness in analysis and design;
- fault rate in analysis and design;
- percentage of faults found in analysis and design;
- overall productivity.

The PIE should also bring to the availability of better documentation, both for internal purposes and for the customers.

These goals will be achieved through the adoption of SDL (Specification and Description Language), a formal specifications language particularly suited for specifying and describing real-time systems. SDL has been developed and standardized by ITU in the recommendation Z.100; the latest versions of the language expanded the language considerably and today SDL is a “complete” language supporting also object-oriented design by a type concept that allows specialization and inheritance to be used for most of the SDL concepts. The basic theoretical

model of a SDL system consists of extended Finite State Machines that run in parallel, one independent of each other, and communicate with discrete signals. A SDL system consists of the following components: Structure (system, block, process and procedure hierarchy), Communications (signals with optional signal parameters and channels, or signal routes), Behavior (processes), Data (abstract data types) and Inheritance (relations and specialization). The distinguishing features of SDL can be summarized as follows:

- it is an international widely accepted standard, guaranteeing long lifetime and controlled evolution as well as cross-project, cross-organization validity;
- it is an industrially proven language;
- it is specifically designed for describing complex real-time systems, where parallel activities communicate with each other through discrete events;
- it is powerful in its capabilities and user-friendly in its graphical representation;
- it is formally specified and therefore possible to analyze, simulate and translate;
- it is supported by powerful computer based tools.

SDL specifications are usually complemented by Message Sequence Charts (MSC); the MSC (defined by ITU in the recommendation Z.120, first published in 1992) is a trace language for specifying the communication behavior of real-time systems, in particular telecommunications equipment. An MSC shows, in graphical form, the sequence of messages sent between system components and their environment and is intuitively easy to use and understand. As such, it offers a powerful support for the dynamic behavior of an SDL system. Other related techniques are TTCN and ASN.1 notations. They emerge from the need of methods and tools that support verification and validation of both the standards and their current implementation. TTCN (Tree and Tabular Combined Notation, standard ISO/IEC 9646-3, X.290) is a language for the specification of tests for communicating systems that introduces the concept of abstract test suites. ASN.1 (Abstract Syntax Notation One; standard ISO/IEC 8824) is a generic notation for the specification of data types and values, particularly purposed for the description of information that is independent of the transfer format.

The adoption of SDL will be supported by the adoption of the SDT CASE tool-kit, that is developed by Telelogic (Sweden). SDT is a family of separate tools intended for the design of complex real-time systems in adherence with the SDL language and the other related notations.

The SDT tools can be used throughout the system life cycle. In the following the components that are planned to be used are briefly described.

- the SDL Editor is used for creating, editing and printing specifications using the graphical SDL notations, performing also various syntax checks at editing time;
- the MSC Editor supports creating, editing and printing of Message Sequence Charts in accordance with Z.120;
- the Analyzer performs syntactic and semantic checks of the SDL descriptions and converts also from the graphical representation (SDL-GR) to the textual Phrase Representation (SDL-PR); reverse conversion is also possible;
- the Simulator builds an executable program for understanding and debug the behavior of the system;
- the Validator can be used to build an executable program in the form of an “advanced self-exploring” simulator which helps in finding errors and inconsistencies in an SDL system and to verify the consistency with Message Sequence Charts;
- the Code Generator can be used to build applications for both host and target environments thanks to a C Code Generator, a Master Library and a pre-compiled Application Library;

- the TTCN link can be used to check the consistency between an SDL system and a test specifications expressed in TTCN.
- Other components include: Type Viewer, Cross Reference Viewer, Coverage Viewer, Abstract Data Types Library, Print Utility, Help Utility, Preference Manager, Post Master, Performance Simulator.

The adoption of SDL has been planned as a long lasting effort made up of four main steps:

Step 1 - tool selection;

Step 2 - customization for the specific environment;

Step 3 - pilot application;

Step 4 - deployment and widespread adoption.

The SPECS Project (started in March 1997 for a duration of 15 months) is identified with parts of the second, third and fourth steps and is intended to cover the following activities:

- technical set-up (procurement; installation and configuration; inclusion in software factory and storage under configuration management);
- methodology definition (drafting of guidelines focusing on: approach to be followed in using the language and the tool, granularity level to be used in specifications; strategy to be followed in cases of features affecting several development areas; definition of common interfaces; approach to be used for re-using existing code within the SDT generated frames; guidelines for interpreting the generated code, etc.);
- monitoring of the application to brand new development (technical and methodological support needed by groups adopting SDL/ SDT for the development of features consisting of brand new software code; in this case all the phases of: modeling, simulation, validation, code generation, host testing and target testing will have to be performed; crucial aspects will be: system modeling, interface description and process modeling.);
- monitoring of the application for reverse engineering (technical and methodological support needed by groups adopting SDL/ SDT for features with a limited impact within already developed software subsystems; crucial aspects will be the code generation and the inclusion of existing software code, as well as performance monitoring and regression testing);
- impact analysis (evaluation of the impacts from introducing the new approach and tools at both methodological and technical levels);
- trial for automatic test generation from SDL specifications (exploring the testing issues related with the joint adoption of SDL and TTCN. The activity will involve the evaluation of the ITEX package, the running of case studies for definition of test suites, semi-automatic test generation, generation of MSC from TTCN, debugging and maintenance activities in TTCN, running of conformance test suites in early phases of the life cycle, etc);
- alignment of quality indicators (definition of new indicators suited to the SDL/ SDT philosophy, trying to define comparability rules with previous data, if possible);
- alignment of project guidelines;
- training;
- dissemination;
- measurement;
- deployment decision;
- reporting;
- management.

## **The impacts and the experience gained**

It seems worthwhile to close the paper with a list of the most important managerial lessons that the authors think to have learnt in the running of the SPI Project.

The following aspects have been crucial for the success of the initiative:

1. the role of the improvement approaches

2. the role of the SPI organization
3. the role of senior management
4. the role of measurement
5. the role of the assessment
6. the role of case-studies
7. the role of the deployment-plan
8. the role of technology innovation
9. the role of training and dissemination
10. the role of the technical staff
11. the role of product improvements
12. the role of the customer.

Here below such aspects are dealt with in more details.

**The role of the improvement approaches** (*"Follow you, follow me"*)

SPI must be foreseen as a long term effort, since it takes several projects to define improvements, apply them, collect quantitative measures and deploy improvements in the large. In this context ISO 9000 is seen as a pre-requisite, PDCA is the driving factor, management-by-metrics is essential to check improvements from a quantitative perspective and assessments help in quantifying the status reached.

**The role of the SPI organization** (*"Eight days a week"*)

To keep the SPI program successful, a SPI program office, staffed with few full time SPI experts and moderated by a senior manager has to be set-up with the responsibility to drive with constancy of purpose the SPI program, to organize the activities and to report the status. Middle management and staff from the affected development areas has to be involved in working groups to define specific actions. A steering committee, which is made up of the senior management, has to follow up, control and drive the program, underlining its importance for the business goals of the company. Last but not least, the enthusiasm and influence of the WG Leaders is of great importance for the success of the initiative.

**The role of senior management** (*"Bridge over troubled water"*)

A 100% agreement with the senior management has to be reached on the objectives and actions; it has to drive the program and to provide sufficient budget. Moreover, the senior management has to be the sponsor of the SPI program and has to make its sponsorship visible to all the affected staff.

**The role of measurements** (*"I can't tell you why"*)

It is essential to report positive and negative results to the management as well as to the affected developers. The results of the working groups and the related quantitative measures have to be reported typically monthly.

**The role of case-studies** (*"Light my fire"*)

Case studies are a necessity for big processes and/or technology changes, on one side to test the applicability of a theory to the concrete task and to provide the needed adjustment to the existing development environment, and on the other side to convince skeptical developers on the gain that can be obtained from the activity. Therefore the most affected and most hesitant developers should play an active role in performing the case studies; after experiencing the new practices, these previously doubtful staff members become often the best advocates of the activity.

**The role of the assessment** (*"Just like starting over"*)

An assessment performed by an external, objective organization is the best starting point for a SPI program. It is very important to select the most effective and applicable improvements and to group them in short, medium and long term activities. The short term activities have to be started immediately to benefit of the momentum from the assessment.

**The role of the deployment plan** (*"The times they are-a-changing"*)



It is very important to keep a constant alignment between the rules and the project practices, especially in large software producing units, where the deployment of improvement actions is rather difficult.

**The role of technology innovations** (*"With a little help from my friends"*)

A lot of SPI activities can only be effectively performed by introducing new technology and tool support both for the development activities (CASE tools, CAST tools, etc.) and for the supporting functions (planning tools, configuration management etc.) as well as for the technology infrastructure (communication facilities, software development workplaces, etc.). When these requirements are thoroughly elaborated and justified within a comprehensive strategy, it is also easier to get the investment approval from the senior management. In fact, without strong background on the expected gains, the investment proposals will have to go through the usual long questioning and will be most probably reduced or postponed due to budget restrictions and/ or different priorities.

**The role of training and dissemination** (*"Blowin' in the wind"*)

To introduce new methods, technologies and tools requires a tailored accompanying training program to be defined. This is not only important to teach the news but also to overcome doubts on the applicability of specific actions. A general dissemination and discussion with the affected development personnel has to be performed regularly.

**The role of the technical staff** (*"We are the champions"*)

Software process improvement has to deal with processes mastered by technical staff who is very proud of his/ her work and is not willing to change his/ her habits only to follow an external guideline which would be felt as an unpleasant command and sometimes even an abuse of power. Henceforth, special care shall be paid in sowing the seeds of process improvement in the designer community and to breed the early adopters. Middle management has to be involved in the definition of activities as well as in the definition of concrete deployment plans.

**The role of product improvements** (*"The long run"*)

The increase of product size and complexity can often override the improvements made at project level. To this end, it is needed that SPI runs at a speed that is greater than the growth of the product and that, at the same time, actions must be taken to improve the product for sake of simplifying software development and minimizing source code.

**The role of the customer** (*"Stairway to heaven"*)

In SPI first wave, a key customer contributed by performing his own audit and by following up our activities. This was a big advantage since we got additional information about our customer wishes and we could increase his trust on our products. Moreover, customer focus keeps sustained the management commitment.

## Acknowledgements

Our thanks go to G. Cecchetto, Italtel BURM Research and Development Director, and D. Ongaro, Italtel BURM Director, who gave sponsorship and directions to the overall SPI Program.

The SPI Program benefited of the wisdom and experience of the Members of the Steering Committee, notably: Cecchetto, Bazzana, Di Muro, Graessle, Humml, Lora, Luciani, Maiocchi, Marino, Piacentini, Premoli, Ravasio, Roderi.

The PIPO team proved to be a major driving force of the Program and thus our thanks go to: Cordone, Nespoli, Porta, Rumi

WG Leaders also played a fundamental role in the SPI Program and thus acknowledgements have to be made to: Delmiglio, Kniessner, Monaldi, Tchorz, Scotto di Vettimo, Stomboli, Travaglini.

Last but not least, we have to thank the European Commission for the financial support given to the SPECS Project (Number 23875), run under the ESSI Initiative as part of ESPRIT

Framework IV Programme. We are especially indebted to the Project Officer, Mr. Holmes, for his continuing support.

## References

- [1] M. Mouly, M. Pautet, *The GSM System for Mobile Communications*, Europe Media Publications, 1993
- [2] T. Mehner, The Siemens Software Process Assessment Approach, in *Better Software Practice for Business Benefits*, to be published
- [3] G. Damele, S. Humml, A. Lora, D. Scignaro, G. Bazzana, G. Rumi, *Italtel's experiences in software process improvement in Better Software Practice for Business Benefits*, to be published
- [4] S. Scotto di Vettimo, A. Lora, G. Lombardi, G. Bazzana, *Usage of formal specifications languages in the development of GSM telecommunications systems*, Genie-Logiciel, 1996
- [5] G. Damele, A. Lora, *The software process improvement program at Italtel* in *Proceedings of the 1<sup>st</sup> BEST International Conference*, Brescia, November 1996
- [6] R. Delmiglio, L. Annoni, A. Manini, C. Trevisson, G. Bazzana *Test automation in telecommunication systems*, Proceedings of EuroStar 97 Conference, Edinburgh, November 1997

# MODAL: the CEGELEC Software Process Memory

Yves Benoit  
*CEGELEC, Clamart*

## Abstract

Cegelec, the number one electrical engineering firm worldwide, employs 1,400 software developers in 23 entities around the world. In 1986, Cegelec embarked on an ambitious software improvement plan which eventually resulted in the implementation of a standard development process, the MODAL methodology.

Since 1995, Cegelec development centres are evaluated based on the Capability Maturity Model (CMM) using the CBA IPI assessment method developed by the Software Engineering Institute.

With this system, the maturity of each Cegelec entity can be assessed using the same yardstick, each entity can be improved in a co-ordinated manner, and the common MODAL methodology can be enhanced through shared experience.

## 1. CEGELEC and Software

CEGELEC, the number one electrical engineering firm worldwide, employs some 28,000 people. It is the third largest company in the Alcatel Alsthom group, which numbers more than 191,000 people and has annual sales of 186 billion Francs (US \$ 28 billion).

1400 systems developers make up teams spread throughout 23 entities located in a large number of countries (USA, Belgium, Great Britain, France, Germany, India, Brazil and others). 50% are English-speaking and 50% French-speaking, and design control-command software, customised products and PLC programs.

## 2. Creation of the MODAL Common Methodology

### 2.1. Context

Given the critical nature and importance of software in the systems it supplies, between 1986 and 1989 Cegelec undertook a vast project to improve software globally throughout the company, resulting in the establishment of the company's own common methodology, MODAL.

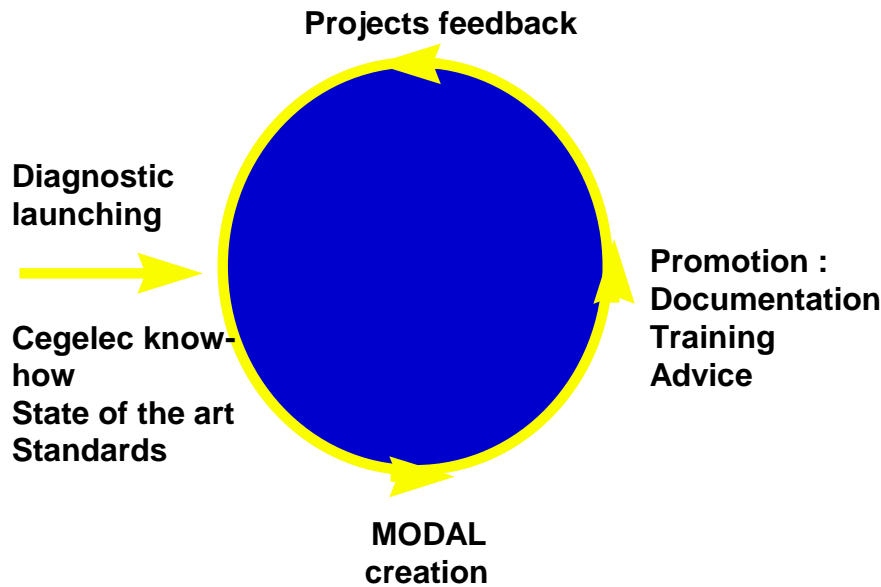
The original number of 450 systems developers has now risen to 1400; software - which was not its core business - now represents a significant part of its added value.

Software production at Cegelec is part of the wider context of production of control-command systems. MODAL therefore concerns software production and also activities both upstream and downstream, such as Specification-Development and integration of the systems of which the software forms a part.

### 2.2. Development Process

The MODAL methodology was produced as a result of the know-how of the Cegelec software engineers, the expertise of a consulting firm and also international software engineering standards such as IEEE, IEC, GAMT17 (French army standard) and the AFNOR standards (French standards association). The project was led by a team from the Cegelec research and development centre, formed especially to create MODAL and distribute it throughout the company.

Fig. YBENOIT.1 below shows how MODAL has been elaborated



**Fig. YBENOIT.1 : Life cycle of MODAL**

The improvement project was launched in September 86 by Cegelec managers to all the entities in the group and was completed in January 89. As a first step, a general audit was undertaken in order to assess the maturity of the development process in the various plants. From this base 7 working groups bringing together approximately 7 people from each of the various plants, hence some 50 people, spent 6 months on the project. Each group produced an effort of the order of 40 days per person, hence more than 10 man-years in total. The topics covered were: Specification, Development, Tests, Project Management, Quality Assurance, Configuration Management and Metrology. Pilot projects were used to validate the proposed approaches and methods.

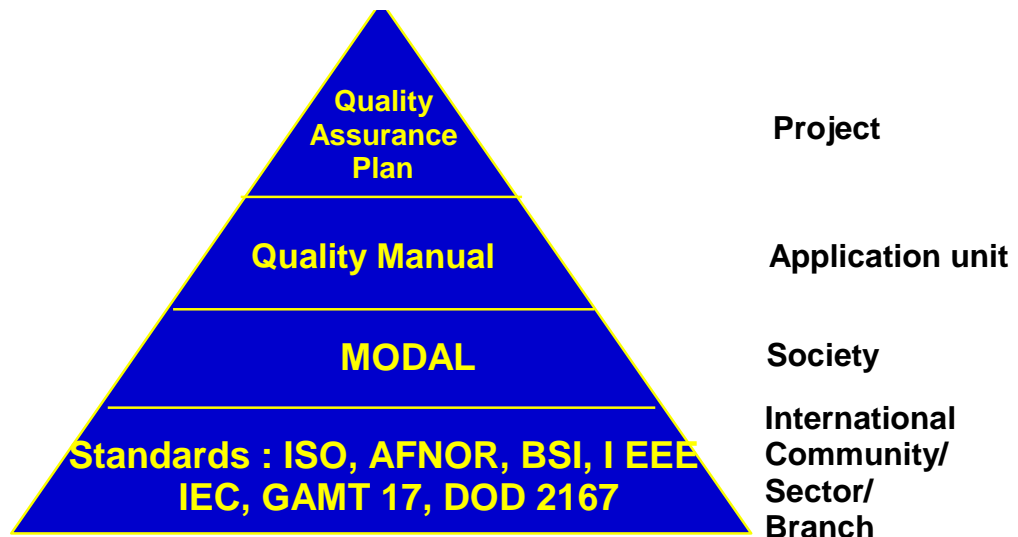
MODAL was produced to comply with the standards indicated above. Additional comparative studies have been carried out to place it in relation to the BS, DOD 2167, RGAERO 40 standards and to the CMM<sup>SM</sup> model.

The resulting products were: comprehensive documentation in English and French, a complete training course for teams and the definition of a set of tools.

### 2.3. Standard Process

The MODAL methodology, the product of the company's know-how, has thus become the standard process for software development at Cegelec. The quality manuals for the entities refer employees to it for all software development and add to it to keep track of the specific details of their business. Project leaders use MODAL and derive from it the process for their projects according to their specific requirements.

Fig. YBENOIT.2 below shows the place of MODAL in the Quality System of each unit



**Fig. YBENOIT.2 : MODAL and the Quality System**

### **3. Distribution and Development**

The methodology is supported by the team responsible for its creation.

#### **3.1. Distribution**

MODAL is distributed in the form of documentation, training, support and development tools.

- The documentation structure consists of 5 levels:
  - At level 0, general documents such as Standard and Glossary.
  - At level 1, "Procedures" which describe, for each phase of the life cycle and each business category, the tasks to be performed, incoming and outgoing products, responsibilities and checks undertaken.
  - At level 2, "Writing Rules" which specify the typical plan and content of the various headings.
  - At levels 3 and 4, "Recommendations" and "Guides" which offer help with choosing techniques and with their implementation.

Highly structured, this documentation is very simple to deal with. It exists in English and French, on paper and CD-ROM. It can be placed on the intranet for the entities.

Fig. YBENOIT.3 : below shows how the MODAL documentation is organised

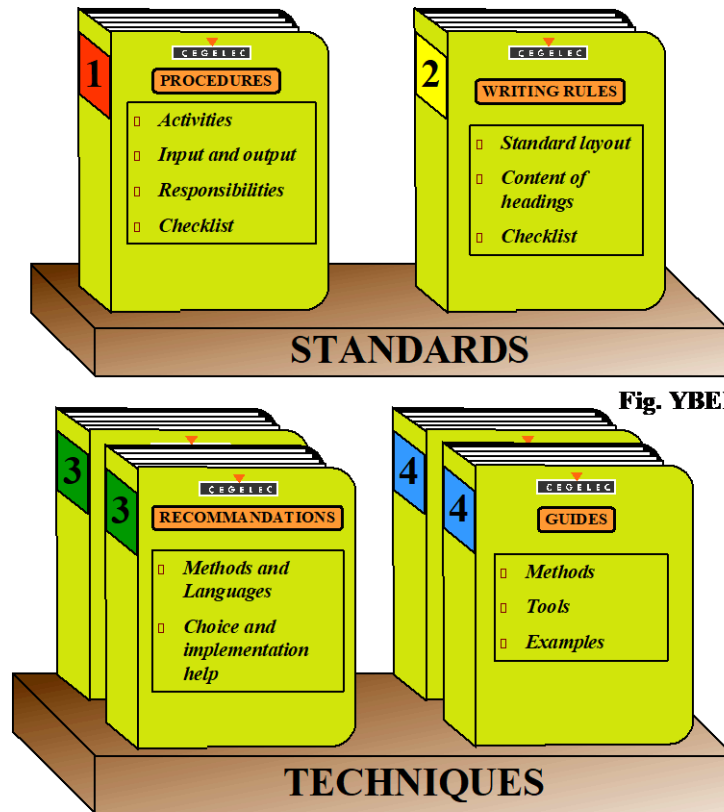


Fig. YBENOIT.3 : MODAL documentation

- The "Software Training Centre", established to distribute MODAL, is responsible for offering training courses to the various software production businesses. The various MODAL topics take up 40 training days. Courses are defined according to the profile of those taking part. On average, each person requires 15 days' training. To date 400 sessions have been held, representing more than 11,000 student days. 1,500 software developers, both internal and external to Cegelec have undergone this training. Fig. YBENOIT.4 : below shows the repartition of the effort dedicated to the training

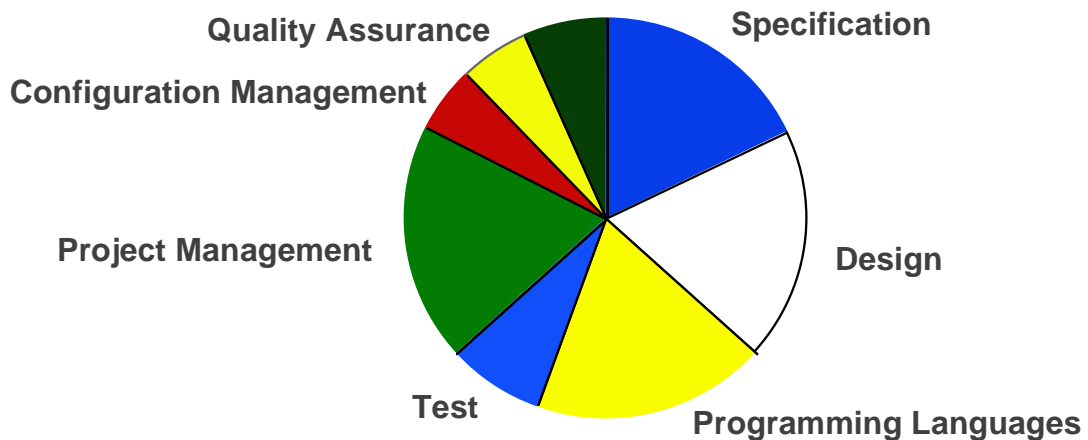


Fig. YBENOIT.4 : MODAL training courses

- Advice is offered by the consultants supporting MODAL. It is on hand for the development teams to supplement the training, for quality control engineers and the sales teams.  
Training and advice are means by which ways of improving the methodology can be detected at an early stage.
- For each phase of the life cycle, MODAL offers a set of methods, techniques and rules as well as a set of tools for automating activities, standardising the result of these activities and limiting faults due to the checks performed by these tools.

### **3.2. Developments**

Developments to MODAL result from new user needs, developments in techniques and developments in standards.

Each plant appoints a manager to be their Methodology Correspondent. All Methodology Correspondents meet approximately every 2 months and define the improvements to be made to MODAL and its distribution methods. The "MCs" are the customers of the team responsible for MODAL.

The MODAL users' club has functions similar to those of a "SPIN" (Software Process Improvement Network). It links all MODAL users, whether internal or external to the company, who wish to get together to discuss a given topic. This club is also the source of suggestions for improvements.

Users can also send requests for developments directly to the team responsible for MODAL.

Such requests are analysed, processed and go through a number of modifications including validation by the MCs. The MODAL team can also send requests for development directly to take account of development needs spotted during support actions and lastly, as a result of development in techniques and standards.

Current work concerns object-related techniques, system metrication and engineering. A major effort in the latter area, involving 4 multi-entity working groups, has led to the creation of an extended MODAL, which is currently being validated on pilot projects.

### **3.3. Results**

MODAL has demonstrated its effectiveness by assisting in the successful completion of projects of varying sizes and in particular large-scale projects such as the Channel Tunnel which called for 1100 man months over 3 years.

In 93 it was formalised by a directive from the Cegelec general management stipulating the use of MODAL in all company software development.

The process has increased in maturity in the entities, as demonstrated by the granting of ISO 9001 certification to most of the software production centres, all over the world, between 1991 and 1995.

Those with a high software component in their developments rely on certain guides for interpreting the standard:

- In France and the USA: ISO 9001 with guide ISO 9000-3
- In Belgium: ISO 9001 with guide ISO 9000-3 and ITQS certification
- In Great Britain: TickIT

Moreover, numerous licences for MODAL have already been granted both inside and outside the Alcatel Alsthom group. Interested firms have acquired these licences to improve their own development process by adapting MODAL to their context. Most commonly, this involves a mini-transfer of technology with training and support.

MODAL is a living standard development process which represents the current know-how of the company in the matter of software development.

## 4. Institutionalisation of the Improvement Process

The Cegelec board, in line with the directive from its parent company, Alcatel Alsthom, requested all its entities to carry out a self-assessment exercise using the CMM<sup>SM</sup> model from the Software Engineering Institute (SEI) for comparison and to implement an improvement programme, enabling them to reach level 3 of the model in stages.

### 4.1. Adoption of the CMM Model

Cegelec has been using the CMM model and the associated evaluation method CBA IPI (CMM Based Appraisal for Internal Process Improvement) since 1995, as a result of which it is able to determine the maturity of its development centres, as well as any improvements to be implemented on a local level and to MODAL.

The model and the evaluation method used are available in English and in French. The latest version, recognised by the SEI, is the result of a translation undertaken by ASEC of Montreal and validated by a number of French companies. The evaluations are carried out in accordance with the SEI directives and led by authorised "Lead Assessors", who are Cegelec employees.

Aimed initially at the French-speaking entities in the company, the evaluations will gradually be extended to those in English-speaking countries. Evaluations of the eleven entities with more than 40 software developers are due to take place between 1995 and 1998: one in 1995, two in 1996, four in 1997 and six in 1998.

Five Cegelec entities have up to now been assessed using the CBA IPI and 2 more will be evaluated before the end of the year.

### 4.2. Approach to Improving the Process, at the "Corporate" Level and at the Level of Each Plant

The "corporate" SEPG was created in January 95, based on the team responsible for MODAL. Apart from developing MODAL and distributing it via the Software Training Centre, it is also currently responsible for conducting the CMM evaluations of the various entities and assisting them to draw up and implement their improvement programmes.

Software Improvement Groups (SIG), corresponding to the local entity SEPGs, are either merged with the quality control groups, or distinct entities, depending on the breadth of the improvement programmes. They are the priority contacts for the "corporate" SEPGs. Contact is made periodically through the Methodological Correspondents (MC).

Fig. YBENOIT.5 : below the relation between the SEPG corporate and the SEPG local.

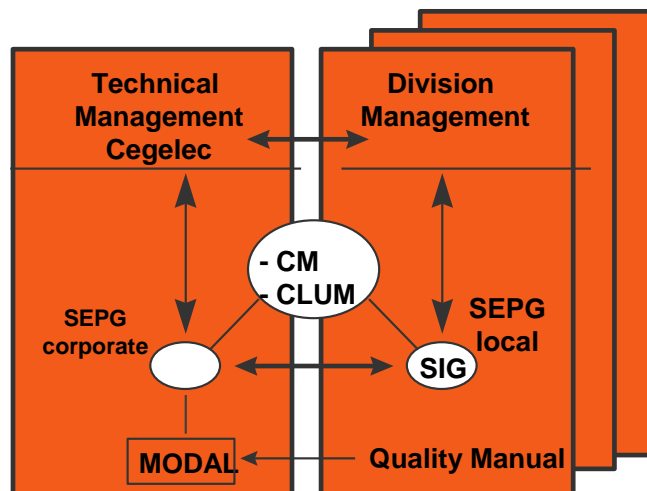


Fig. YBENOIT.5 : SEPG corporate and the SEPG local.



Cegelec has decided to create its own group of "Lead Assessors", authorised by the SEI, who are capable of overseeing the assessments of its software development centres. This group, which is part of the "corporate" SEPG, was formed by a transfer of technology undertaken in French by ASEC with the approval of SEI.

For their assessments, the "lead assessors" rely on teams composed equally of internal staff and external observers in the entity under evaluation.

Being attached to the "corporate" SEPG, they are in a position to contribute to the generalisation, by way of the MODAL common methodology, of the process features identified in the various entities.

It should be noted that the CBA IPI method, which offers an exact picture of the strengths and weaknesses of the entity under evaluation and creates a real dynamic for improvement, reinforces the credibility of the assessors and hence the SEPG in the eyes of the entity. It therefore creates good conditions for collaboration between the "corporate" SEPG and the entity concerned, when it comes to drawing up and implementing its improvement programme. The "corporate" SEPG is thus in a better position to co-ordinate improvements in all the entities.

On the other hand, the internal "lead assessor" must be sufficiently knowledgeable to convince certain sponsors that putting an improvement programme in place is more important than gaining a level. The final mark, which is a simple piece of arithmetic, possibly even too simple, should probably be weighted by an indicator of how fast the maturity of the process is improving.

The "lead assessors" should also consciously retain the detachment which makes it easier to assess widely varying outside companies. In this spirit, they can also get involved in companies not belonging to the Cegelec group.

### **4.3. Enhancements to MODAL**

This approach has made it possible to enhance MODAL by comparing its practices to the CMM model.

The assessments undertaken by the "lead assessors" belonging to the corporate SEPG can be used to identify channels for improving MODAL as well as "best practices" derived from the entities which also result in its enhancement.

The improvement programmes, resulting from the assessments, also lead to the MODAL methodology being better positioned in each organisation, even though it was initially centred primarily around the projects.

### **4.4. Difficulties to overcome**

It is difficult to show quantitative return on investment before CMM level 3, in particular when project manager don't have enough time to draw up the project balance at the end of the project.

Furthermore, the training needs are often underestimated by the middle-management.

That's why the commitment of executive management is prerequisite to initialise and support the process improvement.

It's necessary for all levels of organisation management and staff to commit themselves and to set aside the unavoidable conflicts between various engineering groups so that all can share the objective of improving the company's process.

### **4.5. Investment Corresponding to the Improvement Exercise**

The "corporate" improvement effort, devoted for the past ten years to improving the process, corresponds to a direct investment of 90 MF (15M US \$). This includes the setting up and development of the methodology, consultancy and training and, when assessments are undertaken, the cost of the external assessors at the site being assessed.

This does not include the cost of the assessed personnel nor the assessors originating from that site. Nor does it include the improvement efforts of each entity, tied to its own quality control system and hence to its own budget.

In effect the group has committed itself to an investment of at least 140 MF (24M US \$) over 10 years.

Investment for the years to come places emphasis on assessments and the associated improvement programmes, at both "corporate" and entity level.

## 5. Conclusion

By an approach of continuous improvement, CEGELEC has been able to establish a common corporate culture of using MODAL and to keep a firm hand on its software development. Assessments, conducted by authorised "lead assessors", provide CEGELEC with the means to draw up a coherent report on the maturity of its entities and to implement improvement projects to suit each of them, without moving away from the company's common process software.

## Références

- [1] Richard Basque, Une série de trois articles sur le CMM-SM et sur la méthode CBA-IPI, in: *INFO-QUEBEC, revue de la FIQ (Fédération de l'Informatique du Québec)*, 1996.
- [2] Barry W. Boehm, *Software Engineering Economics*, Prentice Hall, 1981.
- [3] Dunaway, Donna K., and Masters, Steve, *CMMSM-Based Appraisal for Internal Process Improvement (CBA IPI): Method Description*, CMU/SEI-96-TR-007, April 1996.
- [4] F.P. Brooks, *The mythical man-month*, Addison Wesley, 1982.
- [5] CEGELEC, *Standard for software Production*, , ref BCI 50 005a-en, 1991.
- [6] IEEE Software Engineering standards, Collection IEEE 1991
- [7] ISO 9001, *Systèmes qualité, modèle pour l'assurance de la qualité en conception/développement, production, installation et soutien après la vente*, 1994.
- [8] Mark C. Paulk, Bill Curtis, Marie Beth Chrisis, Charles V. Weber, *Modèle d'évolution des capacités V1.1*, french version, CMU/SEI-93-TR-24.- ESC-TR-93-177, 1993  
*Pratiques du Modèle d'évolution des capacités logiciel, V1.1*, french version, CMU/SEI-93-TR-25.- ESC-TR-93-177, 1993.
- [9] P. Nguyen-Duc.- *Complexité du logiciel : des modèles pour la maîtrise du processus*. - Paris : *Revue des télécommunications*, 2<sup>ème</sup> trimestre 1994

# Evaluating Software Product Quality

Jørgen Bøegh

DELTA Software Engineering, Hørsholm, Denmark

## Introduction

Software quality evaluation is the systematic examination of the extent to which the software is capable of fulfilling specified quality requirements.

This paper discusses the needs for and advantages of software product evaluation. It reviews the relevant international standards in the area, covering both published and forthcoming standards.

Practical software evaluation schemes have been introduced in recent years. Some of these schemes are described and their market acceptance reviewed. Finally, the experiences gained with the MicroScope evaluation scheme based on almost 80 commercial evaluations are presented.

## Market Needs

The use of software grows dramatically and so does the number of critical computer systems. Faults in critical systems may lead to serious consequences. Therefore the quality of the software of these systems is important, both for individuals, for companies and for the society in general. This leads to a growing demand for quality evaluation of software products.

### *Critical application*

Obviously most software quality evaluations are conducted in the area of critical applications. This includes national critical applications like defence systems, where very large software developments are carried out. Here the evaluation effort is often of the same magnitude as the development effort.

Until now human life critical systems have been the main target for independent third party quality evaluations. Such systems include traffic control systems, medical systems, process control systems, robots etc. Also for this type of applications the evaluation effort can be very large. Often public authorities require independent evaluations of such systems.

Other systems are equally critical. The modern society depends on such software systems. Examples are electronic payment systems, public administration systems and telephone systems. There are equally high quality requirements to this class of systems and quality evaluations are also extensive.

Similarly the class of corporate critical systems like production systems, financial systems, consumer products including software and customer databases should be considered as systems with high quality requirements. However, this type of software is often neglected with respect to quality evaluation. As companies realize the advantages of ensuring the quality of their software, the market for software evaluations could increase substantially.

### *Market advantages*

At the moment most independent quality evaluations are done because they are required by law or public authorities. There are, however, other good reasons for demanding software

evaluations.

In some cases a software developing company may be requested by an acquirer to agree as part of the development contract to accept an independent quality evaluation. This can actually be an advantage for both parties since disputes about the delivered software can be referred to the evaluation and thereby legal actions can be avoided.

Some evaluation schemes are devoted to issuing quality marks or seals. The aim is also here to give a marketing advantage to software products of good quality. Several quality marks have been introduced but until now they have not been generally accepted in the market.

Yet another group of evaluation schemes are used for comparing similar software products. Many software magazines apply this approach for benchmarking software packages, but it is also relevant in other circumstances, for example in an acquisition situation when choosing between different suppliers of software. In any case a quality seal could surely have a positive influence on a buying decision.

## **International Standardisation**

The market need for quality evaluations is also reflected in the standardisation activities. Currently several standards in this area are either being finalised or being revised to fit with the present state of the art. The requirements of the standardisation work and the most important standards are presented here.

### ***Evaluation requirements***

By definition, quality evaluation is the systematic examination of the extend to which an entity is capable of fulfilling specified requirements. Hence software product evaluation must follow some strict rules and satisfy some basic requirements. Otherwise evaluation results will not be valid and hence not recognized. This is true in particular for independent third party evaluation. ISO Guide 25 states requirements for testing laboratories and emphasizes the following requirements for evaluation:

- **Repeatability:** Repeated evaluation of the same product to the same evaluation specification by the same evaluator shall give the same result.
- **Reproducibility:** Repeated evaluation of the same product to the same evaluation specification by different evaluators shall give the same result.
- **Impartiality:** Evaluation shall be free from unfair bias towards achieving any particular result.
- **Objectivity:** The evaluation shall be obtained with the minimum of subjective judgement.

These requirements must be fulfilled by any reliable evaluation scheme. In addition there may be other considerations such as cost effectiveness of the evaluation, inclusiveness (the evaluation covers all quality characteristics) and indicativeness (when some discrepancies or other problems are found by the evaluation, their causes and required actions are indicated).

### ***ISO/IEC 9126: Quality characteristics***

The international standard ISO/IEC 9126 is the relevant standard for defining software quality and is recommended to be used for quality evaluation in most situations. In ISO quality is defined as “the totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs”. ISO/IEC 9126 suggests a hierarchical quality model with six quality characteristics and attached subcharacteristics:

**Functionality:** A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs. Subcharacteristics are suitability, accuracy, interoperability, compliance and security.

**Reliability:** A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time. Subcharacteristics are maturity, fault tolerance and recoverability.

**Usability:** A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by stated or implied set of users. Subcharacteristics are understandability, learnability and operability.

**Efficiency:** A set of attributes that bear on the relationship between the level of performance of

the software and the amount of resources used, under stated conditions. Subcharacteristics are time behaviour and resource behaviour.

**Maintainability:** A set of attributes that bear on the effort needed to make specified modifications. Subcharacteristics are analysability, changeability, stability and testability.

**Portability:** A set of attributes that bear on the ability of software to be transferred from one environment to another. Subcharacteristics are adaptability, installability, conformance and replaceability.

ISO/IEC 9126 is applicable for most software applications. However, in some specific situations it may be better to use another quality model, for example for security evaluations.

### ***ISO/IEC 12119: Quality requirements and testing***

This standard is based on the German standard DIN 66285. It is applicable to software packages. It establishes a set of quality requirements and it provides instructions on how to test a software package against these requirements (instructions for testing, in particular for third part testing). In contrast with ISO/IEC 14598 it only deals with software packages as offered and delivered. It does not deal with their production process including development activities and intermediate products like specifications and source code. ISO/IEC 12119 uses ISO/IEC 9126 as the underlying standard for defining quality of a software product.

### ***ISO/IEC 14598: Software product evaluation***

ISO is currently preparing a new standard for software product evaluation. It is also intended to be used in conjunction with ISO/IEC 9126. The new standard is a multipart standard consisting of the following parts:

**Part 1: General overview** - This part provides an overview of the other parts and explains the relationship between ISO/IEC 14598 and the quality model in ISO/IEC 9126. It defines the technical terms used in the standard, it contains general requirements for specification and evaluation of software quality and it clarifies the concepts. Additionally, it provides a framework for evaluating the quality of all types of software products and it states the requirements for methods of software product measurement and evaluation.

**Part 2: Planning and management** - This part provides requirements and guides for a supporting function responsible for the management of software product evaluation and for technologies necessary for software product evaluation. The responsibilities of this supporting function include people motivation and education relevant to the evaluation activities, preparation of suitable evaluation documents, standards and responding to queries on evaluation technologies. The main targets for evaluation support are the software development and system integration projects, which include software acquisition, both at a project and organisation level.

**Part 3: Process for developers** - This part provides requirements and recommendations for the practical implementation of software product evaluation when the evaluation is conducted in parallel with the development and carried out by the developer. The evaluation process described defines the activities needed to analyse evaluation requirements, to specify, design, and perform evaluation actions and to conclude the evaluation of any kind of software product. The evaluation process is designed to be used concurrently with the development. It needs to be synchronised with the software development process and the entities be evaluated as they are delivered.

**Part 4: Process for acquirers** - This part contains requirements, recommendations and guidelines for the systematic measurement, assessment and evaluation of software product quality during acquisition of off-the-shelf software products, custom software products, or modifications to existing software products. The evaluation process described helps to meet the objectives of deciding on the acceptance of a single product, or for selecting a product from among alternate products. The evaluation process may be tailored to the nature and integrity level of the application. It is also sufficiently flexible to accommodate the wide range of forms and uses of software in a cost-effective manner.

**Part 5: Process for evaluators** - This part provides requirements and recommendations for the practical implementation of software product evaluation when several parties need to understand, accept and trust evaluation results. The process described defines the activities needed to analyse evaluation requirements, to specify, design and perform evaluation actions and to conclude the evaluation of any kind of software product. The evaluation process may be used to evaluate already existing products, provided that the needed product components are available, or to

evaluate products in development. This part may be used by testing laboratories when providing software product evaluation services.

**Part 6: Documentation of evaluation modules** - This part describes the structure and contents of an evaluation module. An evaluation module is a package of evaluation technology for a specific software quality characteristic or subcharacteristic. The package includes descriptions of evaluation methods and techniques, inputs to be evaluated, data to be measured and collected, acceptance criteria, and supporting procedures and tools. This part should be used by testing laboratories and research institutes when developing evaluation modules.

## **Practical Evaluation Schemes**

There are a number of practical software quality evaluation schemes in use. Some of these are outlined here. The selection is not exhaustive, but gives an impression of the trends in the area.

### ***The SCOPE experiment***

The ESPRIT project SCOPE (Software CertificatiOn Progamme in Europe) was the first major international attempt to set up a certification scheme for software product evaluation [1]. SCOPE lasted from 1989 to 1993. The project involved 13 companies from 8 countries with a total effort of 110 person years.

The SCOPE project was successful although it failed to set up a certification scheme. The main achievement was that SCOPE developed a framework for software quality evaluation which is now widely accepted and used as a basis for evaluations around the world.

The main results of SCOPE was an evaluation method, a collection of evaluation technologies, and extensive practical experience.

The SCOPE project carried out 30 trial evaluations [2]. These case studies were conducted in two phases. In the first phase six evaluations were carried out applying different evaluation procedures and techniques. The results were analysed and used for planning the second phase of case studies. Here all evaluations followed the same evaluation procedure and care was taken to select software products for evaluation that covered a wide range of applications and software development approaches.

The SCOPE evaluation method was developed as a result of an analysis of the trial evaluations. It was documented in the "Evaluators Guide" which was submitted to ISO for consideration [3]. This document has now been adapted and published as ISO/IEC 14598-5: Process for evaluators.

The concept of evaluation modules was also an important outcome from SCOPE [4]. It was introduced to make it easy and flexible to manage the use of the different evaluation technologies.

### ***MicroScope evaluations***

In Denmark the MicroScope approach to software evaluation was introduced by DELTA Software Engineering in 1991 [5]. MicroScope is based on the results of the SCOPE project and follows the standards ISO/IEC 9126 and ISO/IEC 14598.

The MicroScope evaluations are being used in a number of situations. The most common purposes are to state the conformance to a specified external standard or regulation and to validate that the level of documentation and safety for a software product is satisfactory.

The evaluations are based on an agreement between a client and DELTA on which quality characteristics of the software product should be considered and which evaluation modules should be used. The MicroScope evaluation modules are checklist based. There are 12 modules covering all six characteristics of ISO/IEC 9126. The evaluations are performed at one of four possible levels for each relevant characteristic corresponding to the criticality of the product.

MicroScope emphasizes the evaluation of the workmanship of the software and related documentation, i.e. that design descriptions, coding standards, test documentation etc. comply with the best state of practice in the software industry.

At this time almost 80 commercial evaluations have been conducted following the MicroScope approach.

### ***TÜV Nord evaluations***

TÜV Nord in Germany has developed an evaluation method mainly aimed at process control

and real time systems with safety relevance. The evaluation method is based on several standards including IEC 880, draft IEC 1508 / IEC 65A, DIN V VDE 801 and DIN 19250. TÜV Nord received an accreditation by DEKITS as a software testing laboratory for evaluating software according to these standards.

Off-the-shelf software is evaluated on the basis of ISO/IEC 12119. TÜV Nord is accredited as a testing laboratory by the Gütegemeinschaft Software Association.

The mentioned standards are mainly concerned with functionality. TÜV Nord is also elaborating quality profiles using e.g. ISO/IEC 9126. They have been involved in national and European research projects which have resulted in the adoption of new methods and tools like TASQUE [6], CATS [7], and SQUID [8]. These tool are used to enhance the evaluation capabilities.

### ***The ASSESPRO prize***

In Brazil the Technological Center for Informatics Foundation (CTI) is in charge of a major effort to provide software product evaluation services to the Brazilian software industry [9], [10], [11]. They have developed the method MEDE-PROS based on the international standards ISO/IEC 9126, ISO/IEC 12119 and ISO/IEC 14598 drafts.

The method is checklist based and has several similarities with the MicroScope approach. The checklists are continuously being improved and includes now more than 100 questions. It evaluates the “product description, documentation and programs and data” according to ISO/IEC 12119. The main emphasis of MEDE-PROS evaluations are on functionality and usability for software packages.

The evaluation method is applied by ASSESPRO (The Brazilian Association of Software Houses) for awarding the “Best Software Product of the Year” in Brazil. The ASSESPRO prize only includes software packages and is given in six categories:

- Systems for documentation and planning support
- Systems software and systems of support to software development
- Tools for graphical design
- Information and services automation systems
- Engineering, scientific and industrial automation systems
- Education and entertainment systems

Each year since 1993 between 20 and 50 software packages have been evaluated for the ASSESPRO prize and a considerable statistical material have been collected.

Currently the MEDE-PROS evaluation method is also being applied to support a Brazilian software export initiative with the aim of increasing the Brazilian share of the world market.

### ***The SQUID approach***

The SQUID approach to software quality evaluation is slightly different. It is intended to be used during the software development as described in ISO/IEC 14598-3 [12]. The aim of the SQUID method is to provide support to a software developer. It is an approach to modeling, measuring, and evaluating software quality during the development process. SQUID is supported by a toolset currently under development.

The toolset assists in quality specification, quality planning, quality control and quality evaluation. More specifically, for quality specification it provides the means to establish targets for the product quality requirements and evaluate their feasibility. Then, the toolset supports the identification of internal software product and process attributes that must be controlled during the development process to fulfill the project quality requirements. This is called quality planning and control. Finally, the toolset helps to assess the fulfillment of project quality requirement.

Ongoing evaluation of the SQUID approach and toolset is part of the work [13]. One particularly interesting experiment is to apply SQUID as a supporting tool for a third party testing laboratory in connection with independent software product evaluations.

### ***Other initiatives***

Several other attempts to develop quality certification schemes and seals have been done in different countries during the last ten years.

One of the first initiatives to develop a quality seal for software products was the German GGS

controlled by the Gütegemeinschaft Software Association [14]. The GGS association was founded in the mid eighties with the aim to define quality criteria for software products and to organize a software quality certification scheme. This resulted in 1990 in the publication of the German standard DIN 66285 which defines the quality requirements for a software package and specifies the testing procedure that could lead to a certificate. This standard was adapted for international standardization and published by ISO in 1994 as standard ISO/IEC 12119.

The GGS seal has never been really successful in Germany and only few software products have been awarded the GGS seal.

In 1996 the French national standardization body AFNOR initiated the development of a marking of software products called NF Logiciel [15]. This quality mark should be applicable to any type of software products. It is also based on the standard ISO/IEC 12119 and requires a product to be composed of “product description”, “user documentation” and “program and data”. For a software product to obtain a NF Logiciel marking it requires that the claims in the “user documentation” can be verified in the “program and data” by an independent evaluator. Several trial evaluations were started, but the quality mark has not yet been adopted in France.

In Italy an initiative to implement an evaluation scheme for software based on ISO/IEC 9126 called Q-Seal was initiated. It applies a predefined profile based on characteristics, sub-characteristics and levels. Some case studies were conducted in 1995-96.

The National Computerization Agency in Korea started in 1996 to set up a software product evaluation scheme. They decided from their organization’s perspective to concentrate on “custom-made software” and they are following an approach to integrate software product evaluation and software process evaluation into a common framework. The process evaluation is based on the software life cycle processes defined in ISO/IEC 12207.

Finally the Swedish Association of Software Houses SPI (Föreningen Svensk Programvaruindustri) has prepared an annotated translation of the standard ISO/IEC 14596-5 [16] and also developed a series of ‘small’ evaluation modules. A first experimental software product evaluation was successfully completed in 1997.

## **Support for Evaluation**

In order to help companies involved in software evaluation some supporting actions have been established. This includes the EuroScope network of testing laboratories and the ESSI-SCOPE dissemination project supported by the European Commission.

### ***EuroScope***

Some of former SCOPE partners and other interested parties work together in the EuroScope network. The purpose of the network is to collaborate on marketing activities and knowledge transfer. A longer term goal is to develop a scheme for mutual recognition of evaluation results.

### ***ESSI-SCOPE***

ESSI-SCOPE is a project under the European Commission ESSI programme on software best practice. This project addresses a range of topics related to software product quality such as the treatment of software as a corporate asset, the existing and emerging standards related to software products, software product quality characteristics and their application in practice, approaches to the evaluation of software products, process improvements recommended to achieve greater product quality, the results of evaluation work in Europe, and tools and techniques available for software evaluation.

The project carries out a range of dissemination activities. This includes a newsletter, a World Wide Web service, an electronic mailing list, an ftp service for documents, workshops across Europe, and a European conference.

## **MicroScope Experiences**

MicroScope is an example of a commercially successful software evaluation scheme. A considerable number of evaluations have been conducted since its introduction in 1991. The accreditation obtained in 1996 confirmed the soundness of the scheme.



### **Evaluation procedure**

The MicroScope evaluation procedure consists of five activities which are conducted in cooperation with the client of the evaluation. The activities are performed on the basis of data and other information provided by the client or produced by other activities during the evaluation. The following figure shows the evaluation process.

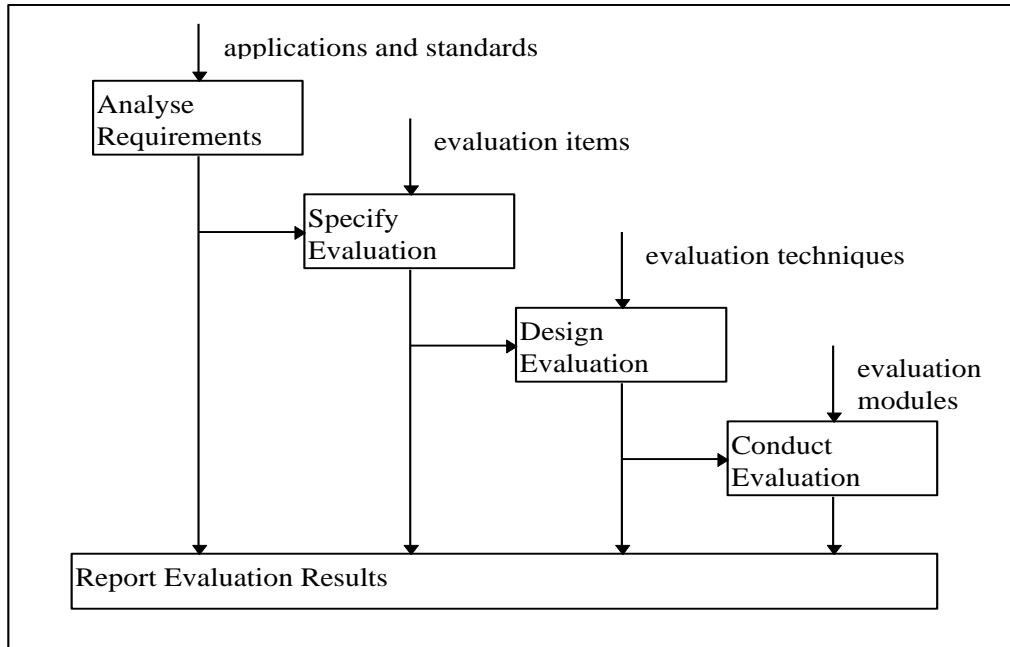


Fig. 1: The evaluation Process

The first activity is the analysis of evaluation requirements. Here the actual evaluation requirements are established.

The second activity is the specification of the evaluation which produces an evaluation specification on the basis of the evaluation requirements and the description of the product provided by the client.

The third activity is the design of the evaluation which produces an evaluation plan on the basis of the evaluation specification, the components of the software product to be evaluated and the evaluation methods proposed by the evaluator.

The fourth activity is the execution of the evaluation plan which consists of inspecting, modelling, measuring and testing elements of the product according to the evaluation plan. The actions performed by the evaluator are recorded and the results obtained are put in a draft evaluation report.

The final activity is the conclusion of the evaluation which consists of the drafting and approval of the evaluation report and the disposal by the evaluator of the product components evaluated.

The MicroScope evaluation procedure complies with the international standard ISO/IEC 14598-5: Process for evaluators.

### **Commercial evaluations**

DELTA Software Engineering has obtained a considerable experience by offering the MicroScope evaluation service on commercial conditions. MicroScope was launched in 1991 and until now about 80 software evaluations have been conducted.

The evaluations cover a large number of applications areas. This includes fire alarms, burglar alarms, off shore systems, gas burners, railway signals, process control systems, medico systems, automatic weighting systems, and windmills. The largest number of evaluations have been in safety critical areas. The following table shows the effort used on different categories of evaluations.

Application category	Effort
Off shore	48 %
Fire alarms	23 %
Railway	12 %
Other	17 %

The off shore evaluations weight quite heavily in this summary due to the size of these evaluations. In numbers they account for less than 10 % of the evaluations.

One of the main experiences is that the software developers learn from their evaluation experiences. It is very clear that the second time a company applies for an evaluation the quality of their software is higher than the first time. The following table indicates the quality of software that has been evaluated under the MicroScope approach.

Product quality	Product
Very high	25 %
High	35 %
Medium	25 %
Low + Very low	15 %

In particular the quality of the development documentation has increased. A few years ago companies seemed to produce very limited documentation whereas now it is common to see extensive and good quality documentation such as design documents and well documented source code.

### ***Evaluation modules***

MicroScope includes a set of 12 evaluation modules. They are checklist based and comprise more than 1800 questions. A disadvantage of the checklist approach is that answers to individual questions often rely on the judgement of the evaluator. This loss of objectivity can be minimized by carefully formulating the questions so that they can be answered unambiguously and by avoiding very short checklists. Some experiences with checklist based evaluations are reported in [17].

The possible answers to the checklists are such that 0, 1, and 2 points are given, or it is decided, that the answer is not applicable (N/A). The answer of 2 indicates that the feature is present, and the formulation is such, that this is positive for the product. When 0 points is given, it implies an absence of a desired feature.

For each checklist a score is calculated by counting the number of points given, and the total number of points that the product could have received when excluding the N/A questions. The ratio between these two numbers is taken and multiplied by 10, to arrive at a final score for a checklist between 0 and 10, independent of the number of questions on the checklist.

The evaluation modules cover all quality characteristics identified in ISO/IEC 9126. The following table categorises the evaluation modules according to quality characteristics.

Quality characteristic	Evaluation module
Functionality	Requirements specification Safety Test documentation prEN 54 (fire alarms) NAWI (nonautomatic weighting instr.)
Reliability	Reliability
Usability	User manual ISO/IEC 9241
Efficiency	Source code
Maintainability	Design documentation Source code
Portability	Source code

In practice, the application of some evaluation modules may be irrelevant for an evaluation, and in other cases it may be convenient to restrict the scope of application of some of the chosen evaluation modules to samples from the documentation received for evaluation.

In any case, in order to keep the evaluation within reasonable limits for time and cost, an agreement must be reached concerning the choice of evaluation modules and the depth to which they are used.

Before starting an evaluation, the evaluator must ensure that the needed product information is made available by the client. The specific product information requirements are stated in the evaluation modules.

### ***An evaluation example***

To give an example, a MicroScope evaluation of the software part of a fire alarm, which is a typical example of a small evaluation, will require in the order of one to two weeks effort. Such an evaluation will usually require the application of five evaluation modules:

- Usability - User manual
- Maintainability - Design documentation
- Functionality - Safety
- Functionality - prEN 54

The purpose of applying the last evaluation module is to demonstrate compliance with the standard prEN 54: Components of automatic fire alarm detection systems: Part 2: Control and indicating equipment. It should be emphasized that the software evaluations are performed in conjunction with thorough hardware and system tests of the fire alarms.

In a real MicroScope evaluation (here anonymized) conducted at DELTA Software Engineering the following scores were achieved and observations made:

#### Usability - User Manual: Score 8.6

Observations: The user manual is well structured, with the appropriate level of details for users who have been trained in the operation of the system. On the other hand there is lacking an upper level description of the system including examples of typical systems. There is no identification of the specific software version for which it is relevant. Of the questions on the checklist, 25% were N/A, because the system is embedded, and the user manual therefore does not need to cover explanations of platform, operating system, software environment, back-up etc.

#### Maintainability - Design documentation: Score 7.5

Observations: The modularity and breakdown of the system are very good, and it is a convincing reflection of the implementation. In addition, the completeness and consistency of the design documentation is very good. On the other hand, a simple introduction to the system and its design is lacking, together with descriptions of data structures and explanations of variables and constants. The self-descriptiveness of the design documentation is weak, as illustrated through lack of consistent document identification, tables of contents, glossaries, and introductions.

#### Functionality - Safety: Score 7.3

Observations: The selftest facilities and the supervision of the hardware are good from a safety point of view, as well as the user interface. The programming style is well structured. On the other hand, the documentation of fault handling is weak, and is mainly restricted to information contained in module headers, so there is no central place where it is identified what may happen, and what the systems reactions are. Use of interrupts complicates the safety analysis, as does the use of the language C. Of the questions on the checklist, 25% were N/A, because they were related to fail-safe features. No true fail-safe features are present in the software, but because of the application, they are not needed.

Functionality - prEN 54: Here (PASSED/FAILED) is given for each of 16 requirements from the standard. In this case 15 out of 16 requirements from prEN 54 were PASSED.

### Conclusion of the evaluation

The software and the corresponding documentation for the XXX fire alarm unit have been assessed with regard to its conformity with the relevant requirements of prEN 54. This has been done by conducting a MicroScope evaluation to assess the design documentation, the user manual, the safety features of the software system, as well as the conformity with prEN 54.

The conclusion is that the software is in sufficient conformity with the standard for use in a fire alarm unit. With regard to the nonconformity found with one of the requirements of prEN 54, it should be noted that the judgement is based on the fact that no documentation was found for the requested feature. If the software is in conformance, this may be documented. If the nonconformity is real, and the matter is deemed to be sufficiently important, restrictions may be introduced, so that the system can only handle 512 fire detectors and/or manual call points.

### **Accreditation**

DELTA Software Engineering received an accreditation according to EN45001 in the start of 1996 covering MicroScope evaluations. The accreditation confirms the compliance of MicroScope with the standards ISO/IEC 9126 and ISO/IEC 14598-5 (draft).

In order to achieve the accreditation, a quality documentation consisting of 39 documents totaling 2100 pages was produced. The documentation includes a quality system, the relevant standards, the operating procedures and test instructions. The experiences with the accreditation process were positive. The accreditation body handled the process fast and efficiently. It took six months from the application was forwarded to the accreditation was issued.

### **Licenses**

The MicroScope evaluation method and evaluation modules have been licensed to companies in Greece and Hungary and other companies have expressed their interest. Such arrangements provide an efficient start-up of software evaluation services for testing laboratories which are new in this field.

### **Conclusion**

As the number of critical software applications grows the need and demand for software quality evaluation increase. International standards are being prepared to support evaluation and practical software product evaluation schemes are available to the market. However, the field is not mature yet, and there is still a need to experiment, collect experiences, and to improve the evaluation methods and technologies.

### **References**

- [1] Robert P., Roan A., The SCOPE Project: an Overview, in: *Approving Software Products*, pp. 9-22, Ehrenberger W. (Editor), Elsevier Science Publishers B. V., The Netherlands, September 1990
- [2] Welzel D., Hausen H.-L., Bøegh J., A Metric-based Software Evaluation Method, in: *Software Testing, Verification, and Reliability*, pp. 181-194, John Wiley & Sons, Vol. 3, 1993
- [3] Bøegh J., Hausen H.-L., Welzel D., Guide to software product evaluation - The evaluator's guide, *SCOPE Technical Report SC.92/099/ECT.jb. GMD.hlh.dw/T2.1.2/DR/04*, 1992
- [4] Bøegh J., Evaluation Modules: The Link between Theory and Practice, in: *Proceedings of the Second IEEE International Software Engineering Standards Symposium*, pp. 170-174, IEEE Computer Society Press, 1995

- [5] Kyser H., MicroScope, the Evaluation of Software Product Quality, *DQD-5012200*, DELTA 1995
- [6] Anders U., Flor R., TASQUE-TÜV, in: *Software Metrics for Product Assessment*, pp. 198-200, Bache R., Bazzana G., McGraw-Hill, London, 1994
- [7] Werkzeuge fuer den standardisierten Software-Sicherheitsnachweis (SOSAT-3). *Tectnicher Fachbericht*, Hamburg, March 1993, supported by BMBF (Bundesministerium fuer Bildung und Forschung)
- [8] Kitchenham B., Pasquini A., Anders U., Bøegh J., de Panfilis S., Linkman S., Automating Software Quality Modelling, Measurement and Assessment, in: *Reliability, Quality and Safety of Software-Intensive Systems, IFIP TC5 WG5.4 3rd International Conference on Reliability, Quality & Safety of Software-Intensive Systems (ENCRESS'97)*, pp. 43-53, Gritzalis D. (Editor), Chapman & Hall, London, 1997
- [9] Tsukumo A. N., Oliveira A., Rêgo C. M., Salviano C. S., Azevedo G. F., Costa M. C., Colombo R. M. T., Jino M., A Framework for Incremental Evaluation of Software Product Quality Based on ISO/IEC 9126, in: *Proceedings of the 6th Software Quality Management*, pp. 46-63, Bath, 1997
- [10] Tsukumo A. N., Oliveira A., Rêgo C. M., Azevedo G. F., Maldonado J. C., Aguayo M. T., Jino M., Tutumi R., The Second Experiment of Application of ISO/IEC 9126 Standards on Quality Evaluation of Brazilian Software Products, in: *Proceedings of the 6th International Conference on Software Quality*, pp. 46-63, Ottawa, 1996
- [11] Tsukumo A. N., Capovilla C. R., Rêgo C. M., Jino M., Maldonado J. C., ISO/IEC 9126: An Experiment of Application on Brazilian Software Products, in: *Proceedings of the Second IEEE International Software Engineering Standards Symposium*, pp. 184-190, IEEE Computer Society Press, Montréal, 1995
- [12] Bøegh J., de Panfilis S., SQUID: A Method for Managing Software Quality During the Development, in: *Proceedings of the European Space Agency Product Assurance Symposium and Software Product Assurance Workshop*, pp. 217-220, ESA Publication Division, Noordwijk, 1996
- [13] Kitchenham B., Linkman S., Pasquini A., Nanni V., The SQUID approach to defining a quality model, in: *Software Quality Journal*, 1997
- [14] Knorr G., The Gütegemeinschaft Software - A major Concept in the Certification of Software Quality, in: *Approving Software Products*, pp. 135-138, Ehrenberger W. (Editor), Elsevier Science Publishers B. V., The Netherlands, September 1990
- [15] Geyres S., NF Logiciel: affordable certification for all software products, in: *Achieving Software Product Quality*, pp. 125-135, van Veenendaal E., McMullan J. (Editors), Uitgeverij Tutein Nolthenius, The Netherlands, September 1997
- [16] Battison R., Bengtsson J., Källgren R., Piper L., Ran M., Samuelsson H., Stiernborg J., Manual för utvärdering av programprodukter enligt ISO 14598-5 och ISO 9126, *SPI - Föreningen Svensk Programvaruindustri*, 1996
- [17] Andersen O., Kyser H., Reproducibility of checklists, in: *Software Metrics for Product Assessment*, pp. 160-166, Bache R., Bazzana G., McGraw-Hill, London, 1994

### **Further reading**

Bache R., Bazzana G., Software Metrics for Product Assessment, McGraw-Hill, London, 1994

Rae A., Robert P., Hausen H.-L., Software Evaluation for Certification, McGraw-Hill, London, 1995

### **List of standards**

ISO Guide 25: General requirements for the technical competence of testing laboratories, 1990

ISO/IEC 9126: Information technology - Software product evaluation - Quality characteristics and guidelines for their use, 1991

ISO 9241: Ergonomic requirements for office work with visual display terminals (VDT's) - Part 11: Guidance on usability, 1996

ISO/IEC 12119: Information technology - Software packages - Quality requirements and testing, 1994

ISO/IEC 12207 Information technology - Software life cycle processes, 1995

ISO/IEC 14598: Information technology - Software product evaluation

- Part 1: General overview
- Part 2: Planning and management
- Part 3: Process for developers
- Part 4: Process for acquirers
- Part 5: Process for evaluators
- Part 6: Documentation of evaluation modules

EN45001: General criteria for the operation of testing laboratories, 1991

prEN 54: Components of automatic fire alarm detection systems: Part 2: Control and indicating equipment, 1992

IEC 880: Software for computers in the safety systems for nuclear power stations, 1987

IEC 1508: Functional safety: Safety related systems, 1997

DIN V VDE 801: Principles for computers in safety-related systems, 1994

DIN 19250: Fundamental safety aspects to be considered for measurement and control equipment, 1991

DIN 66285: Informationsverarbeitung - Anwendungssoftware - Gütebedingungen und Prüfbestimmungen, 1990

### **Curriculum Vitae for Jørgen Bøegh**

Jørgen Bøegh has a degree in Mathematics and Computer Science from Aarhus University. He is currently with DELTA Danish Electronics, Light & Acoustics in the Software Engineering

Division working as a project manager.

He has been involved in research in communication security and personal safety of software based systems. He was involved in the ESPRIT I project REQUEST (RELIability and QUality in European Software Technology) from 1985 to 87. From 1986 to 89 he was responsible for DELTA's participation in the MAP projects 'Network Security' and 'Software Integrity'.

In 1988 and 89 he managed an industrial collaborate project on integration of computer aided engineering tools within the Danish electronics industry.

From 1989 to 93 he was involved in the ESPRIT II project SCOPE (Software CertificatiOn Programme in Europe) and from 1994 to 96 he has been responsible for DELTA's participation in the ESPRIT III project SQUID (Software QUALity In the Development process) as well as the ESSI project PET (Prevention of Errors through Test).

From 1995 he has been responsible for DELTA's participation in the ACTS project Prospect and from 1997 also for the project VALSE (Validating SQUID in Real Environments) and the ESSI project EPIC (Exchanging Process Improvement experiences across SMEs by Conferencing on the Internet).

He was the head of the Danish delegation to the international standardization group ISO/IEC JTC1 SC7 and was appointed as editor of ISO/IEC 14598 Parts 3 and 6.

He is the author of several scientific papers and a book on object oriented software development. His research interests include software quality specification and evaluation, software measurement and testing, and software best practices.

## **DELTA Danish Electronics, Light & Acoustics**

DELTA Danish Electronics, Light & Acoustics is an independent private organisation affiliated to the Danish Academy of Technical Sciences (ATV) and approved by the Danish Ministry of Trade and Industry. DELTA has approximately 200 employees of which 100 are engineers or scientists.

The main objectives of DELTA are to contribute to innovation in electronics, optics and acoustics, and to promote the application of electronics in industry and society in general.

DELTA is active in research, development and testing within the field of electronics, for Danish as well as international companies, organisations and authorities. In addition to carry out projects for companies, DELTA has the obligation to evaluate emerging technologies. This obligation is partly met by DELTA participating in pan-European projects, partly by carrying out development work for major international research organisations.

DELTA Software Engineering is one of six divisions in DELTA. It consists of a Best Software Practice Group and a Communications Technology Group.

The Best Software Practice Group has a long tradition for working in the area of software quality. Software product evaluation based on MicroScope is a main activity and major research projects have been carried out in this area as well as contributions to international standardization. Software process maturity evaluation and process improvement is another major activity. The group has adopted the Bootstrap method and has performed more than 20 Bootstrap assessments during the last year.

The Communications Technology Group has an extensive knowledge in the area of broadband communication (ATM) and Internet technologies. Advanced experiments are carried out related to tele-education and CSCW. The group aims at promoting the use of new communication technology through co-operation with companies in research and development projects, as well as dissemination of results from research activities.

# The Reliability of ISO/IEC PDTR 15504 Assessments

**Jean-Martin SIMON**

A.Q.T.  
19, place de la  
Ferrandière  
69003 Lyon  
France  
jms.aqt@wanadoo.fr

**Khaled EI EMAM**

Fraunhofer Institute for  
Experimental Software  
Engineering  
Sauerwiesen 6  
D-67661 Kaiserslautern  
Germany  
elemam@iese.fhg.de

**Sonia ROUSSEAU<sup>(1)</sup>**

**Eric JACQUET<sup>(2)</sup>**  
(1)SANOFI Recherche  
sonia.rousseau@tls1.elfsanofi.fr  
(2)SANOFI Pharma  
eric.jacquet@tls1.elfsanofi.fr  
9, rue du Président S. Allende  
94256 GENTILLY  
France

**Frederic BABEY**

AFNOR  
Unité Conseil  
Tour Eurpe  
92049 Paris La Défense  
Cedex – France  
101513.2013@compuserve.com

## Abstract

*During phase two of the SPICE trials, the Proposed Draft Technical Report version of ISO/IEC 15504 is being empirically evaluated. This document set is intended to become an international standard for Software Process Assessment. One thread of evaluations being conducted during these trials is the extent of reliability of assessments based on ISO/IEC PDTR 15504. In this paper we present the first evaluation of the reliability of assessments based on the PDTR version of the emerging international standard. In particular, we evaluate the interrater agreement of assessments. Our results indicate that interrater agreement is considerably high, both for individual ratings at the capability attribute level, and for the aggregated capability levels. In general, these results are consistent with those obtained using the previous version of the Software Process Assessment document set (known as SPICE version 1.0), where capability ratings were also found to have generally high interrater agreement.*

## 1. Introduction

The international SPICE (Software Process Improvement and Capability dEtermination) Project developed a set of documents describing a model for software process assessment. These documents, known as SPICE version 1.00, were handed over to the ISO/IEC JTC1/SC7 Working Group 10 to evolve them to an international standard. Under the auspices of ISO/IEC, the documents are known by their number 15504. The 15504 documents have to go through a series of ballots by national bodies before they become an international standard. Subsequent to each ballot, the documents may be changed to address the ballot comments. The most recent balloting stages for 15504 are as follows:

- A *Proposed Draft Technical Report* (PDTR) ballot
- A *Draft Technical Report* (DTR) ballot

Following a successful DTR ballot, the 15504 documents will become a Technical Report Type 2. This is a designation given to a standard under trial. A TR-2 is expected to be revised within two to three years after its publication, with the intention of making it a full International Standard. A more detailed review of the standardization process for 15504 may be found in [8].

Since the beginning of the effort to develop an international standard for software process assessment, the importance of empirical evaluation of the evolving document set was recognized. This recognition is manifested through the SPICE Trials, which are conducted by the SPICE Project [16]. The first phase of the trials empirically evaluated the SPICE version 1.00 documents, and was completed in calendar year 1995. The second phase of the trials is now underway, and is expected to terminate in the Summer of 1998. This second phase is empirically evaluating the ISO/IEC PDTR 15504 document set.

One of the issues studied in the SPICE trials is the reliability of assessments [3]. In general, reliability is concerned with the extent of random measurement error in the assessment scores. There are different types of reliability that can be evaluated. For example, one type is the internal consistency of instruments (see [3][4][14]). This



type of reliability accounts for ambiguity and inconsistency amongst indicators or subsets of indicators in an assessment instrument as sources of error. In addition, in the context of the first phase of the SPICE trials, a survey of assessor perceptions of the repeatability of assessments was recently conducted [6].

Interrater agreement is another type of reliability. It is concerned with the extent of agreement in the ratings given by independent assessors to the same software engineering practices. As with many other process assessment methods in existence today (e.g., TRILLIUM-based assessments and the CBA-IPI developed at the SEI), those based on 15504 rely on the judgement of experienced assessors in assigning ratings to software engineering practices. This means that there is an element of subjectivity in their ratings. Ideally, if different assessors satisfy the requirements of the SPICE framework and are presented with the same evidence, they will produce exactly the same ratings (i.e., there will be perfect agreement amongst independent assessors). In practice, however, the subjectivity in ratings will make it most unlikely that there is perfect agreement. The extent to which interrater agreement is imperfect is an empirical question.

High interrater agreement is desirable to give credibility to assessment results, for example, in the context of using assessment scores in contract award decisions. If agreement is low, then this would indicate that the scores are too dependent on the individuals who have conducted the assessments. In addition, higher interrater agreement is expected to be associated with lower cost assessments since a consensus-building stage of the assessment method amongst the assessors would consume less time.

During the first phase of the SPICE trials, a number of interrater agreement studies have been conducted [5][7][9][10]. The general conclusion from these studies was that considerable variation in interrater agreement was witnessed, and so models were developed to explain this variation (as in [7]). The most relevant previous study in the current context is that reported in [13], where elements of the capability dimension were the unit of analysis (as opposed to process instances being the unit of analysis). That study found that interrater agreement is generally high. In this paper we present the first *evaluation* of the interrater agreement of process capability ratings done according to the ISO/IEC PDTR 15504 document set. This evaluation was conducted within the second phase of the SPICE trials.

Briefly, our results indicate that the capability ratings at each of the first three levels of the ISO/IEC PDTR 15504 capability dimension are highly reliable, and that the computed capability levels assigned to these processes are also highly reliable. These results are encouraging for current and potential users since they indicate that assessments using the emerging international standard maintain high reliability levels after the evolution to the PDTR version.

The next section of the paper provides an overview of the ISO/IEC PDTR 15504 practices rating scheme in the documents used during this study. Section 3 presents the research method that was followed for data collection and for evaluating interrater agreement. In section 4 we present the interrater agreement analysis results. We conclude the paper in section 5 with a summary and directions for future work.

## **2. The Capability Rating Scheme in ISO/IEC PDTR 15504**

The ISO/IEC PDTR 15504 architecture is two dimensional. Each dimension represents a different perspective on software process management. The first is the process dimension, and the second is the capability dimension.

The process dimension is divided up into five process categories. Within each category is a set of processes. Each process is characterized by a process purpose. Satisfying the purpose statement of a process represents the first step in building process capability (capability Level 1). The process categories are summarized in Table 2, and their associated processes are summarized in Table 3.

The capability dimension consists of six capability levels. Within levels 1 to 5 there exists one or two attributes that can be used for evaluating achievement of that level. The levels and their associated attributes are summarized in Table 4. A four-point achievement scale can be used to rate the attributes during an assessment. These are designated as F, L, P, N, and are summarized in Table 5. It is also possible to convert the F, L, P, N ratings of attributes into a single number that characterizes the capability of a process. The scheme for doing so is summarized in Table 6.

Within the context of a ISO/IEC PDTR 15504 assessment, the scope of an assessment is an organizational unit (OU) [8]. This is defined as all or part of an organization with a coherent sphere of activity and a coherent set of business goals. The characteristics that determine the coherent scope of activity - the process context - include the application domain, the size, the criticality, the complexity, and the quality characteristics of its products or services.

Ratings during an assessment are of process instances [8]. A process instance is a singular instantiation of a process that is uniquely identifiable and about which information can be gathered in a repeatable manner.

Process Category	Description
<b>Customer-supplier</b>	The <i>Customer-Supplier</i> process category consists of processes that directly impact the customer, support development and transition of the software to the customer, and provide for its correct operation and use.
<b>Engineering</b>	The <i>Engineering</i> process category consists of processes that directly specify, implement, or maintain a system and software product and its user documentation. In circumstances where the system is composed totally of software, the Engineering process deals only with the construction and maintenance of such software.
<b>Management</b>	The Management process category consists of processes which contain practices of a generic nature which may be used by anyone who manages any sort of project or process within a software life cycle.
<b>Support</b>	The <i>Support</i> process category consists of processes which may be employed by any of the other processes (including other supporting processes) at various points in the software life cycle.
<b>Organization</b>	The <i>Organization</i> process category consists of processes which establish the business goals of the organization and develop process, product, and resource assets which, when used by the projects in the organization, will help the organization achieve its business goals. Although organizational operations in general have a much broader scope than that of software process, software processes are implemented in a business context, and to be effective, require an appropriate organizational environment.

**Table 2:** Description of the process categories.

Process Category		Process	
ID	Title	ID	Title
<b>CUS</b>	<b>Customer Supplier process category</b>		
	<b>CUS.1</b>	Acquire software	
	<b>CUS.2</b>	Manage customer needs	
	<b>CUS.3</b>	Supply software	
	<b>CUS.4</b>	Operate software	
	<b>CUS.5</b>	Provide customer service	
<b>ENG</b>	<b>Engineering process category</b>		
	<b>ENG.1</b>	Develop system requirements and design	
	<b>ENG.2</b>	Develop software requirements	
	<b>ENG.3</b>	Develop software design	
	<b>ENG.4</b>	Implement software design	
	<b>ENG.5</b>	Integrate and test software	
	<b>ENG.6</b>	Integrate and test system	
	<b>ENG.7</b>	Maintain system and software	
<b>SUP</b>	<b>Support process category</b>		
	<b>SUP.1</b>	Develop documentation	
	<b>SUP.2</b>	Perform configuration management	
	<b>SUP.3</b>	Perform quality assurance	
	<b>SUP.4</b>	Perform work product verification	
	<b>SUP.5</b>	Perform work product validation	
	<b>SUP.6</b>	Perform joint reviews	
	<b>SUP.7</b>	Perform audits	
	<b>SUP.8</b>	Perform problem resolution	
<b>MAN</b>	<b>Management process category</b>		
	<b>MAN.1</b>	Manage the project	
	<b>MAN.2</b>	Manage quality	
	<b>MAN.3</b>	Manage risks	
	<b>MAN.4</b>	Manage subcontractors	
<b>ORG</b>	<b>Organization process category</b>		
	<b>ORG.1</b>	Engineer the business	
	<b>ORG.2</b>	Define the process	
	<b>ORG.3</b>	Improve the process	
	<b>ORG.4</b>	Provide skilled human resources	
	<b>ORG.5</b>	Provide software engineering infrastructure	

Table 3: The processes and process categories.

ID	Title
<b>Level 0</b>	<p><b>Incomplete Process</b></p> <p>There is general failure to attain the purpose of the process. There are no easily identifiable work products or outputs of the process.</p>
<b>Level 1</b>	<p><b>Performed Process</b></p> <p>The purpose of the process is generally achieved. The achievement may not be rigorously planned and tracked. Individuals within the organization recognize that an action should be performed, and there is general agreement that this action is performed as and when required. There are identifiable work products for the process, and these testify to the achievement of the purpose.</p>
<b>1.1</b>	<b>Process performance attribute</b>
<b>Level 2</b>	<p><b>Managed Process</b></p> <p>The process delivers work products of acceptable quality within defined timescales. Performance according to specified procedures is planned and tracked. Work products conform to specified standards and requirements. The primary distinction from the Performed Level is that the performance of the process is planned and managed and progressing towards a defined process.</p>
<b>2.1</b>	<b>Performance management attribute</b>
<b>2.2</b>	<b>Work product management attribute</b>
<b>Level 3</b>	<p><b>Established Process</b></p> <p>The process is performed and managed using a defined process based upon good software engineering principles. Individual implementations of the process use approved, tailored versions of standard, documented processes. The resources necessary to establish the process definition are also in place. The primary distinction from the Managed Level is that the process of the Established Level is planned and managed using a standard process.</p>
<b>3.1</b>	<b>Process definition attribute</b>
<b>3.2</b>	<b>Process resource attribute</b>
<b>Level 4</b>	<p><b>Predictable Process</b></p> <p>The defined process is performed consistently in practice within defined control limits, to achieve its goals. Detailed measures of performance are collected and analyzed. This leads to a quantitative understanding of process capability and an improved ability to predict performance. Performance is objectively managed. The quality of work products is quantitatively known. The primary distinction from the Established Level is that the defined process is quantitatively understood and controlled.</p>
<b>4.1</b>	<b>Process measurement attribute</b>
<b>4.2</b>	<b>Process control attribute</b>
<b>Level 5</b>	<p><b>Optimizing Process</b></p> <p>Performance of the process is optimized to meet current and future business needs, and the process achieves repeatability in meeting its defined business goals. Quantitative process effectiveness and efficiency goals (targets) for performance are established, based on the business goals of the organization. Continuous process monitoring against these goals is enabled by obtaining quantitative feedback and improvement is achieved by analysis of the results. Optimizing a process involves piloting innovative ideas and technologies and changing non-effective processes to meet defined goals or</p>

	objectives. The primary distinction from the Predictable Level is that the defined process and the standard process undergo continuous refinement and improvement, based on a quantitative understanding of the impact of changes to these processes.
<b>5.1</b>	<b>Process change attribute</b>
<b>5.2</b>	<b>Continuous improvement attribute</b>

**Table 4:** Overview of the capability levels and attributes.

<b>Rating &amp; Designation</b>	<b>Description</b>
Not Achieved - N	There is no evidence of achievement of the defined attribute.
Partially Achieved - P	There is some achievement of the defined attribute.
Largely Achieved - L	There is significant achievement of the defined attribute.
Fully Achieved - F	There is full achievement of the defined attribute.

**Table 5:** The four-point attribute rating scale.

<b>Scale</b>	<b>Process Attributes</b>	<b>Rating</b>
Level 1	Process Performance	Largely or Fully
Level 2	Process Performance Performance Management Work Product Management	Fully Largely or Fully Largely or Fully
Level 3	Process Performance Performance Management Work Product Management Process Definition and Tailoring Process Resource	Fully Fully Fully Largely or Fully Largely or Fully

**Table 6:** Scheme for determining the capability level rating for the first three levels.

Instructions for Conducting Interrater Agreement Studies

- For each process, divide the assessment team into two groups with at least one person per group.
- The two groups should be selected so that they both meet the minimal assessor competence requirements with respect to training, background, and experience.
- The two groups should use the same evidence (e.g., attend the same interviews, inspect the same documents, etc.), assessment method, and tools.
- The first group examining any physical artifacts should leave them as close as possible (organized/marked/sorted) to the state that the assessees delivered them.
- If evidence is judged to be insufficient, gather more evidence and both groups should inspect the new evidence before making ratings.
- The two groups independently rate the same process instances.
- After the independent ratings, the two groups then meet to reach consensus and harmonize their ratings for the final ratings profile.
- There should be no discussion between the two groups about rating judgment prior to the independent ratings.

**Figure 11:** Guidelines for conducting interrater agreement studies.

### **3. Research Method**

#### **3.1 Data Collection**

For conducting interrater agreement studies, we divide the assessment team into two groups. In the current study, each of these groups had one assessor. Ideally both assessors should be equally competent in making attribute achievement ratings. In practice, both assessors need only meet minimal competence requirements since this is more congruent with the manner in which the 15504 documents would be applied. Each assessor would be provided with the same information (e.g., all would be present in the same interviews and provided with the same documentation to inspect), and then they would perform their ratings independently<sup>1</sup>. Subsequent to the independent ratings, the two assessors would meet to reach a consensus or final assessment team rating. In the context of SPICE, this overall approach is being considered as part of the trials [3]. General guidelines for conducting interrater agreement studies are given in Figure 11. The actual phases of the assessments where the data was collected are summarized below.

---

<sup>1</sup> Under this requirement, one assessor may obtain information that was elicited by the other assessor, which s/he would have not asked for. The alternative to this requirement is that the two assessors interview the same people at different times to make sure that they only obtain the information that they ask for. However, this requirement raises the risk that the interviewees “learn” the right answers to give based on the first interview, or that they volunteer information that was asked by the first assessor but not the second. Furthermore, from a practical perspective, interviewing the same people more than once to ask the same questions would substantially increase the cost of assessments, and thus the cost of conducting the study. It is for this reason that these studies are referred to as “interrater” agreement since, strictly speaking, they consider the reliability of ratings, rather than the reliability of whole assessments. The study of “interassessment” agreement would involve accounting for variations in the information that is collected by two different assessors during an assessment.

### 3.1.1 Preparation Phase

As required by the ISO/IEC PDTR 15504 Part 2, we are defining the *input* at the beginning the assessment. This consists of:

- a) the identity of the sponsor of the assessment and the sponsor's relationship to the organisational unit being assessed,
- b) the assessment purpose including alignment with business goals,
- c) the assessment scope including:
  - ⇒ the processes to be investigated within the organisational unit,
  - ⇒ the highest capability level to be investigated,
  - ⇒ the organisational unit that deploys these processes,
  - ⇒ the process context
- d) the assessment constraints which may include:
  - ⇒ availability of key resources,
  - ⇒ the maximum amount of time to be used for the assessment,
  - ⇒ specific processes or OU's to be excluded from the assessment,
  - ⇒ the minimum, maximum or specific sample size or coverage that is desired for the assessment,
  - ⇒ the ownership of the assessment outputs and any restrictions on their use,
  - ⇒ controls on information resulting from a confidentiality agreement.
- e) the identity of the model used within the assessment,
- f) the identity of the assessors, including the competent assessor responsible for the assessment,
- g) the identity of assesses and support staff with specific responsibilities for the assessment,
- h) any additional information to be collected during the assessment to support process improvement or process capability determination.

During the preparation, an important issue is to collect the context of the organisational unit since the result of the assessment is context dependant. Being "context dependant" can best be explained through an example.

In our example, we can consider two organisations, the first is developing a software package with 2000 users on a world wide basis; the second is a production department which provides a specific MIS application to 20 users who are in the same building. The way those two organisations should organise their Help Desk in order to provide the best "customer service" (CUS.5) is completely different. For example:

- a) The first one established a service level agreement with dedicated resources and formal procedures to handle any request and to manage interviews and questionnaires to appraise user satisfaction.
- b) The second one mandated his project leader to log any request and to meet on a regular basis the users to appreciate their level of satisfaction.

In the first case, the dispositions are fitted to the complexity and the magnitude of the requirements. However, those dispositions seem too over played to the second organisation. The assessors therefore have the responsibility to tune their judgement about the capability attributes for the relevant process according to the context where the instance is assessed within.

The context tackles the following parameters :

- a) the size of the organisation being assessed;
- b) the number of organisational units involved in the assessment;
- c) the demographics of the organisational unit,
- d) the application domain of the products or services of the organisational unit, the level of organisational participation in performing the assessment (collecting the information, demonstrating conformance);

- e) the maturity of the supplier-sponsor relationship (the level of trust between the organisation and sponsor);
- f) the needs of the sponsor;
- g) the size, criticality and complexity of the products or services,
- h) the characteristics of the project for which the processes are evaluated (Process instance).

### **3.1.2 Data Collection Phase**

To conduct the assessment, we used the interview technique based on the assessment model described in Part 5 of ISO/IEC PDTR 15504, plus documents examination.

If necessary, we provide some additional base practices to the model Part 5 for some processes where we deem the Part 5 is too vague. For example, for the CUS.3 Process, we add the following base practices to the CUS.3.7 Deliver and install software:

- a) CUS.3.7.0 Identify requirements for replication, packaging, storage, handling before delivery
- b) CUS.3.7.1 Identify Infrastructure Environment for delivery
- c) CUS.3.7.2 Identify training requirements for the client for delivery
- d) CUS.3.7.3 Identify duties from the customer or the client for delivery
- e) CUS.3.7.4 check delivery before installation
- f) CUS.3.7.5 Perform the installation of the software
- g) CUS.3.7.6 validate the installation

For all of the processes in scope of the assessment, for the capability dimension, we are deciding the set of capabilities to apply to all processes. For example, we claim coverage of levels 1 to 3 only when the ISO/IEC PDTR 15504 assessment scope is to have a first diagnostic of the processes in use within the organisation.

### **3.1.3 Ratings Phase**

Each assessor collected his own assessment record during the interview. At the end of the day, each assessor took some time to review his own record and to make the process attributes ratings. Therefore, a specific meeting is dedicated to consolidate the assessment record and to establish a consensus between the 2 assessors when some divergence arises for one or several attribute ratings. This aspect is very important since one of the assessors may have missed or misunderstood some information. In the case that 2 both assessors have missed some information, the sponsor (or the interviewee(s)) is contacted to obtain the missing information.

### **3.1.4 Debriefing**

At the end of the assessment week (the number of days may depend on the number of assessed processes), the 2 assessors present to the interviewees the main results of the assessment. The objectives of this presentation are:

- a) to remind them about the concepts of ISO/IEC PDTR 15504
- b) to ensure the understanding of the meaning of the attributes by the interviewees,
- c) to consolidate with the interviewees the results of the assessment.

During this meeting, the interviewees have the opportunity to “negotiate” the results by, for example, presenting further evidence.. At this time, the results are only presented using a graphical approach.

### **3.1.5 Reporting**

We perform the final assessment report where we provide synthesis results (weaknesses and strengths) process per process at the organisation unit level. This



global analysis is completed with the detailed analysis result for every assessed process for the considered project. This report is therefore sent to the sponsor for approval.

### 3.2 Description of Organization and Projects

In our study, we used data from two assessments that were conducted in France during the Phase 2 of the SPICE trials. In these assessments, the ISO/IEC PDTR 15504 documents were used. The company where the assessments was conducted is called SANOFI.

The SANOFI company belongs to the ELF Group; its activities focus on drug research and production. All pharmaceutical molecules must undergo six to twelve long years of development from the moment of their discovery to the time they are given product licence approval. SAOFI R&D has 2,500 employees, in nine units located in six countries (France, UK, Italy, Hungary, Spain and USA). From the research stage on the compound, to international commercialisation, Sanofi R&D (Sanofi Research and Development) controls each phase to test scientifically both the indications for and the effects of the compounds.

The IS (Information Systems) departments interact with all of these activities as a support service. Computerized systems are necessary for several domains : discovery; preclinical studies, clinical investigation and support. Development methods are either conventional (V model) or prototype based. Software packages are largely used. Architecture is still "mainframe" for some systems but mostly "Client-Server". IS departments manage the computerized systems life cycle from the initialisation of the system to the retirement ; they are used to work closely with Users and with the support of the Research Quality Assurance.

Two OU's within this company were assessed. A combination of organizational and project level processes were assessed in each OU. Three projects were assessed in the first OU and two projects in the second OU. The characteristics of these five projects are summarized in: The processes that were assessed and the number of instances in each are summarized in Table 8.

	X1	X2	X3	Y1	Y2
<b>Size of project in terms of effort</b>	3 man/year	2,5 man/year	1 man/year	2 man/year	1 man/year 35
<b>Programming language,</b>	C, Visual basic + off-the-shelf softwares	Third generation language	specific SQL	C, Visual basic + on-the-shelf softwares	specific SQL
<b>Development or maintenance projects</b>	maintenance	maintenance	validation	maintenance	maintenance
<b>Application domain</b>	Electronic document management	data processing : collection, processing, visualisation	data base , Client-server	Electronic document management	data base , Client-server

**Table 7:** Characteristics of assessed projects.

Process	Number of Instances
ORG.1	2
ORG.2	2
ORG.3	2
ORG.4	2
ORG.5	2
CUS.3	5
CUS.4	4
CUS.5	5
ENG.7	4
SUP.1	4
SUP.2	4
MAN.1	4
<b>Total</b>	40 process instances

**Table 8:** Number of instances of each process assessed.

### 3.3 Description of Assessors

The same two assessors conducted both assessments. Both assessors met the minimal requirements stipulated in the ISO/IEC PDTR 15504 documents. In terms of experience and background, this is summarized in Table 9.

	Assessor A	Assessor B
<b>years in the software industry</b>	14	3
<b>years in process assessment and improvement</b>	7 (including software quality improvement)	2
<b>assessment methods &amp; models they have experience with</b>	ISO9001, SPICE V1, and ISO/IEC PDTR 15504	ISO 9001, Bootstrap, and ISO/IEC PDTR 15504
<b>number of SPICE-based assessments done in the past</b>	6 (approximately 150 process instances)	3 (approximately 90 process instances)
<b>internal vs. external to the organization</b>	external	external

**Table 9:** Experience and background of assessors.

### 3.4 Evaluating Interrater Agreement

To evaluate interrater agreement, we can treat the ISO/IEC PDTR 15504 achievement ratings as being on a nominal scale. Cohen [1] defined coefficient Kappa ( $\kappa$ ) as an index of agreement that takes into account agreement that could have occurred by chance. The value of Kappa is the ratio of observed excess over chance agreement to the maximum possible excess over chance agreement. See [11] for the details of calculating Kappa.

If there is complete agreement, then  $\kappa=1$ . If observed agreement is greater than chance, then  $\kappa>0$ . If observed agreement is less than would be expected by chance, then  $\kappa<0$ . The minimum value of  $\kappa$  depends upon the marginal proportions. However, since we are interested in evaluating agreement, the lower limit of  $\kappa$  is not of interest.

The variance of a sample Kappa has been derived by Fleiss et al. [12]. This would allow testing the null hypothesis that  $\kappa=0$  against the alternative hypothesis  $\kappa\neq 0$ . If we use a one-tailed test, then we can test against the alternative hypothesis  $\kappa>0$ , which is more useful. This means we test whether a value of Kappa bigger than zero as large as the value obtained could have occurred by chance.

The standard version of the Kappa coefficient assumes that all disagreements are equally serious. A weighted version of Kappa that allows different levels of seriousness to be attached to different levels of disagreement has been defined [2]. The weighted version of Kappa was used in previous studies on the reliability of process assessments [5][13]. We also use the same weighting scheme as applied in previous studies in the SPICE trials [5][13]. This assigns greater seriousness to disagreements on non-adjacent categories on the four-point achievement scale, and hence essentially treats it as an ordered scale.

### 3.5 Interpreting Interrater Agreement

After calculating the value of Kappa, the next question is “how do we interpret it?” A commonly used set of guidelines in previous interrater agreement studies (e.g., see [5][13]) are these of Landis and Koch [15].

In addition, we can test the hypothesis of whether the obtained value of Kappa meets a minimal requirement (following the procedure in [11]). The logic for a *minimal* benchmark requirement is that it should act as a good discriminator between assessments conducted with a reasonable amount of rigor and precision, and those where there was much misunderstanding and confusion about how to rate practices. It was thus deemed reasonable to require that agreement be at least moderate (i.e.,  $\text{Kappa} > 0.4$ ). This minimal requirement on interrater agreement has been used in previous studies in the SPICE trials that evaluate the reliability of process capability ratings [13].

Kappa Statistic	Strength of Agreement
<0.00	Poor
0.00-0.20	Slight
0.21-0.40	Fair
0.41-0.60	Moderate
0.61-0.80	Substantial
0.81-1.00	Almost Perfect

**Figure 12:** The interpretation of the values of Kappa.

We evaluate whether interrater agreement using weighted Kappa is greater than moderate agreement for each of the five attributes in levels 1 to 3 of the capability dimension. When performing so many statistical tests, the probability of incorrectly rejecting one of these null hypotheses (Type I error) is approximately 0.4. This means that there is reasonably high probability that at least one significant result would be found. We therefore use a Bonferroni adjusted alpha level for all hypothesis tests (see [17]).

## 4. Results

The results of evaluating interrater agreement for the five capability attributes are shown in Table 10. As can be seen, ratings on all five attributes have at least moderate agreement at an experiment-wise alpha rate of 0.1. These results concur in general with evaluations of

interrater agreement of capability ratings for the previous version of the document set (known as SPICE Version 1.0) [13][14].

For the interrater agreement of capability level ratings for each of the processes, the results also indicate statistical significance at an alpha level of 0.1 (see Table 10). Therefore, agreement is higher than moderate agreement.

The combination of these results indicates that whether one uses the attribute ratings or the capability ratings, their reliability is higher than moderate agreement. If it is accepted that moderate agreement is a minimal for practical usage, then these results are encouraging for users of ISO/IEC PDTR 15504.

It should be noted that these results have limitations in terms of their generalizability. First, further research is necessary to determine whether similar results would be obtained for a different pair of assessors. While both assessors who took part in this study met the requirements for qualified assessors as stipulated in the ISO/IEC PDTR 15504 documents, further empirical investigation is necessary to ascertain whether any assessors that meet these requirements can attain such interrater agreement results. Second, the assessments from which our data were collected were conducted using a particular assessment method. This method is similar to the method used in previous interrater agreement studies [13][14]. However, it remains to be investigated whether the usage of different methods will produce similar results.

Attribute #	Description of Attribute	Weighted Kappa Value	Interpretation
1.1	<i>Process performance attribute</i> The extent to which the execution of the process uses a set of practices that are initiated and followed using identifiable input work products to produce identifiable output work products that are adequate to satisfy the purpose of the process.	0.78*	Substantial
2.1	<i>Performance management attribute</i> The extent to which the execution of the process is managed to produce work products within stated time and resource requirements.	0.64*	Substantial
2.2	<i>Work product management attribute</i> The extent to which the execution of the process is managed to produce work products that are documented and controlled and that meet their functional and non-functional requirements, in line with the work product quality goals of the process.	0.60*	Moderate
3.1	<i>Process definition attribute</i> The extent to which the execution of the process uses a process definition based upon a standard process, that enables the process to contribute to the defined business goals of the organization.	0.64*	Substantial
3.2	<i>Process resource attribute</i> The extent to which the execution of the process uses suitable skilled human resources and process infrastructure effectively to contribute to the defined business goals of the organization.	0.86*	Almost Perfect
Capability Level	Process capability calculated according to scheme in Table 6.	0.70*	Substantial

**Table 10:** Interrater agreement evaluation results (\* indicates statistical significance).

## 5. Conclusions

In this paper we have presented the method and results of a study to evaluate the interrater agreement of the ISO/IEC PDTR 15504 emerging international standard for software process assessment. The study was based on two assessments conducted in France during the second phase of the SPICE trials. The results of the study indicate that the interrater agreement of these assessments was high, raising confidence in the usage of this version of the 15504 document set for process assessments.

Further studies of interrater agreement are planned during the second phase of the SPICE trials. As well as evaluations, we plan to develop models to explain the variation in the reliability of assessments in order to provide guidelines for increasing reliability.

## 6. Acknowledgements

This study is supported by ELF *Innovation* to promote software process assessment as a new technique for Software Quality Management.

## 7. References

- [1] J. Cohen: "A Coefficient of Agreement for Nominal Scales". In *Educational and Psychological Measurement*, XX(1):37-46, 1960.
- [2] J. Cohen: "Weighted Kappa: Nominal Scale Agreement with Provision for Scaled Disagreement or Partial Credit". In *Psychological Bulletin*, 70(4):213-220, October 1968.
- [3] K. El Emam and D. R. Goldenson: "SPICE: An Empiricist's Perspective". In *Proceedings of the Second IEEE International Software Engineering Standards Symposium*, pages 84-97, Canada, August 1995.
- [4] K. El Emam and N. H. Madhavji: "The Reliability of Measuring Organizational Maturity". In *Software Process Improvement and Practice Journal*, 1(1):3-25, September 1995.
- [5] K. El Emam, D. R. Goldenson, L. Briand, and P. Marshall: "Interrater Agreement in SPICE Based Assessments: Some Preliminary Results". In *Proceedings of the Fourth International Conference on the Software Process*, pages 149-156, December 1996.
- [6] K. El Emam and D. R. Goldenson: "An Empirical Evaluation of the Prospective International SPICE Standard". In *Software Process Improvement and Practice Journal*, 2(2):123-148, 1996.
- [7] K. El Emam, R. Smith, and P. Fusaro: "Modelling the Reliability of SPICE Based Assessments". In *Proceedings of the International Symposium on Software Engineering Standards*, pages 69-82, 1997.
- [8] K. El Emam, J-N Drouin, and W. Melo (eds.): *SPICE: The Theory and Practice of Software Process Improvement and Capability Determination*. IEEE CS Press, 1997.
- [9] K. El Emam, L. Briand, and R. Smith: "Assessor agreement in rating SPICE processes". In *Software Process Improvement and Practice Journal*, 2(4):291-306, John Wiley, 1997.
- [10] K. El Emam and P. Marshall: "Interrater agreement in assessment ratings". In K. El Emam, J-N Drouin, and W. Melo (eds.): *SPICE: The Theory and Practice of Software Process Improvement and Capability Determination*. IEEE CS Press, 1997.
- [11] J. Fleiss: *Statistical Methods for Rates and Proportions*, John Wiley & Sons, 1981.
- [12] J. Fleiss, J. Cohen, and B. Everitt: "Large Sample Standard Errors of Kappa and Weighted Kappa". In *Psychological Bulletin*, 72(5):323-327, 1969.

- [13] P. Fusaro, K. El Emam, and B. Smith: "Evaluating the interrater agreement of process capability ratings". To appear in *Proceedings of Metrics'97*, 1997.
- [14] P. Fusaro, K. El Emam, and B. Smith: "The Internal Consistencies of the 1987 SEI Maturity Questionnaire and the SPICE Capability Dimension". To appear in *Empirical Software Engineering: An International Journal*, Kluwer Academic Publishers, 1997.
- [15] J. Landis and G. Koch: "The Measurement of Observer Agreement for Categorical Data". In *Biometrics*, 33:159-174, March 1977.
- [16] F. Maclennan and G. Ostrolenk: "The SPICE Trials: Validating the Framework". In *Software Process Improvement and Practice Journal*, 1:47-55, 1995.
- [17] J. Rice: *Mathematical Statistics and Data Analysis*. Duxbury Press, 1987.

# Process Improvement for Better Software Quality

Katalin Balla  
IQSOFT Ltd., Budapest  
dr. Tamás Langer  
IQSOFT Ltd., Budapest

## Introduction

This article presents a process improvement experiment carried out at a Hungarian software company, IQSOFT Ltd. The experiment was called IQPM<sup>2</sup>: the first two letters have been taken from the company name, while PM<sup>2</sup> is the name of the structured methodology used to guide the work.

After a short presentation of the company, the background in quality management is described and the decision for starting IQPM<sup>2</sup> is justified.

The next chapter contains a more detailed description of IQPM<sup>2</sup>, presenting the project's main goals, plan, expected outcome, and basic results.

In the last chapter we emphasise those ideas resulting from IQPM<sup>2</sup>, which help the company in its quality-oriented work.

## The environment of the experiment

IQSOFT Ltd. is one of the main representatives of the software industry in Hungary. The company was established in 1990 by people working at that time at the Computer Research and Innovation Center. The company is a medium-sized one, having 70 employees currently. IQSOFT Ltd. has three main software activity types: software development (mainly in databases, using 4GL development tools), systems integration, and software implementation. The projects are generally small to medium size and can differ widely in their characteristics.

### *What we did beforehand in quality management...*

Since 1993 efforts have been made at IQSOFT to develop and introduce an internal quality management system. The basic reason for IQSOFT's management decision was - by that time - the emerging request to be ISO certified, formulated by a foreign customer. However, it's worth noting that many employees of the company have been implicated in the former quality-oriented research (done before 1989 at the Computer Research and Innovation Center), so, besides joining the nowadays popular (and market-requested) trend of ISO certification, some IQSOFT employees were already familiarized with the concept and importance of software quality.

The approach to quality was the process-based one, as suggested by the ISO 9000 series.

In the trial of building up and introducing a software quality management system, IQSOFT focused on the necessity of using structured system development and project management methodologies, together with their computer aided tools. Most of the employees are familiar

with structured methodologies (e.g. SSADM, Oracle\*Case Method, OMT), and some elements have been used before (further details in [1]). The reason for focusing on them was the belief that using a structured, well documented way of working, having a well defined process life-cycle-model (which covers the entire life-cycle, not only the parts being present in a structured system-development and a project management methodology ) will lead to introducing the concepts of quality management, at least in ISO terms.

### *...and what we learned from it*

Efforts made in building up and introducing a quality management system at IQSOFT have not produced the desired results. Both the diversity of projects (IQSOFT's projects are difficult to standardize) and human factors (software developers are more resistant to accepting standard prescriptions) are believed to be the reasons for this slow introduction.

Another reason for quality management system implementation difficulties has been identified as the attitude of the Hungarian software customers: they are only beginning to understand the value of a well defined, organized way of developing and introducing software products. As a consequence, they have been reluctant to invest in these improvement activities. Instead, they prefer to implement a working prototype in the shortest possible time, rather than waiting for a better documented, better-tested, and higher quality product.

In the trial of building up and introducing ISO conforming software quality management system, as the work progressed and as we understood the standards better, we started to have the feeling of lacking some important elements. For instance, the absence of differentiating between the elements of a software quality system caused the software product itself to be neglected. We came to the idea that a software quality management system will not be really operational and useful if we build it taking into account the ISO 9000-prescriptions only. (Some research done in parallel confirmed our ideas. According to [2], [3], for instance, the elements of a software quality framework are: products, processes and resources. Their correlation should be taken into account, and ISO 9000 prescriptions should be customized as such. The company's characteristics should also be taken into account when introducing a software quality management system. For a more detailed analysis of the successes and failures of software quality oriented work done at IQSOFT, see [4]).

Simply: it became obvious that a software quality management system, or even a structured system development or project management methodology cannot be introduced at once at IQSOFT. It also became obvious that organizational change and a change in working style is required at the company as a first step towards quality management.

## **The IQPM<sup>2</sup> project**

### *Justification...*

IQSOFT management has taken the lessons learned from these preliminary studies and attempted to focus its efforts on continuous feedback based improvement of the existing quality oriented elements of the business.

IQPM<sup>2</sup> project was started as result of winning the EU PHARE tender "Technology Development and Quality Management (TD&QM)" (in 1995) and it had as a basic goal the introduction of a companywide project management system at IQSOFT.

As - for the reasons described in the previous chapter - the introduction of any kind of a methodology has proved to be difficult at once, the management decided to use the guidelines provided by a structured methodology for introducing a project management methodology. Taking into account the former co-operation with Lucas Management Systems and the knowledge in Artemis Project View, PM<sup>2</sup> methodology of Artemis International and distributed towards Hungary by Metier Plancon (NL) has been chosen for supporting the project.



**...timing, project team...**

The IQPM<sup>2</sup> project started in February 1996 and finished on the 30<sup>th</sup> May 1997. The project team was composed of a project owner, a project leader, project members, external experts and one external consultant from Metier Plancon, Netherlands. Positions, roles and responsibilities in the project have been established, as well as the formal reporting mechanism.

**...main goals...**

The primary objectives of the IQPM<sup>2</sup> project have been:

- To implement a comprehensive project management system for software projects at IQSOFT Ltd., using Artemis International's structured methodology PM<sup>2</sup>.
- To analyze and document the IQPM<sup>2</sup> results as they relate to quality and cost improvements.
- To establish a project management system at IQSOFT (supported by a software tool; preferably Artemis Project View) which will provide a software industry model.
- Gaining experience in the product "Artemis Project View" (IQSOFT Ltd. is a distributor for Artemis Project View and provides training and assistance for its customers.), related methodologies, and associated software tools will be an outcome of the IQPM<sup>2</sup>.
- Since IQSOFT Ltd. can be classified as a typical software company, the IQPM<sup>2</sup> results will be transferable to other software companies.

**... and project evolution**

The activities of the IQPM<sup>2</sup> project were performed in the sequence suggested by the PM<sup>2</sup> methodology.

The basic phases (workpackages) were:

- Workpackage 1 (WP1): Awareness and requirements gathering
- Workpackage 2 (WP2): Project Management Standards and Procedures
- Workpackage 3 (WP3): Solution Definition and Implementation
- Workpackage 4 (WP4): Baseline Projects
- Workpackage 5 (WP5): Measuring and Evaluation.

Adding milestones suggested by key events, the high-level project plan is the one presented in Figure BK-LT-1.

**Error! Not a valid link.**

**Figure BK-LT-1.: Project plan of IQPM<sup>2</sup> project**

Basic scheduled dates have been reached, although some modifications have been made to the plan according to needs expressed during evolution.

**Some information about the project's main activities**

The scope of the phase *WPI* was to make the company aware of the needs for project management issues and to define those needs. After preparation (understanding the essence of the interview-technique suggested by PM<sup>2</sup> methodology for data-gathering, preparing the interview-questionnaires), data collection followed. IQSOFT management (managing director, technical director, financial director, marketing director) and 8 project leaders were interviewed.

After finishing the interviews, their results were analyzed. The result of the analysis was made public in a *Presentation* (29th of March 1996.)

*Definition sub-phase of WPI* started on the 1st of April 1996 and ended on the 19th of April. Although PM<sup>2</sup> methodology suggests running another more detailed set of interviews, PM<sup>2</sup> management decided to have less formal discussions instead, as practically all project management related problems and needs had been defined in the previous sub-phase. Consequently open questions and suggestions of the interviewees were presented in a structured way to those interested, and remarks and suggestions were taken into account.

The Project management needs definition was agreed by the IQPM<sup>2</sup> project team. The needs definition document was edited. This agreement can be considered to be one of the main milestones of the project: consensus was reached about the necessity and usefulness of a project management system, as well as about the basic contents of such a system. It also became obvious that a detailed description of roles and responsibilities at the company is needed.

Phase WP2 was started after these agreements. The main goal of this phase was to develop and document IQSOFT's own Project Management Standards and Procedures. The work was strictly based on results of WP1. The *IQSOFT Project Management Standards and Procedures Handbook* was worked out, made public and was subject to discussions and comments. The Handbook was modified according to the comments. The *IQSOFT Project Management Standards and Procedures* was presented the beginning of September 1996. To this presentation potential project leaders of IQPM<sup>2</sup> pilot-projects were invited. At the end of the presentation, the *IQSOFT Project Management Standards and Procedures Handbook* was accepted.

We mention that the *IQSOFT Project Management Standards and Procedures Handbook* contains definitions for IQSOFT projects, a classification of typical IQSOFT projects. For instance:

**Definition:** Project: a sequence of activities consisting of more than one atomic activity, which is carried out with a well defined scope by two or more persons, having the value of x Million HUF or more.

- Remarks:
  - Projects are carried out in certain *time* limits, but the time element does not make some activities - which do not fit the definition of the project - become a project.
  - Standards refer to projects, but elements of the standards can be used in any other activities.

**Definition:** Project types: at IQSOFT the following project types have been identified:

1. *Development project*: a project developing a new software system while using existing tools.
2. *Implementation project*: a project which introduces existing software - developed by IQSOFT or by any other company - in conformity with the user's requirements.
3. *System-integration project*: a project which integrates existing software (including operating systems, other basic software etc.) - developed by IQSOFT or by any other company - and hardware in conformity with the user's requirements.
4. *Research project*: a project which is started at the company's own initiative or as a result of a contract with an external organization. Its goal is to do research in different areas and to produce experience which will be used in other projects.
5. *Maintenance and support project*: a project carried out in the scope of providing support for a customer in the usage of a software system, developed, introduced or integrated by IQSOFT.
6. *Other project*: any project which does not fit in the first 5 categories (e.g.: project for producing a feasibility study).

- Remarks:
  - The different project types are characterized by their goal, the activities to be performed and their sequence, the members of the project and the required project organization, the methodologies and the tools to be used, the result of the project.
  - The above mentioned project types generally do not exist "on their own"; it can happen that in a project we find elements from more types. We define the type of project by taking into account the type that has the most elements present in the project.

The Handbook offers guidelines for the project life cycle model to be used, describing the activities, participants, input, output, deliverables for each phase, and gives precise indications for project initiation, planning, execution and follow up. The concept of project management model and project type model was introduced.(See detailed description in the

next chapter.) This makes it possible to use the standards for all project types being run at IQSOFT.

A set of needed documents was developed, which contain indications regarding the contents and outline of the documents.

- Project proposal
- Draft project plan
- Contract
- Project initiation document
- Detailed project plan
- Draft system plan
- Detailed system plan
- Test plan
- Test log
- Defect log
- User handbook
- Maintenance handbook
- Declaration of acceptance
- Hand-over confirmation
- User requirements - maintenance
- Maintenance log
- Project events log
- Invitation for review
- Review log
- Management report
- Change request
- Changing the project plan
- Project closedown document

Project plan models have been worked out. Possibilities to connect these plans to MS Exchange, Schedule+, MSAccess have been considered and described. The IQPM<sup>2</sup> project team recommends the use of these facilities for an integrated, in house system for project planning and follow up.

Phase *WP3- Solution Definition and Implementation* - has initially been planned for developing the IQSOFT -specific configuration and handbook for the Artemis Project View. IQPM<sup>2</sup> project management decided that project management standards and procedures should not be connected to any software: implementing Artemis Project View, or any other software was not made mandatory. Using MSProject was recommended.

Phase *WP4 - Pilot Projects* - started on the 11th of September 1996 and ended on the 30th of April 1997. All baseline projects used the standards and procedures delivered in phase WP2. A reporting mechanism was worked out regarding the use of standards and procedures.

The three baseline projects were:

1. Project1 - developing an integrated workflow system
2. Project2 - developing a data-recording system, making possible further development connected OCR
3. Project3 - developing an information system which fits the user's requirements, using databases

All pilot project managers have been trained in the usage of the Standards and Procedures. The evolution of the pilot projects followed the standards' prescriptions both in terms of life cycle and documents prepared. Besides the reports required by the standards, the project managers had the task of reporting on the usage of the standards and procedures. In these reports they mentioned elements of the standards and procedures which they used, elements considered useful / useless, elements that have to be changed etc. Important feedback became available regarding the standards and procedures.

We mention that the pilot project managers had the task of using the Standards and

procedures in a *flexible way, as it best helps their work*, so that the IQPM<sup>2</sup> team can use the feedback of the pilot projects in finalizing the Standards and procedures, by making it best fit IQSOFT's needs.

All pilot projects encountered some delay. In all cases the delay was the result of modification in the user's organization, due to which formal project closedown could not take place, although the deliverables were handed over in conformity with the plans. However, IQPM<sup>2</sup> project management took the decision to close phase WP4 on the 30th of April 1997, despite the fact that the pilot projects had not been formally closed. Justification of this decision is the fact that in terms of IQPM<sup>2</sup> all projects produced the expected results. A significant proportion of the Standards and procedures were used, as well as a large number of recommended documents. Feedback on Standards and Procedures was provided, recommendations regarding changes in the handbook were made.

According to the feedback, pilot projects needed the following effort in order to apply the Standards and Procedures:

Pilot project	Total effort	Effort spent on applying Standards and Procedures	Percentage of total effort needed for applying Standards and Procedures
Project1	4 man months	6-7 man days	~ 8,75 %
Project2	11 man months	23 man days	~ 10,45 %
Project3	8 man months	10 man days	~ 6,2 %

**Table 1.: Effort needed to use PM standards and procedures**

All pilot project managers considered the Standards and Procedures useful, especially in terms of introducing a more organized way of working. Standard document templates have been considered useful as well, with the remark that some modifications still have to be made. *Phase WP5: Measuring and Evaluation* started on the 1st of May 1997 and ended on the 30th of May 1997.

One important step in measuring and evaluation was the Bootstrap assessment carried out on the 5th and 6th of May 1997, regarding the overall organization and two pilot projects (Project2 and Project3).

According to the Bootstrap assessment, the overall organization has maturity level 2, while the pilot projects reached 2.50 in CMM. (See a short description of CMM in Annex.) To be noted that the overall maturity level increased from the previous Bootstrap assessment made in 1994. We consider an important result the fact that issues related to organization are situated on level 3, life cycle independent functions and process related functions reached level 2.75. Bootstrap assessment showed an increase specially on those fields which were present among the IQPM<sup>2</sup> goals: organization, description of standards and procedures, project management practices. The company reached level 4 in terms of detailed planning procedures. It's also worth mentioning that pilot project Project3 reached level 3.25 regarding the process related functions. The Bootstrap evaluation contains important remarks and suggestions regarding the activities to be completed in order to reach a higher level in CMM.

An other important result - already appearing in phase WP4 - was in fact a spinoff: the project manager of pilot project Project3 expressed his need to do measurement on the project he was running. His goal was to gather information on the working effectiveness in working of the project team members. His assumption was that, as a project progresses and people get used to the task, their interest and - in consequence - their productivity decreases. If this is true, it should be taken into account when planning new projects, because it results in a delay which will influence deadlines. Starting with these assumptions, we developed an activity-log (using, in fact, Goal- Question -Metric paradigm). This log was filled in daily by all members of pilot project Project3. The results will be analyzed when the project ends, and if the assumptions prove to be true, will result in a recommendation for making estimations when planning new projects.

*IQPM<sup>2</sup> project closedown* took place on the 30th of May, 1997, with the participation of all

project members. The results were made public on the 5-6th of June, at the regular IQSOFT meeting. As the pilot projects did not end by 30th of May 1997, we planned a follow-up activity, which will be begun on the 1st of June and will end when the last of the pilot projects ends. The scope of these activities is to gather further data and feedback about parts of Standards and Procedures which will be used in this phase of the pilot projects (e.g. handing over of the results, final acceptance procedures etc.)

During the execution of IQPM<sup>2</sup> project all required *reports* (3 progress reports, 1 final report, 1 report for publication) were finalized, submitted to the PHARE - committee and accepted by them.

## Basic results of the IQPM<sup>2</sup> project

In this chapter we emphasize those (expected or not expected) results of IQPM<sup>2</sup> project that helped IQSOFT Ltd. better understand its own activities, processes and structures, in reconsidering its quality-oriented work and in taking the company to a provably higher maturity level (in terms of CMM).

### *Activity types at a software company*

One basic idea which came through during IQPM<sup>2</sup> project is the distinction between two different activity types carried out in software companies: *activities needed for successful project management* and *activities depending on the type of the project needed for successful completion of the project's technical goals*.

Our experience has shown that while technical activities differ widely in the software organization, *project management activities are much more stable*. Hence the idea that project management can be regarded as a common framework for all of the company's projects. (This idea is confirmed by some articles, as well: [5], [6], [7], [8])

Projects can be modeled on both the mentioned activity types, that is a *project-management model* and a *project type model* can be constructed. Integrating the two models we have the *generic project model* (see Figure BK-LT.2.).

**Error! Not a valid link.**

### **Figure BK-LT-2.: Generic project model**

The main advantage of the "generic project model" - idea is the fact that it provides a *common framework* for all projects in an organization. This approach can be extremely useful in case of software companies situated on a low CMM maturity level: it can give a first impulse in the work of standardization.

Applying the model will give a certain feeling of success and motivates for further work in understanding our company in more depth. However, at this point no further visibility in any process of the company is provided. The generic project model is, in fact, a starting point for thinking in an other way about our processes.

At IQSOFT Ltd. the first trials to build up and introduce a software quality management system failed at the step of classifying the projects running at the company. All projects seemed to be different and there was no starting point for any unification.

### *Defining project management*

Project management is the totality of activities carried out for successful completion of the project. In our opinion these activities should be separated from the technical activities of the project.

The *project management model* is concerned with the aspects of project management. It shows the project life cycle from the project management point of view, the activities and their sequence to be performed in each life cycle phase, the input, output and deliverables of each phase / activity, the organization and the role needed for them, the way of reporting and recording progress, the quality objectives to be reached, and the way of checking them.

The project has the following life cycle (in terms of project management):

- Preparation

- Start
- Execution
- Conclusion.

The life cycle - in terms of project management - is shown in Figure BK-LT.2. As one can observe, *the type-model* (described in the next paragraph) *operates in the execution phase*. It is important to mention that at this point the type model is a black box in which any kind of technical activity or sequence of technical activities can be placed.

### ***Defining technical processes: project type models***

The project type model is concerned with the aspects of the technical work done in the project. It shows the technical life cycle, the activities and their sequence to be performed in each life cycle phase, the input, output and deliverables of each phase / activity, the organization and the roles needed for them, the way of reporting and recording progress, the quality objectives to be reached, and the way of checking them.

While the project management models can be considered stable in many organization types, the type models will be different for different organizations and within one organization there probably will also be more project type models. The type models can be of certain levels of detail, also depending on the concrete technology used in the project.

This step has to do with identifying the basic technical processes which are carried out. The identification can be done by looking at the process characteristics. It is helpful if the company managed to identify some basic project. At the same time, having an idea of the possible software process models can be of real help. The model identified can be represented using concepts of well-known models (waterfall model, V-model, spiral model etc.)

At IQSOFT Ltd. we identified for the development projects the type-model shown in Figure BK-LT-3.

### ***Defining products***

Describing the generic project model and the project type models means identification and description of *all project related processes at the company*. As the processes at software companies are there to build software products, the knowledge obtained hereto will provide a better understanding of the *products* which result from these processes. At this point we can start thinking about software *product quality characteristics, metrics* to measure them, and activities which can influence them.

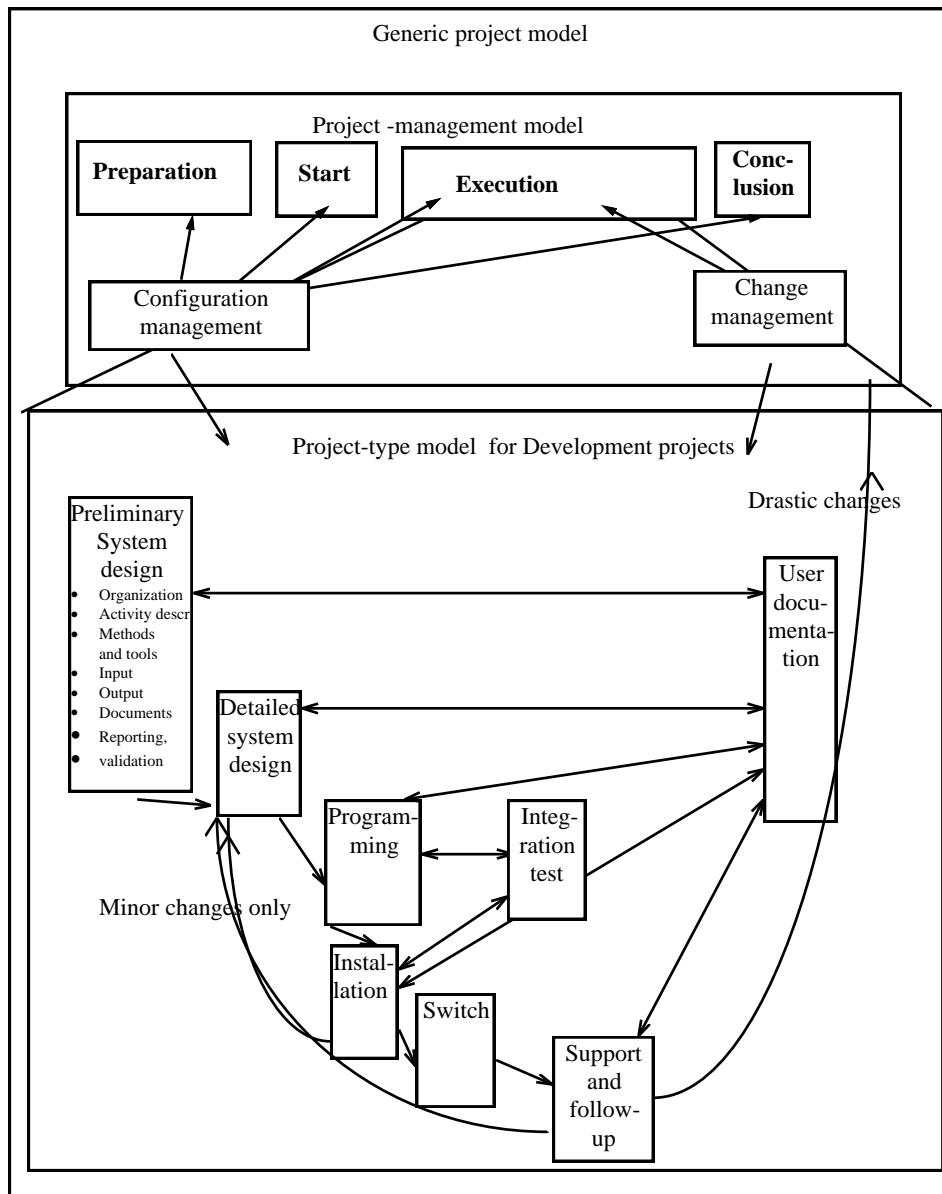


Figure BK-LT-3: Project type-model for development projects

***IQPM<sup>2</sup> results affecting the quality-oriented work at IQSOFT***

IQSOFT was committed to quality improvement many years before PIE project started. A set of quality handbooks were worked out beginning with 1993.

One basic difficulty in introducing overall quality management (and meeting ISO 9000 prescriptions) was the very fact that overall standards and procedures (valid, applicable and useful) for all IQSOFT projects could not be worked out. The reason for this failure was the variety of IQSOFT's projects.

Related to the mentioned, previous difficulties (and, thus, to the failure of ISO 9000 - compatible quality management system) IQPM<sup>2</sup> gave IQSOFT the following experience:

- At a software company the technical activities can be and should be separated from the management activities.
- At a software company project management activities are the most stable ones (they differ from project to project much less than technical activities).
- Therefore, standardization of any kind can be started by looking at project management activities, by introducing an overall project management system.
- Existing project management methodologies may not fit a company's needs.
- Therefore, working out a company's own project management standards and procedures is

- more likely to be successful.
- PM<sup>2</sup> methodology successfully supports the working out and introducing of a project management system which fits a company's own needs.
  - When describing a project management system, a project model can be made for the company. The model contains the elements of a project life cycle in management terms.
  - This model has a link to the technical project models ("project type models").
  - Integrating the two models, the company's generic project model can be built. This way, the difficult problem of having standards applicable for all projects at a company was solved.
  - The whole procedure is carried out in a structured and democratic way, involving all interested parties. This way, the chance of working out and maintaining a useful and widely accepted system is high.

The IQPM<sup>2</sup> project helped IQSOFT to understand the differences between technical and managerial activities, in building the generic project model for the company.

This way, the previously developed quality management system handbooks were completed. As - according to international studies - obtaining an ISO 9000 certificate is possible for companies situated on levels 2-3 in CMM, the approach followed by IQSOFT can help in ISO-registration as well. Our belief is that the procedure we followed within IQPM<sup>2</sup> ensures a more easily accepted and democratic way of building up and introducing elements of a quality management system required by ISO 9000.

#### ***Possibilities of continuing the work done within IQPM<sup>2</sup> project***

IQSOFT management has decided to continue working in a standardized way in terms of project management. Also, the decision has been taken to start a project in order to obtain ISO 9000 certification.

Possibilities of extending the results obtained by IQPM<sup>2</sup> are being considered. The experience obtained by IQPM<sup>2</sup> project team in using PM<sup>2</sup> methodology could be used in other Hungarian software companies.



## Annex: The CMM model

The CMM (Capability Maturity Model) was developed by the Software Engineering Institute of the Carnegie Mellon University, USA. It is a five level model which characterizes the maturity of a software company.

The CMM is subject to changes and updates. The companies situated on different maturity levels are supposed to have the following characteristics and specific elements (according to [6], [9]):

Level	Characteristics	Elements
1. Initial	The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics.	?
2. Repeatable	Basic <i>project management procedures</i> are established to track cost, schedule and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.	Requirements Management Software Project Planning Sw Project Tracking and oversight (in earlier versions: Project Management) Sw Subcontract Management Sw quality assurance Sw configuration management
3. Defined	The software process for both management and engineering activities is documented, standardized and integrated into <i>a standard software process of the organization</i> . All projects use an approved, <i>tailored</i> version of the organization's standard software process for developing and maintaining software.	Organization Process Focus Organization Process Definition Training Program Integrated Sw Management <i>Software Product Engineering</i> Intergroup Co-ordination Peer Reviews
4. Managed	Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.	Quantitative Process Management Software Quality Management
5. Optimizing	Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.	Defect Prevention Technology Change Management Process Change Management

**Table 2.: Characteristics and elements of software organizations situated on different levels of CMM**

## **References**

- [1] Balla, K.: Attempts to introduce software quality management. In: Austrian-Hungarian Workshop on Software Engineering, pg. 6.1.-6.10, Klagenfurt, 8-9 April 1994.
- [2] Trienekens, J.J.M.: Quality management in software production, a customer oriented approach, in: *Integration in Production Management Systems*, H.J. Pels, J.C. Wortmann (eds.), Elsevier Science Publishers B.V., 1992, pg. 127-135.
- [3] Kusters, R., Heemstra, F., Trienekens, J. : Defining systems quality: involving end-users, in: *Proceedings of the Profiting from Software Measurement Conference*, European Function Point User Group, 25-26 March 1993, Bristol, England, pg. 1-12.
- [4] Balla, K.: *Software quality management : ISO 9000- but not only*. Research Report. Technical University Eindhoven, Graduate School of Industrial Engineering and Management Science, March 1995. (EUT/BDK/68, ISBN 90-386-0055-0, ISSN 0929-8479)
- [5] Schulenklopper, J., Swede, V. van, Vliet, J.C. van: Situational metrics. In: *Software Quality Assurance and Measurement - A Worldwide Perspective*. pp. 294-302. Edited by Norman Fenton, Robin Whitty, Yoshinori Iizuka. International Thomson Computer Press, 1995. ISBN 1-855032-174-4
- [6] Humphrey, Watts: *Managing the Software process*. Addison-Wesley Publishing Company, 1989.
- [7] Weinberg, Gerald M. : *Quality Software Measurement*. Vol.1.: *System Thinking*. Dorset House Publishing, 1992. ISBN 0-932633-22-6
- [8] Smith, David J., Wood, Kenneth B.: *Engineering Quality Software*. A review of current practices, standards and guidelines including new methods and development tools. Second Edition. Elsevier Publishing, 1989.
- [9] Paulk, Mark C.: The Evolution of the SEI's Capability Maturity Model for Software. In: *Software process - Improvement and Practice*, Pilot Issue, August 1995, pp.:3-15

**Acronyms used in the article:**

Acronym	Explanation
IQPM <sup>2</sup>	Name of the project carried out at IQSOFT Ltd., Hungary, using the structured methodology PM <sup>2</sup>
PM <sup>2</sup>	Structured methodology which supports the introduction of a project management methodology at a company (PM x PM = PM <sup>2</sup> ). The methodology was developed by Artemis International
PM	Project Management
WP1	Workpackage 1 of IQPM <sup>2</sup> project
WP2	Workpackage 2 of IQPM <sup>2</sup> project
WP3	Workpackage 3 of IQPM <sup>2</sup> project
WP4	Workpackage 4 of IQPM <sup>2</sup> project
WP5	Workpackage 5 of IQPM <sup>2</sup> project
CMM	Capability Maturity Model

Professional CV

Name: Balla  
 Surname: Katalin  
 Civil status: married

She took her informatician diploma in 1984, from Babes Bolyai University of Science, Cluj, Romania. She worked first as a programmer then as a software engineer at CMCT Romania between 1984-1990. She was a scientific columnist (review and radio), in Bucharest, Romania (1990-1993). Beginning with 1993 she is the software quality manager of IQSOFT Ltd. Budapest, Hungary. She is the lecturer of the software quality management and project management courses given by IQSOFT. Parallel she followed a 1 year post-graduate course in QM at the Technical University Budapest. Presently she does her Ph.D. studies in software quality management at the Technical University Budapest and Technical University of Eindhoven, the Netherlands. She participates in 2 international projects with the topic of quality management and project management.

Professional CV

Name: dr. Langer  
 Surname: Tamás  
 Civil status: married

He is a founding member of IQSOFT and has been Technical Director since its inception. He gained his first degree in 1972 at ELTE (Eötvös Lóránd University) Budapest in Mathematics. He studies for his Doctorate in Computer Science at the same University and was awarded it in 1976.

Dr. Langer joined INFELOR as a research fellow and later Head of Department, working on MProlog from 1972-83, he continued this work at SZAMKI and SZAMALK. In 1983 he joined SZKI as the development project manager for the MProlog system.

Dr. Langer is co-author of three monographs on System Programming, Programming Languages and Programming Methodology.

In 1988 he received the State Award for contribution to the development of MProlog.

IQSOFT Ltd. is one of the main representatives of software industry in Hungary. The company has been established in 1990 by people working by that time at the Computer

Research and Innovation Center. The company is a medium-sized one, having in present 70 employees. IQSOFT Ltd. has three main software activity types: software development (mainly in data-base environment, using 4GL development tools), software localization, and software implementation. The projects are generally small to medium size and can differ widely in their characteristics. Since 1993, efforts have been made at IQSOFT to develop and introduce an internal quality management system (QMS), which fits ISO 9000 requirements. Project management related topics are also present among IQSOFT Ltd.'s services offered to other companies, as well as a wide range of technical courses given in data-base systems, object oriented programming etc. The company has a strong research division, which, besides studying new software systems / facilities in order to integrate them in the company's daily activity, participates in more international research projects. IQSOFT is an Artemis dealer.

# Building Resource and Quality Management Models for Software Inspections

Lionel C. Briand

*Fraunhofer IESE, Kaiserslautern, Germany*

Oliver Laitenberger

*Fraunhofer IESE, Kaiserslautern, Germany*

Isabella Wieczorek

*Fraunhofer IESE, Kaiserslautern, Germany*

## Introduction

Inspection of software artefacts, e.g., code, is a well-accepted approach to improve software quality and to lower software development costs [1][2][18]. However, within a particular organisation or across organisations, inspections vary widely with respect to their defect detection effectiveness and efficiency (i.e., cost-effectiveness) [3][4][6]. In addition, what is meant by inspection effectiveness and efficiency is often not clearly defined in quantitative terms. As a consequence, because the use of collected inspection data is not specified, project managers only perceive the cost of inspections and not the benefits of achieving higher product quality [18]. In this paper, we describe and use quantitative predictive models of inspection effectiveness and efficiency within a given environment in order to control the quality of inspected products and manage inspection resources. First, we provide an operational definition of inspection effectiveness and efficiency and discuss their underlying assumptions. Then, we identify some of the factors that may have an influence on inspection effectiveness and efficiency. Based upon these factors, we develop models for the prediction or evaluation of inspections. Finally, we develop scenarios describing how to use these models for prediction of the required inspectors' preparation effort to achieve a certain level of inspection effectiveness and the evaluation of efficiency after inspection completion. We consider this to be important from a practical point of view as managers get concrete guidelines on how to use these models.

As the models take into account the characteristics of a given environment, e.g., the inspection process, the functional relationship of the factors may be different across organizations or even for different families of projects within an organization. However, we argue that accurate and useful models for inspection effectiveness and efficiency can be developed with relatively small data sets. In this study, we have used data coming from less than 40 inspections. Thus, we believe that

our approach for building and using the models is applicable and valid in many other environments.

This paper is organized as follows: Section “Context of the Study” presents an overview of the context in which inspections were performed. Section “Data collection plan and procedures” describes the data collection plan and procedures used to build effectiveness and efficiency models. Section “Analysis” presents the data analysis we performed and the models we constructed. Section “Conclusion” presents our conclusions and plans for future work.

## Context of the Study

We first describe the environment in which the study was performed. Then we explain the inspection process performed in the environment under study. Finally, we identify requirements for a better management of inspections.

### Development environment

This study was performed on a project developing software for motor management systems. Thus, the products under development fall into the broad category of embedded real-time systems. Software implements a major functionality of an electronic control unit. The size of the control unit software is about 80 KLOC of highly complex C code. The software development process consists of the following main phases: the development of functional specifications, the design and the development of the code documents, and the integration of the code documents into an executable program version. Functional specifications are developed by functional specification analysts using the customer’s requirements document. The software developers use the functional specifications to develop the design and code documents. This includes the unit testing and debugging. In an inspection, the design/code documents are verified against the functional specification.

### Inspection process

Step	Description	Moderator	Inspector	Author
1	Planning	X		
2	Individual Preparation		X	
3	Inspection Meeting	X	X	X
4	Rework			X
5	Follow-Up	X		X

**Table BLW. 1: Inspection steps and involved roles**

Inspections are performed according to a well-defined process (Table BLW.1) in which a team of software developers analyzes a functional specification and its corresponding design and code document. The goal of an inspection is to find defects early on during development in order to save effort throughout subsequent phases of the development process. An inspection involves a moderator, between one and three inspectors, and the authors of the inspected specification/design/code documents. During the planning phase, the author hands out the developed specification, design, and code documents for inspection. The moderator assigns inspector(s), sets up a date for the inspection meeting, and distributes the documents to be inspected to the inspectors. Thus, the moderator plans and leads the inspection. S/he is responsible for the conformance of the actual inspection process to the defined inspection process. In the preparation phase, inspectors individually check the design/code documents against the functional specification to detect defects. The inspectors look for defects in an ad-hoc manner [10][11]. If an inspector detects a defect, it will be classified and documented on a defect report form. The moderator, the author of the inspected document, and the inspectors participate in the inspection meeting. The major goal of the inspection meeting is to decide for each defect detected during individual preparation whether it is a real defect or a false positive. A false positive is an issue that is logged as a defect throughout individual preparation but turns out not to be a real defect, i.e., affecting the correctness of the document inspected. Additional defects may be detected during the meeting but this is not the meeting’s primary goal. All real defects are logged during the meeting

on a meeting log. These are also logged on the meeting log. During rework activities, the authors of the functional specification and design/code documents use the meeting log to make appropriate corrections. Throughout the follow-up phase the moderator verifies whether the authors of the functional specification and design/code have addressed all the defects reported on the meeting log in a satisfactory manner.

### **Requirements for a better management of inspections**

When performing inspections, inspection managers (e.g., project manager, quality assurance manager) and moderators have to answer three major questions to plan, control, and assess inspections:

- 1 How to assign resources to an inspection in an optimal way?

This question refers to the amount of effort and sometimes, the number of inspectors, that should be assigned to an inspection. In the current literature, the reported numbers are not uniform. Bisant and Lyle [15] have found performance advantages in an experiment with two persons: one inspector and the author. Weller presents some data from a field study using three to four inspectors [4]. Bourgeois presents data showing that the optimal size is between three and five people [21]. Even less data is available regarding how much preparation effort is necessary to find defects in a given software artefact.

- 2 When to stop inspecting a software artefact?

Once an inspection is completed, a decision must be made whether the software artefact under inspection is of sufficient “quality” or whether it should be reinspected. Unfortunately, the literature mentions only a few approaches to make such decisions on an objective, quantitative basis. One approach is proposed by Ebenau [16] who applied techniques from statistical process control to inspections. With the help of control charts, he identifies software artefacts that are potentially error prone or poorly inspected, respectively. These software artefacts may be candidates for reinspection. Another approach is proposed by Vander Wiel and Votta [17]. They use capture-recapture models to estimate the number of remaining defects in the inspected software artefact. The decision about reinspection is based on this estimate.

- 3 How to assess the efficiency of inspections?

When introducing new reading techniques, instruments (e.g., checklists), or training in pilot projects, the next step is to assess the impact of such a change. How to assess, at a reasonable cost, a change in the inspection process?

To achieve the objectives stated above, one must determine the factors that may influence the (cost-) effectiveness of inspections. Factors include inspection rate, preparation effort [3][16], size and complexity of the inspected software artefact [3][16], human factors (e.g., skill, experience and training) [23], or the reading technique [19] [20] used to detect defects.

This paper presents a set of analyses which address the issues mentioned above by exploring and modelling the relationships between factors and inspection (cost-) effectiveness.

## **Data collection plan and procedures**

### **Plan**

During initial data analysis, we realized that most of the defects detected were found in the individual preparation phase. Consequently, to address the issues mentioned in Subsection “Requirements for a better management of inspections”, the following measurement goals were specified:

<b>Analyse</b>	the inspection process	[object]
<b>for the purpose of</b>	prediction	[purpose]
<b>with respect to</b>	preparation effectiveness	[quality focus]
<b>from the viewpoint of</b>	the process improvement team	[viewpoint]
<b>in the context of</b>	a project within company X	[environment]
<b>Analyse</b>	the inspection process	[object]
<b>for the purpose of</b>	evaluation	[purpose]

<b>with respect to</b>	preparation efficiency	[quality focus]
<b>from the viewpoint of</b>	the process improvement team	[viewpoint]
<b>in the context of</b>	a project within company X	[environment]

We have used the GQM template of the Goal/Question/Metric paradigm [7][8] to specify our measurement goals.

The requirements for better managing inspections are addressed in the following way: requirements 1 and 2 are addressed by our first goal by using the specified prediction model to estimate preparation effort based on a selected and adequate level of effectiveness. Requirement 3 is addressed by our second goal since we build a prediction model which we use to evaluate the efficiency of an inspection process. For further details, refer to the Subsection on “Multivariate Analysis”.

### **Procedures**

Throughout the inspection process data is collected using two different forms:

#### *1. Defect report form*

The defect report form is filled out by each inspector during the individual preparation step. It includes:

- the location of defects
- the classification of defects
- the description of defects
- the date when preparation started
- the preparation effort

#### *2. Meeting log*

The meeting log is filled out by the moderator during the inspection meeting and the follow-up phase of an inspection. It includes:

- the location of defects
- a classification of defects
- a description of defects
- collected data about the inspection meeting
  - meeting date
  - meeting effort
  - number of inspectors

All collected data about inspections is entered into a data base. Throughout the analysis we found that information about the size of the inspected documents is rarely available in the data base. Thus, we have collected some additional data about the size of code documents (lines of code) and the size of functional specifications (number of “operations” specified as discussed in Section “Dependent and Independent Variables”).

Based on other published experience [1][3][16], two influential factors which have shown to be important are not considered here because no data could be collected: inspectors’ ability and training, the actual defect density of the inspected documents. Furthermore, no particular reading technique was used.

### **Analysis**

In this Section we investigate the efficiency and effectiveness of the individual preparation step of the inspection process. We first define measures for effectiveness, efficiency, and identify the influential factors we could capture. Then, we perform standard univariate and multivariate analysis to build effectiveness and efficiency models.

### **Dependent and Independent Variables**

First we define the terms *effectiveness* and *efficiency* in the context of our study (dependent variables). To measure effectiveness and efficiency of inspection defect detection, ideal measures would have been to compute a direct estimation of the gains due to inspections in terms of both quality and cost. However, this is extremely difficult since no baseline concerning typical testing



cost and its comparison to inspection cost are available in the environment under study. Also we could not use *software defect removal efficiency*, a measure proposed by Jones [5], as the total number of defects in a design/code document is not known. Thus, as a measure of *effectiveness*, in order to be at least able to compare different inspections, we looked at the rate at which defects were detected in the inspected products. Therefore we decided to measure effectiveness as the density of defects detected.

$$Effectiveness = \frac{\text{Number of detected Defects}}{\text{Size}} \quad (\text{eq. 1})$$

Such a measure makes a major assumption: All inspected documents have a similar actual defect density. The actual defect density is the total number of defects of a document per size unit. We could make this assumption because all inspections were conducted in the same organisation, within the same application domain, and during a short period of time (within 1 year, since the introduction of inspections in the environment under study). However, we are aware that this may introduce uncertainty in our models. The goodness of fit of these models will depend on how close to reality this assumption is.

Efficiency was then defined by normalising effectiveness by the amount of preparation effort spent. The rationale was that efficiency would capture the detection *cost-effectiveness*, the “amount of effectiveness” achieved per unit of effort spent on defect detection.

$$Efficiency = \frac{\text{Number of detected Defects}}{\text{Size} \times \text{PreparationEffort}} \quad (\text{eq. 2})$$

We then considered the following factors that may influence defect detection (independent variables) [3]:

- *Preparation effort*

This is the effort that inspectors spent during preparation to detect defects in the inspected documents. Preparation effort cannot, in our study, be distinguished from inspection rate (i.e. speed at which documents are inspected) since, for most inspections, we only have one inspector and defects are mainly detected during preparation.

- *Size*

In order to define meaningful measures of efficiency and effectiveness, we need a suitable measure of size for the inspected documents. Because in our case, inspected documents were composed of three different parts (specifications, design, and code), identifying a unique size measure was not considered a realistic objective. Therefore we had two alternatives:

1. Use principal component analysis [9] to identify *an optimal linear combination of size measures*, i.e., capturing the largest amount of variance in the sample space defined by all size measures through a linear function of these measures, or
2. Select the *one size measure* that yields the best analysis results, i.e., the predictive models explaining the largest amount of variance for effectiveness and efficiency.

The first possibility did not yield satisfactory results and we therefore adopted the second one, which is in any case a more pragmatic approach since it requires only the data collection of one size measure. As a result, a specification-based size measure was used: the number of operations specified (referred to as *FDEFop*). This measure includes the operations used in a functional specification, e.g., arithmetic operations, logical operations, and signal processing operations. We believe that such a size measure also captures the complexity or cognitive load of performing inspections since inspectors have to go through, and keep in memory, a larger number of dependencies between operations.

We have not taken into account the different levels of operation complexity or the hierarchy decomposition levels of the specification. This should be addressed in the future.

## **Analysis Technique**

To develop prediction models for effectiveness and efficiency, we used regression analysis combined with specific techniques to deal with interactions, outliers, and non-linear relationships [12][13]. Thus, we looked at the relationships between effectiveness/efficiency and the factors for which we could collect data: preparation effort, document size (determining the complexity of inspections).

These relationships appeared to be of exponential nature and we linearized them to simplify the model construction process, i.e., use least-square formulae to compute the coefficients. Logarithmic transformations were performed on each side of the regression equation to obtain a linear model [13][14]. In addition, we standardised the data (see [24]). The reasons for the standardisation are: (1) data is centered and therefore the models include no intercepts, which means that less parameters had to be estimated, (2) the effect of different factors (as measured by regression coefficients) on efficiency/effectiveness is comparable, since regression variables are dimensionless, (3) an investigation of interactions between different factors is made easier. The transformations for standardisation are described in detail in [24] and the literature [13].

Following standard procedures, the “coefficient of determination”  $R^2$  [13] was used to assess the models’ goodness of fit. It can be defined as the proportion of the sample variation in effectiveness/efficiency explained by the model. Since the inclusion of additional variables can never decrease the value of  $R^2$  and normally increases it, we also report the adjusted  $R^2$  [14]. This term is adjusted for the number of variables in the equation. It helps to determine whether including another variable increases the explanatory power of the regression.

### Univariate Analysis

The following two Subsections discuss effectiveness and efficiency, respectively. The regression analysis models we present are built based on standardised data.

- **Effectiveness**

Surprisingly, no significant linear or exponential relationship appears between effectiveness and preparation effort (Figure BLW.1). When looking at it in isolation, it seems that spending more preparation effort does not significantly improve effectiveness. In Section “Multivariate Analysis, we will see that a significant relationship exists between effectiveness and preparation effort but is hidden by the effect of size on effectiveness. On the other hand, there is a significant decreasing exponential relationship (Effectiveness =  $a \times \text{FDEFOP}^b$  where  $b < 0$ ) between effectiveness and the number of specification operations (Figure BLW.2). A similar exponential model of the form Effectiveness  $e^{-a+b\text{FDEFOP}}$  has been tried but appeared not to fit the data as well. This comment is also valid for all subsequent models in this analysis.

After linearizing through a logarithmic transformation of both axes, we obtain a  $R^2 = 0.44$ , which is statistically significant. Effectiveness falls sharply between 0 and 30 specification operations and then reaches a minimum level.

Several potential causes can be identified a priori and need to be investigated: fatigue effect, lack of motivation, higher complexity of large documents. But one of the most likely direct causes is that inspectors spend proportionally less preparation effort, and are therefore less effective on large documents. Such a phenomenon was also observed in [3].

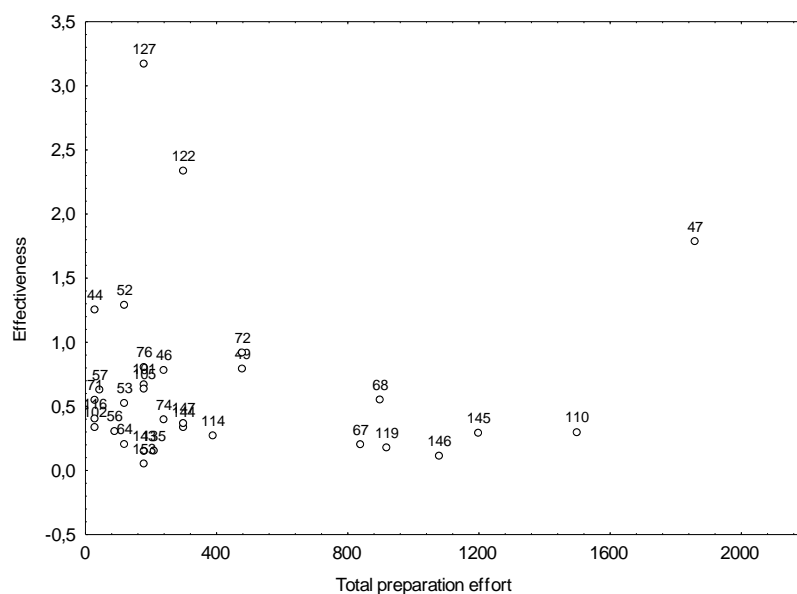
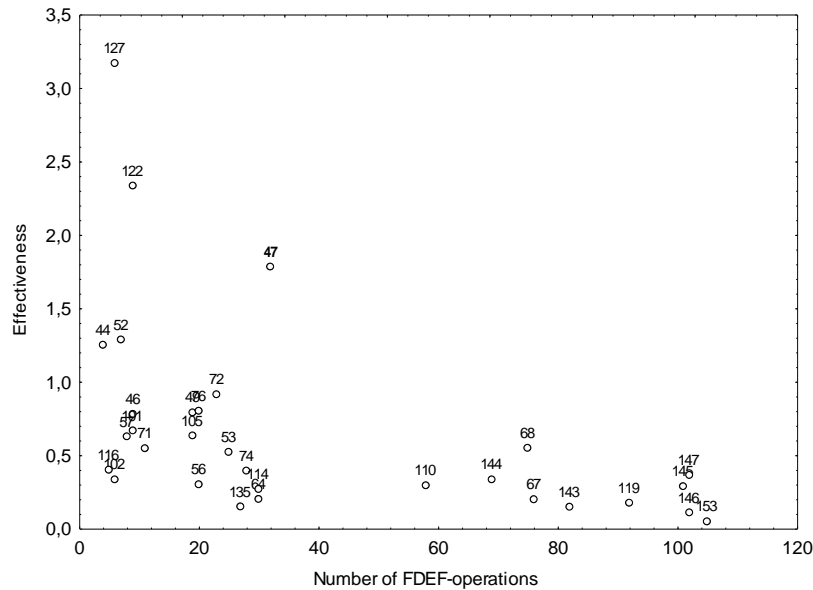


Fig. BLW.1: Effectiveness versus Preparation Effort



**Fig. BLW.2: Effectiveness versus Size**

Observation 47 is an outlier and represents a document that was inspected twice with 3 inspectors, thus showing a higher effectiveness than expected. One additional reason for high effectiveness may be that the 3 inspectors compensated for the large size of the inspected document. Unfortunately, we do not have more observations of that type to investigate this in more detail.

Inspections in which 2 inspectors inspect a particular document are around expected effectiveness values, even though they took a very large effort to reach that level of effectiveness. In any case, having 2 inspectors did not seem to help. However, we need more observations to confirm this result. One possible explanation is that, in many cases, the second inspector was a trainee.

We also investigated whether preparation effort and size are correlated. We found a significant but weak positive relationship ( $R^2 = 0.24$ ). This shows that the amount of preparation effort is not mainly driven by the size of the inspected documents.

• **Efficiency**

The reader will recall that efficiency is effectiveness divided by preparation effort and attempts to capture the relative cost of achieving a given level of effectiveness, i.e., the cost-effectiveness of the preparation process.

The scatterplot in Figure BLW.3 suggests there is a monotonic decreasing exponential relationship (of the form:  $\text{Efficiency} = a \times \text{PreparationEffort}^b$  where  $b < 0$ ) between efficiency and preparation effort. When linearizing the relationship, linear regression analysis yields a  $R^2 = 0.62$ . Following the same procedure, we obtain a  $R^2 = 0.83$  between efficiency and the number of specification operations (Figure BLW.4). When comparing the two regression beta coefficients, we see that the number of specification operations seems to have a stronger impact on efficiency than preparation effort (0.91 vs. 0.78 for number of specification operations and preparation effort respectively).

The results above can be interpreted as follows: **efficiency decreases exponentially with both document size and preparation effort**. This effect may be explained by the following hypotheses:

- defects detected later in the preparation process are usually more difficult to detect
- fatigue effects may also affect efficiency over time
- document complexity is higher due to a larger document size and information content
- there is a lack of motivation when inspected documents are too large.

**Multivariate Analysis**

Next we combine the effect of preparation effort and document size to predict effectiveness and efficiency.

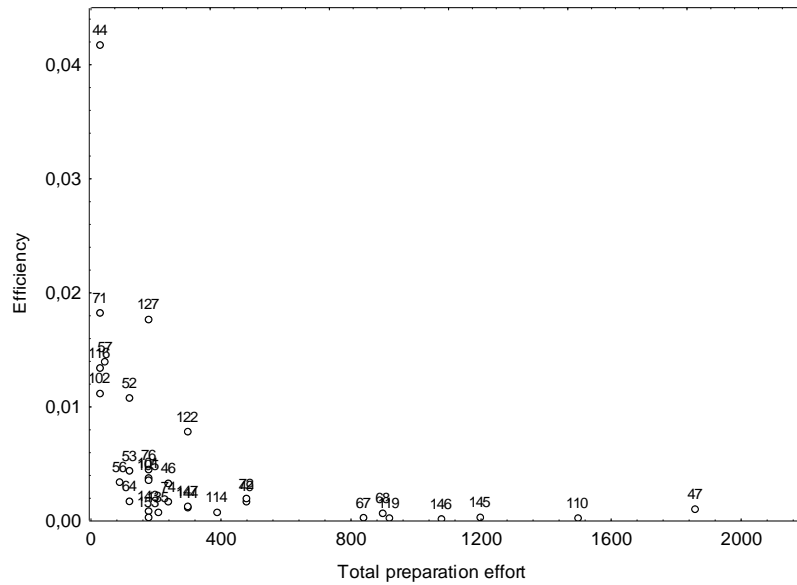


Fig. BLW.3: Efficiency versus Preparation Effort

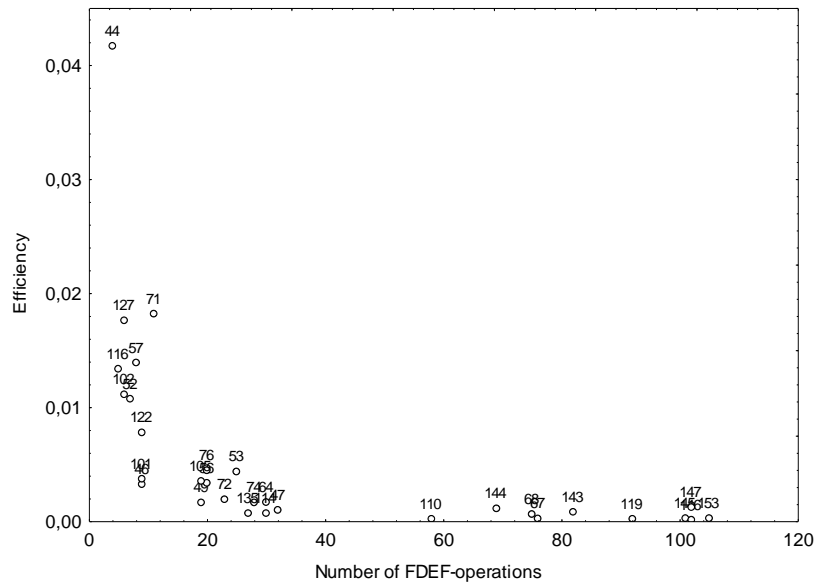


Fig. BLW.4: Efficiency versus Size

• Effectiveness

The model assumes an increasing exponential relationship with decreasing slope between effectiveness and preparation effort (i.e.,  $\text{Effectiveness} = a \times \text{PreparationEffort}^b$  where  $0 < b < 1$ ). This is justified by the fact that additional preparation effort should of course help detect more defects but that we also expect that various factors will decrease its impact over time on effectiveness. Example of such factors are: fatigue effects, more difficult defects are found later on in the preparation process. In addition, consistent with the univariate analysis results above, we assume a decreasing exponential relationship between effectiveness and the number of specification operations. Table BLW.2 describes the best model we identified based on the above assumptions, which specifies that there is a combined effect of preparation effort and the number of specification operations. In order to test our assumptions regarding the exponential shape of the relationships, we also attempted to build a linear model. But the results turned out, as expected, to be much poorer. In addition, using a different type of exponential model, i.e.,  $y=e^{a+bx}$ , did not fit the data as well. The first row describes the linearized model used to fit the standardised data (depicted with \*) where no interaction term was found to be significant.

Linearized Model based on standardised data (* symbol)	$\ln(\text{Effectiveness})^* = a \times \ln(\text{FDEFOP})^* + b \times \ln(\text{PreparationEffort})^*$ where $a < 0$ and $0 < b < 1$
Corresponding exponential model	$\text{Effectiveness} = \text{FDEFOP}^a \times \text{PreparationEffort}^b \times c'$

**Table BLW.2: Linearized and exponential model for Effectiveness**

The second row shows the actual exponential model where the constants  $a'$ ,  $b'$ , and  $c'$  can be computed from  $a$  and  $b$  by using the following transformations (where  $\sigma$  is the standard deviation of the sample):

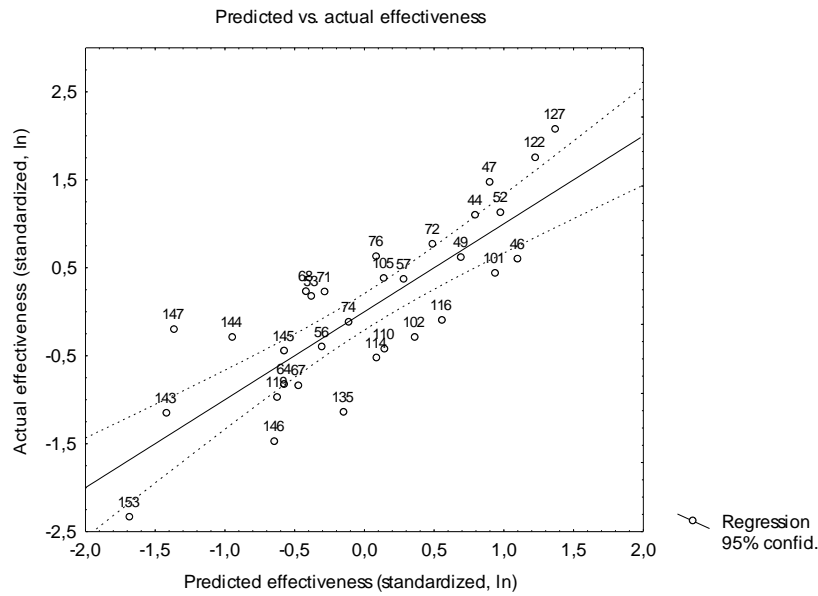
$$a' = a \times \frac{\sigma(\ln(\text{Effectiveness}))}{\sigma(\ln(\text{FDEFOP}))} \tag{eq. 3}$$

$$b' = b \times \frac{\sigma(\ln(\text{Effectiveness}))}{\sigma(\ln(\text{PreparationEffort}))} \tag{eq. 4}$$

$$c' = \frac{e^{\overline{\ln(\text{Effectiveness})}}}{e^{(\ln(\text{FDEFOP}) \times a' + \ln(\text{PreparationEffort}) \times b')}} \tag{eq. 5}$$

Further details on the derivation of  $a'$ ,  $b'$ , and  $c'$  are provided in [24]. The multiplicative, exponential model in Table BLW.2 suggests that when size (number of specification operations) increases, then the positive impact of preparation effort on effectiveness gets weaker. This can easily be explained by the fact that additional effort has more effect on the quality of small documents. When trying to combine the effect of preparation effort and number of specification operations using the linear model specified in Table BLW.2, we obtain Table BLW.3 through regression analysis. We can observe that coefficients have values consistent with our hypotheses, i.e.,  $a < 0$  and  $0 < b < 1$ . In this case, as opposed to univariate analysis results, preparation effort appears to be a very significant explanatory variable for effectiveness when used in combination with number of operations. This may be explained by the fact that the relationship between preparation effort and effectiveness was hidden by the stronger relationship linking number of operations and effectiveness (see univariate analysis).

As mentioned above, Table BLW.3 was computed by using standardised variables and the regression coefficients computed here are comparable in a given sample and referred to as *beta* coefficients. The comparability of beta coefficients stems from the fact that all variables are dimensionless and hence comparable. The beta coefficients indicate the average standard deviation change in effectiveness with a standard deviation change in preparation effort or size, when the other variable is held constant. From Table BLW.3, one can see that both preparation effort and the number of specification operations are very significant (i.e., show very low p-values). Consequently,  $R^2$  has improved (as well as the adjusted  $R^2$ ) when compared to the best univariate analysis  $R^2$  we obtained in the previous Subsection (i.e.,  $R^2 = 0.68$  versus  $R^2 = 0.44$ ). As opposed to the results obtained in the univariate analysis and as common sense would suggest, preparation effort shows a positive relationship with effectiveness, i.e., spending more preparation time helps improve the effectiveness of preparation. However, coefficient  $a'$  in Table BLW.3 is negative and this shows that the larger the number of specification operations, the lower the impact of preparation effort on effectiveness. Figure BLW.5 shows the scatterplot of the actual versus the predicted effectiveness. The axes in Figure BLW.6 represent actual and predicted effectiveness after logarithmic transformation and standardisation. This is depicted by (*standardized,ln*) in all the figures of the paper.



**Fig. BLW.5: Predicted vs. Actual Effectiveness with 95% confidence interval (Linearized model in Table BLW.2)**

			R <sup>2</sup>	0,68
			R <sup>2</sup> Adj	0,66
<b>Parameter</b>	<b>Estimate</b>	<b>Std Error</b>	<b>F-Ratio</b>	<b>p-value</b>
a	-1,15058	0,15	-7,60	0,0000
b	0,60760	0,13	4,55	0,0001

**Table BLW.3: Regression results for linear model in Table BLW.2**

Parameter	Estimate
a $\tilde{O}$	-1,01582
b $\tilde{O}$	0,5359
c $\tilde{O}$	0,56103

**Table BLW.4: Values for a', b', and c' for exponential Effectiveness model in Table BLW.2**

A project manager should use the model above to determine, based on the level of effectiveness s/he wants to achieve (e.g., depending on the criticality of the document inspected), what amount of preparation effort is necessary. Although the model's accuracy still needs to be improved by including the other factors (to be identified) of effectiveness, it is accurate enough to provide a valuable guideline. These results are in addition encouraging in the sense that building such a management model for inspections seems to be a feasible undertaking with a relatively small data set of about 30 inspections.

**• Efficiency**

Consistent with the results above, we assume decreasing exponential relationships between efficiency and the two variables: FDEFOP, PreparationEffort. Following a procedure similar to the one above, we transformed them before applying regression analysis. The specified model is described in Table BLW.5. The best linearised multivariate model is additive (no interaction terms) and is the one that simply combines preparation effort and document size. Again, no significant interaction term was identified. This may be explained by the fact that the product term of the two variables ( $\ln(\text{FDEFOP}) \times \ln(\text{PreparationEffort})$ ) is strongly correlated with  $\ln(\text{FDEFOP})$ . This collinearity problem explains that the interaction term is not significant and this issue has no straightforward solution. The data have already been centered and only additional data points might address the problem.

Linearised Model based on standardised data (* symbol)	$\ln(\text{Efficiency})^* = a \times \ln(\text{FDEFOP})^* + b \times \ln(\text{PreparationEffort})^*$ where $a < 0$ and $b < 0$
Corresponding exponential model	$\text{Efficiency} = \text{FDEFOP}^a \times \text{PreparationEffort}^b \times c'$

**Table BLW.5: Linearized and exponential model for Efficiency**

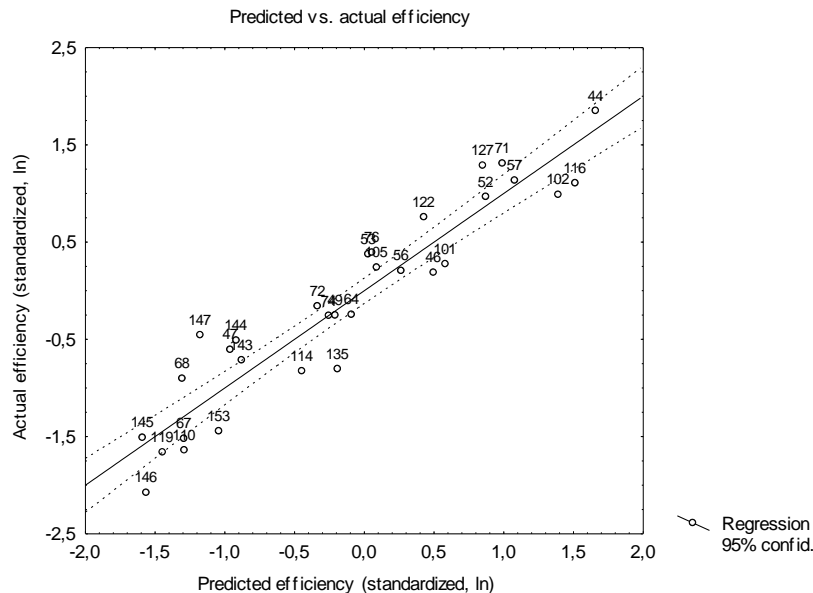
The constants  $a'$ ,  $b'$ , and  $c'$  can be derived from  $a$  and  $b$  by using transformations similar to the ones presented above (where  $\sigma$  is the standard deviation of the sample from which the model is built):

$$a' = a \times \frac{\sigma(\ln(\text{Efficiency}))}{\sigma(\ln(\text{FDEFOP}))} \tag{eq. 6}$$

$$b' = b \times \frac{\sigma(\ln(\text{Efficiency}))}{\sigma(\ln(\text{PreparationEffort}))} \tag{eq. 7}$$

$$c' = \frac{e^{\overline{\ln(\text{Efficiency})}}}{e^{(\ln(\text{FDEFOP}) \times a' + \ln(\text{PreparationEffort}) \times b')}} \tag{eq. 8}$$

The model in Table BLW.5 is very accurate (Table BLW.6:  $R^2 = 0.89$  versus  $R^2 = 0.83$  for the best result of univariate analysis) and can be used in various ways. For example, it can be used as a very accurate baseline of comparison when introducing changes in the preparation process aimed at improving the inspection efficiency. The effect of a change can be assessed as the difference in inspection efficiency between the expected value provided by the model and the actual efficiency value. For example, if a change results in a higher efficiency outside the 95% confidence interval of the model, the change may be deemed successful (with a 5% risk of error) and the new inspection practice may be generalised (progressively and with care) to the whole organisation. Figure BLW.6 shows the scatterplot of the actual versus the predicted efficiency.



**Fig. BLW.6: Predicted vs. Actual Efficiency with 95% confidence interval (Linearized model in Table BLW.5)**

			R <sup>2</sup>	0,89
			R <sup>2</sup> Adj	0,88
<b>Parameter</b>	<b>Estimate</b>	<b>Std Error</b>	<b>F-Ratio</b>	<b>p-value</b>
a	-0,7132	0,09	-7,60	0,0000
b	-0,3263	0,08	-3,94	0,0005

**Table BLW.6: Regression results of linear model in Table BLW.5**

Parameter	Estimate
$\tilde{a}$	-1,01585
$\tilde{b}$	-0,46428
$\tilde{c}$	0,55667

**Table BLW.7: Values for a', b', and c' for exponential Efficiency model in Table BLW.5**

The multiplicative, exponential model in Table BLW.5 suggests that the larger the size (and complexity) of the inspected document, the steeper the decrease in efficiency due to additional preparation effort. This may be explained by a lower motivation and higher fatigue effect when inspected documents are more complex.

### Applying Effectiveness and Efficiency Models

This Section describes typical usage scenarios for the models of effectiveness and efficiency presented above. The effectiveness model can be used for planning purposes, i.e., to predict the preparation effort before an inspection is conducted, or for control purposes, i.e., to achieve a higher level of document quality. The efficiency model can be used for evaluation purposes after an inspection is conducted. As this requires some effort for performing all the necessary calculations, this should be done automatically with a tool. We use the term “Tool based” to characterize the scenario steps that can be performed automatically by a tool in contrast to the term “Human based” for those requiring an interaction with the user, e.g., inspection moderator, project manager.

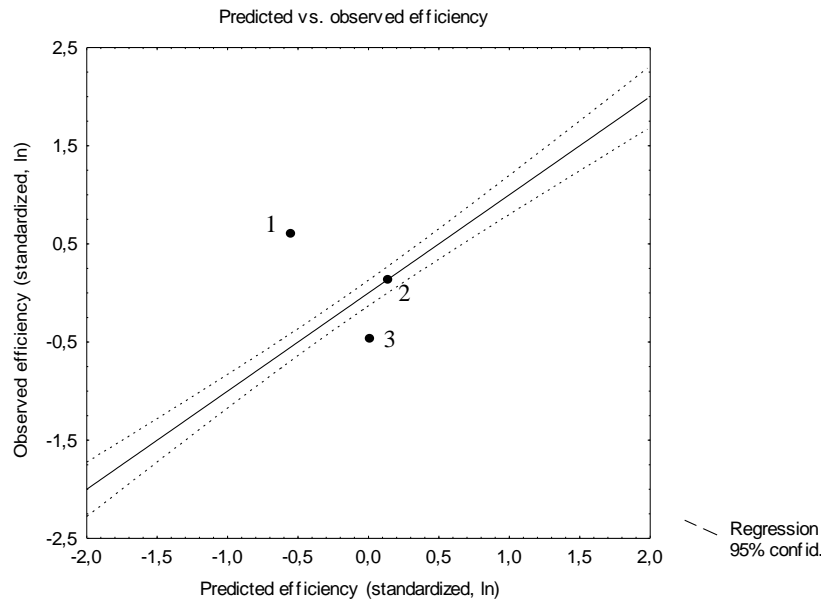
- **Scenario 1:** Procedure for using the effectiveness model for planning preparation effort depending on the criticality and size of the inspected documents.
  - 1 Tool based: Measure the size of the documents to be inspected, i.e., count number of operations in the functional specification.
  - 2 Human based: Select a n-points criticality scale, e.g., three-points scale. Thus, criticality can be defined on an ordinal scale, where 1= uncritical, 2=critical, and 3= very critical. Each level has to be carefully defined in the context of the project and/or the organisation.
  - 3 Human based: We follow the following rationale: The higher the criticality, the higher the level of effectiveness to be achieved, i.e., the higher the required documents quality. Therefore, we map each level on the criticality scale onto quantile values of the effectiveness distribution range. Quantiles must preserve the rank of criticality, e.g., map 1 to the 25% quantile value, map 2 to the 50% quantile value (median), and map 3 to the 75% quantile value. In this example, the rationale for choosing quantile ranges is that the 25% quantile value characterises inspections with low effectiveness. However, these inspections are still within a reasonable range and do not perform too poorly. The 75% quantile value characterises inspections with high effectiveness, but still is achievable, as opposed to outliers in the distribution. However, such a mapping has to be decided by quality managers and developers. Step 3 needs to be performed only once and not for each inspection.
  - 4 Human based: Select the criticality level of the documents to be inspected.
  - 5 Tool based: Determine the target effectiveness value, based on the level of the documents criticality (Table BLW.8). For example, criticality level 1 is mapped to the 25% quantile value of the effectiveness distribution: 0.266, criticality level 2 is mapped to the median: 0.40, criticality level 3 is mapped to the 75% quantile value: 0.789.



Level of criticality	Quantile range	Effectiveness Level
1	25%	0,266
2	50%	0,400
3	75%	0,789

**Table BLW.8: Effectiveness versus Criticality**

- 6 Tool based: Compute the preparation effort according to the model for effectiveness (Tables BLW.1 and BLW.2) where the outputs of steps 1 and 3 are used as inputs. The result is the predicted value for the inspection preparation effort for the considered document
  - **Scenario 2:** Procedure for using the efficiency model for the evaluation of a completed inspection. Evaluation may be motivated by the fact that an inspection was conducted following a new procedure or using a new technique, e.g., checklists.
- 1 Human and Tool based: Collect data about the size of the inspected documents (i.e., number of operations in the functional specification), the number of defects detected in the documents, and the total preparation effort spent during the inspection.
- 2 Tool based: Calculate the *actual* preparation efficiency value for the documents, by applying equation 2 (eq. 2).
- 3 Tool based: Apply the logarithmic transformation and standardise the actual efficiency value.
- 4 Tool based: Apply the logarithmic transformation and standardise size and preparation effort.
- 5 Tool based: Calculate the *predicted* preparation efficiency according to the linearized model for efficiency. Use the linearized, standardised size and preparation effort (outputs from step 4) as an input for the model.
- 6 Tool based: Locate the inspection based on actual and predicted efficiency values in the scatterplot, Figure BLW.7.
- 7 Human based: When evaluating the efficiency of inspections, there are three possible outcomes:
  1. The inspection is located outside the 95% confidence interval and the actual value is higher than the predicted value (see point 1 in Figure BLW.7). We can conclude, with a 5% probability of error, that the inspection was more efficient than predicted. Other confidence intervals [22], e.g., 99%, 90%, could have been selected. From a practical perspective, the inspection may be considered as an improvement over the way other “typical” inspections are conducted.
  2. The inspection is located within the 95% interval (see point 2 in Figure BLW.7). We can conclude, with a 5% probability of error, that the inspection was as efficient as expected.
  3. The inspection is located outside the 95% interval and the actual value is below the predicted value (see point 3 in Figure BLW.7). We can conclude, with a 5% probability of error, that the inspection was less efficient than expected. The completed inspection underperforms “typical” inspections in the environment. Whatever practice specific is to this inspection, it should not be applied any longer, although it may require further investigation before being definitely discarded since new technologies usually involve learning curves.



**Fig. BLW.7: Predicted vs. observed Efficiency: examples of evaluation cases**

The efficiency model can be used as a precise evaluation baseline. If the inspection process is changed, e.g., introducing checklists, it can be evaluated more precisely with the help of this model.

However, assuming that the change results in higher efficiency, and is generalised to the whole organisation, the efficiency model is no longer valid and should be developed again. Fortunately, we have shown that this is possible with a relatively low number of inspections (in this project 34).

## Conclusions

Although they require further investigation, the results of the statistical analysis show that useful models of efficiency and effectiveness can be built in our environment of study. These models enable inspection managers to plan preparation effort, for a given quality requirement, and allow for a precise evaluation of new inspection techniques (e.g., new reading techniques).

From a more general perspective, we believe that such an approach to build effectiveness and efficiency models should be applicable in other environments given a certain number of prerequisites: similar inspected documents (e.g., application domain, structure, and form), similar inspection process. If these conditions are not fulfilled, then these sources of variations have to be taken into account in the models as additional variables. Our results, though, show that such models can be built with a relatively small number of inspections. Therefore, it should be possible, in a given organisation, to build different models for various types of documents or inspection processes.

Carefully planning preparation effort and making sure that inspected documents are of adequate size appears to be very important to achieve an adequate level of inspection effectiveness. This is another example showing that improving inspections is also a matter of better organisation and management and not only the result of using better reading techniques like checklists.

Future work includes the development of more precise prediction and evaluation models and the investigation of the cost/benefit relationship of inspections in comparison with other verification or validation approaches, such as testing. In addition, we intend to collect more data about multiple inspectors' inspections in order to assess their impact in terms of effectiveness and efficiency.

## Acknowledgements

We thank Frank Bomarius, Khaled El-Emam, Thomas Fußbroich, Dirk Hamann, Dietmar Pfahl, Dieter Rombach, and Jürgen Wüst for their comments on the paper. In addition, we thank

our customer for providing us the inspection data.

## References

- [1] Ackerman, A. Frank, Lynne S. Buchwald, and Frank H. Lewski, Software Inspections, An Effective Verification Process, in: *IEEE Software*, Vol. 6, No. 3, May 1989, pp. 31-36.
- [2] Fagan, Michael E., Design and Code Inspections to Reduce Errors in Program Development, in: *IBM Systems J.*, Vol. 15, No. 3, pp. 182-211, 1976.
- [3] Christenson, Dennis A., Steel T. Huang, and Alfred J. Lamperez, Statistical Quality Control Applied to Code Inspections, in: *IEEE J. Selected Areas in Comm.*, Vol. 8, No. 2, pp. 196-200, Feb. 1990.
- [4] Weller, Edward F., in: Lessons from Three Years of Inspection Data, in: *IEEE Software*, Vol. 10, No. 5, Sept. 1993, pp. 38-45.
- [5] Capers Jones, Software Defect Removal Efficiency, in: *IEEE Computer*, vol. 29, No. 4, 1996.
- [6] Kelly, John C., Joseph S. Sherif, and Jonathan Hops, An Analysis of Defect Densities Found During Software Inspections, in: *J. Systems and Software*, Vol. 17, No. 2, Feb. 1992, pp. 111-117.
- [7] V. Basili, G. Caldiera, H.D. Rombach, Goal Question Metric Paradigm, in John J. Marciniak, editor, *Encyclopedia of Software Engineering*, volume 1, pages 528-532, John Wiley & Sons, 1994.
- [8] V. Basili, D. Weiss, A Methodology for Collecting Valid Software Engineering Data, in: *IEEE Transactions on Software Engineering*, 10 (6), 728-738, November 1984.
- [9] Susan C. Weller, A. Kimbal Romney, Metric Scaling: Correspondance Analysis, Series: *Quantitative Applications in the Social Sciences*, 75, SAGE University Paper, 1990
- [10] Tom Gilb, Dorothy Graham, *Software Inspections*, Addison-Wesley, ISBN 0-201-63181-4, 1993.
- [11] Susan H. Strauss, Robert G. Ebenau, *Software Inspection Process*, McGraw Hill, ISBN 0-07-062166-7, 1994.
- [12] Lawrence Lapin, *Statistics for Modern Business Decisions*, Second Ed., Hartcourt Brace Jovanovich, Inc., 1978.
- [13] L. D. Schroeder, D. L. Sjoquist, P. E. Stephan, *Understanding Regression Analysis - An Introductory Guide*, Series: *Quantitative Applications in the Social Sciences*, 57, SAGE University Paper, 1986.
- [14] W. D. Berry, S. Feldman, Multiple Regression in Practice, Series: *Quantitative Applications in the Social Sciences*, 50, SAGE University Paper, 1985.
- [15] Bisant, David B. and James R. Lyle, A Two-Person Inspection Method to Improve Programming Productivity, in: *IEEE Trans. Software Eng.*, Vol. 15, No. 10, Oct. 1989, pp. 1,294-1,304.
- [16] Ebenau, Robert G., *Predictive Quality Control with Software Inspections*, Crosstalk, June 1994, pp. 9-16.
- [17] Scott A. Vander Wiel, L. G. Votta, Assessing software design using capture-recapture methods, in: *IEEE Transactions on Software Engineering*, vol. 19, Nov. 1993.
- [18] Don O'Neill, Issues in Software Inspection, in: *IEEE Software*, January 1997.
- [19] A. Porter, L.G. Votta, V. Basili, Comparing detection methods for software requirements inspections: A replicated experiment, in: *IEEE Transactions on Software Engineering*, 21, June, 1995.
- [20] Victor Basili, Scott Green, Oliver Laitenberger, Filippo Lanubile, Forrest Shull, Sivert Sorumgard, Marvin V. Zelkowitz, The Empirical Investigation of Perspective-Based Reading, in: *Empirical Software Engineering*, 2, 1997.
- [21] K. V. Bourgeois, Process Insights from a Large-Scale Software Inspection Data Analysis, Crosstalk, October 1996.
- [22] S. Chatterjee, B. Price, *Regression Analysis by Example*, Second Edition, Wiley & Sons, 1991.

- [23] C. Sauer, R. Jeffery, L. Lau, P. Yetton, A Behaviourally Motivated Programme for Empirical Research into Software Development Technical Reviews, Technical Report, Center for Advanced Empirical Research, 1996.
- [24] L. Briand, O. Laitenberger, I. Wiczorek, Building Resource and Management Models for Software Inspections, Technical Report ISERN-97-06, International Software Engineering Research Network, March 1997.

## **Appendix**

### **Appendix 1 - The Authors**

**Lionel C. Briand** received the B.S. degree in geophysics and the M.S. degree in Computer science from the university of Paris VI, France, in 1985 and 1987, respectively. He received the Ph.D. degree, with high honors, in Computer Science from the university of Paris XI, France, in 1994.

Lionel started his career as a software engineer at CISI Ingénierie, France. Between 1989 and 1994, he was a research scientist at the NASA Software Engineering Laboratory, a research consortium: NASA Goddard Space Flight Center, University of Maryland, and Computer Science Corporation. He then held the position of lead researcher of the software engineering group at CRIM, the Computer Research Institute of Montreal, Canada.

He is currently the head of the Quality and Process Engineering Department at the Fraunhofer Institute for Experimental Software Engineering, an industry-oriented research center located in Rheinland-Pfalz, a beautiful wine region of Germany. His current research interests and industrial activities include measurement and modeling of software development products and processes, software quality assurance, domain specific architectures, reuse, and reengineering.

**Oliver Laitenberger** received the degree Diplom-Informatiker (M.S.) in Computer Science with a minor in Economics from the University of Kaiserslautern, Germany, in 1996.

He is currently a researcher in the department of “Innovative Software Engineering Approaches (ISE)” at the Fraunhofer Institute for Experimental Software Engineering (IESE) in Kaiserslautern, Germany.

His main research interest concerns reading techniques used in software inspections as well as their experimental validation.

**Isabella Wiczorek** received the degree Diplom-Informatikerin (M.S.) in Computer Science from the University of Koblenz, Germany, in 1994.

In 1995, she worked as a scientist in the Software-Technologie-Transfer-Initiative Kaiserslautern (STTI-KL). Since 1996, she is a researcher in the department "Quality and Process Engineering" at the Fraunhofer Institute for Experimental Software Engineering (IESE) in Kaiserslautern, Germany.

Working in industrial projects, she gained practical experience in the establishment of measurement-based quality improvement programs. Her research interests include software measurement, modeling, and evaluation.

### **Appendix 2 - Profile of the Fraunhofer IESE**

The Fraunhofer Institute for Experimental Software Engineering IESE focuses on research and development in the areas of innovative software development approaches as well as quality and process engineering. It supports the infusion of the results into industrial praxis by a large variety of services. In the area of innovative software development approaches, methods and tools are being developed, experimentally validated and adapted to the specific demands of individual companies. In addition, procedures are being developed, which allow companies to certify their software in terms of reliability and correctness. In the areas of quality and process engineering, software products and their development processes are measured, evaluated and improved in accordance with customer goals. Cooperation with industrial partners include support for introducing continuous improvement programs (e.g., software development) as well as selecting, adapting, evaluating and introducing innovative software development approaches. Supplementary services include training and education, preparation for software certification, curriculum for software process engineers, evaluation reports, patent assessments, software acquisition support and general consulting in software engineering. Partner companies work in areas where the quality of the products and services depend heavily on the quality of the supporting software - e.g., automobiles, household devices, telecommunications, traffic and environment control, banks and insurances. The Institute has been part of the Fraunhofer-Gesellschaft since January 1, 1996, and employs currently more than 40 scientific personnel - to be expanded to 70 scientific personnel within the first five years.

**Address:**

Fraunhofer Institute for Experimental Software Engineering  
Sauerwiesen 6  
D-67661 Kaiserslautern  
Phone +49 (0) 63 01 / 7 07 - 1 01  
Fax +49 (0) 63 01 / 7 07 - 2 00  
email: [info@iese.fhg.de](mailto:info@iese.fhg.de)  
Internet: <http://www.iese.fhg.de>

# Improvements through Configuration & Change Management and Test of Software for a Test Path Generator

Marcus Begemann

DTK GmbH, Palmaille 82, D-22767 Hamburg

Harry Debler

DTK GmbH, Palmaille 82, D-22767 Hamburg

## Abstract

*This paper focuses on the impact of configuration & change management and software test on software process improvement. It is shown how configuration & change management and software test are introduced in a software development process and which results emerged from this.*

*These insights are gained from a Process Improvement Experiment (PIE) in the context of the European Systems & Software Initiative's (ESSI) "Software Best Practice" program. This paper contains first intermediate results of the ESSI Project N°24,078, IMPACTS2. The experiment is concerned with improving the process architecture through configuration & change management and enhanced test strategies for a knowledge-based test path generator. During this experiment, methods and procedures for configuration & change management and test strategies as well as appropriate tools were introduced. The overall goal of the experiment was to increase the maturity level of the software development process with respect to configuration & change management and testing to a maturity level greater than 2, according to the BOOTSTRAP ranking, by the end of 1997.*

*This paper discusses the benefits of configuration & change management for software process improvement, gives a short overview about the underlying experiment, and discusses procedures to measure the results of the experiment.*

## Introduction

DTK is a company engaged in **software quality assurance**, **safety-evaluation** and **safety-realisation**, mainly for software driven analogue and digital railway components and systems, **IT-security** (systems and networks) as well as **logistics**. DTK is acting as a consultant, and as a software development and system house. Thus software development plays a significant, business-relevant role. In this context one of the strategic goals of the company is to permanently raise the quality of its products and services, by continuously improving the maturity of its processes. The introduction of modern techniques and tools is one step on this way

DTK is developing a knowledge-based test path generator for safety relevant analogous hardware components for the use in the railway environment. High transparency, high

traceability up to the developer in the case of incidents and very high product quality is required. It will be demonstrated how the process architecture for this test path generator can be improved by applying configuration & change management methods as well as enhanced test strategies.

## Project Overview

The baseline project which is described here is concerned with the development of one of the strategic products of our company, namely the test path generator for analogous relay-based circuits, for the use in the railway environment. This is a knowledge-based system, known as Relay-MASTER and part of the software product family of DTK (X-MASTER). It mainly consists of the following:

- a specially developed editor for generating models of the electrical circuits
- a control mechanism for model consistence tests
- a rule base, which contains know-how necessary for relay test path generation
- a set of generation routines, which produce test sequences
- different drivers, which write formatted output.

The software for this system is developed in C++ and Allegro Common LISP using the object-oriented paradigm. The system is not yet completed due to constantly changing requirements. A significant part of the development has been performed in the manner of rapid-prototyping. The software consists of roughly 80 modules, 20 per cent of which are multiply reused in different software configurations

## Starting Scenario

Based on the results of a BOOTSTRAP-compliant self-assessment the DTK decided to improve the configuration & change management (respectably we uses the acronym CM for this process) and the software test process. At the beginning we had following situation in our development process:

CM has not yet been declared as an overall company standard. It is not institutionalised and is not used systematically in all applications. There are no clearly defined responsibilities in the area of CM. This is the reason why single versions can often not be systematically identified, followed up and controlled. The administration of versions and releases are handled from project to project differently and strongly depend on the developer. Work being done up to now was supported only by small public domain tools, which have been used only for the administration of software code and not for the administration of documents and test procedures.

For the most software projects the development process of the DTK has been managed by trying to catch the CM process manually in the past years. Therefore, it is not astonishing that minimum of required configuration management is done. Development policies exist but the enforcement of rules depend on designers and programmers. Consequently, object-oriented analysis, design and implementation are emphasised. However increasing complexity of software projects implicates the problem of configuration & change control.

In a small development group the protection of sources files and documents has been dominated by a "*Drawing-board-Principal*". A developer had the possibility to "*checkout*" a source-file while it is not reserved and marked on the board as "*checked out*". Obviously, that such a solution depends directly on the smoothly co-ordination of the team members. In this

small software engineering world the storing of versions and revisions can still be done manually.

The most frustrating software problems are often caused by this "way" of CM solution, because they take a lot of time to fix, they often happen at the worst time, and they are totally unnecessary. Following problems could be identified while developing software at DTK in the last years:

- **Simultaneous updates**  
Two or more programmers work separately on one source file without arrangements. The last one to make changes can easily destroy the others' work.
- **Shared code**  
When bugs are fixed in code shared by several programmers, some of them are not notified.
- **Common code**  
When common program functions are modified in large systems, all the users need to be informed. Without effective code management, there is no way to announce these modifications to all users.
- **Versions**  
Most large programs are developed in evolutionary releases. With one release in customer use, another in test, and a third in development, bug fixes must be propagated between them. In large systems with several simultaneous active releases and many programmers working on bug fixes and enhancements, conflicts and confusion are inevitable.
- **Requirements**  
Documentation is based directly on design or implementation files. If programmers and designers work separately and without any arrangements on source and document files, some of the documents are not up to date.

### **Sample Scenario:**

For a sample scenario "*Requirements*" can be taken as one of the more obvious problems, which can occur without using configuration management. Therefore we describe this problem to understand the difficulties developing software without managing the configuration process. Software development can be divided into different parts. Generally these parts are analysis, design and implementation. All of these phases produces a lot of documents with different meanings. The analysis phase produces scenario and use case documents, the design phase produce sequence diagrams and design descriptions (e.g. UML). All these documents are required to implement the desired program. As a matter of fact each resulting source file will be linked with an implementation document. So a lot of requirements and dependencies between these documents are necessary. In the beginning of the development lifecycle all involved persons can easy manage the problem to "*coherence*" all documents. But our experiences show that the progress of the development lifecycle results in a bad coherence of documents. As a final result the complete program is even worse in documentation and inconsistent in some parts. Configuration management shall fix the problem of inconsistencies by defining the "*right*" way changing documents or sources. The first step to solve the problem is to define requirements for check-in sources or documents. *If a document is dependant on a source file and the source file changes – the check-in is only successful only if the document is also changing (see Fig. 1).*



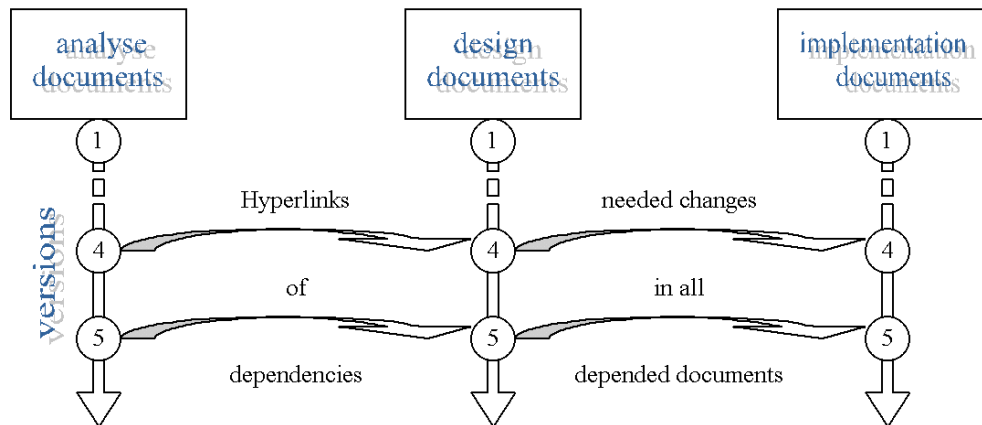


Fig. 1: Documentation changes in the configuration process.

It is easy to understand that these process improvements require some effort for planning. Furthermore it is necessary to establish complex arrangements for the involved persons. For further description of problems during the software developing process see [4].

The other section we wanted to improve was the test-process. The test-process of the DTK included three test activities. First, the software was tested by the developer itself. This activity consisted the debugging of the code and a limited function test. Following problems emerged thereby:

- **The developer tests his own software**  
 The test cases were prepared by the developer. Obviously the developer tests which he wanted to implement, not what was specified.  
 The tests were interrupted usually, if an error was found.  
 After the removal of errors only a few preceded test cases would repeat again. Thereby errors could appear through side effects.  
 A complete and uniform documentation of the test cases, the test run and the test results were missing.
- **Within the integration test** the developer made tests without having an overview about the tests the other developer had make.
- **There were made no static analyses**  
 No static analyses to estimate the quality of the software and to use it as a criterion for the dynamic tests (e.g. evaluate the complexity of the classes or the methods).  
 No defined control of the internal programming rules.
- **No defined test procedure**  
 No criterion for termination for test runs was defined. The developer worked with the following criterions: The first criterion was that he thought the module is all right. The second criterion was that he had not time anymore.

After the developers test, the project manager made a code reading together with the developer to review the module. But this review was only done when the project leader had the meaning that the module has a complex functionality. If errors were detected the developer removed them after the code reading. Then he made a unit test of his own with only a few test cases.

An acceptance test was executed by the user as a last step. This test was very extensive. The user created the test cases against the specification. The test was terminated if all test cases were executed or if too much errors were detected in one function. A complete and uniform documentation of the test cases, the test run or the test results were missing here also.

Therefore the developer had problems to find the errors the user had described. Also often errors were detected very late in the development process (through users) and the placing and removal of the errors produced a high expense. The reason was main emphasis of the test was on the side of the function test.

## **The Plans and the expected Outcome**

The development process should be significantly improved by focusing on the two described key areas: testing and configuration & change management. Thus the most important points of the experience were the following:

- **Introducing configuration & change management techniques and tools,**
- **Introducing systematic test methods and procedures, supported by suitable tools.**

The experience concentrated on that part of the software that was already multiply reused under different configurations (e.g. parts of the upper mentioned rule base). This part should benefit most and had the most significance for the successful evaluation of techniques and tools. It was be referred to as the baseline project.

The experience should be applied to the baseline project in two subsequent phases and should address two groups of programme modules: the baseline project kernel (BPK) and the baseline project subset (BPS). In this time the experiments concentrate on one major part of DTK's Relay-MASTER, which was the BPK. Currently one part of the upper mentioned rule base is considered for these purposes. During the next phase of the project the BPS will be the subject of the experiments. It will additionally comprise the remaining parts of the rule base and a subset of the generation routines, which produce test sequences. The size of the baseline project is currently estimated at 15 to 20 per cent of the whole software package.

### ***Configuration & Change Management***

Process improvement—in the realm of software as well as anywhere—means transforming some process that was always "somehow" working into a well-defined, well-structured, and well-documented process. This makes the underlying know-how transparent and thus accessible to everyone involved: the process finally will become replicable and incrementally improvable in itself.

CM is one aspect improving the software development process.

CM is providing a set of features that contribute considerably to an improved process. It converts the heterogeneous world of electronic products in a software project (sources, data, executables, tools, documents, and so on) into homogeneously represented objects whose interrelationships and interdependencies are well-documented. Process steps from requirement input to software changes to reintegration to reinstallation can be automated (to a certain extent). They become thus well-defined and replicable. In order to evaluate the potential success of this approach to process improvement the following features are considered: [3]

- Implementation of a configuration management plan
- Version Control
- Workspace and Release Management
- Build Management
- Process Management

## Testing

The experiments should take into account the specific techniques of knowledge-based system development, mainly concentrating on the following aspects:

- **Definition of Systematic Test Procedures**  
A manual for the test process should be created. These manual should be included for example description of the test process like organisation, co-ordination, planning, associate people, test procedures, test methods, evaluated test tools.  
The results should be measure to have a comparison and to have a possibility to evaluate the improvements.
- **Development of Automated Test Procedures**  
With the installation of a test tool the static tests should be automated completely and the dynamic tests should be nearly automated and repeatable.
- **Development of Reproducible Test Procedures**  
With the creation of test scripts (which include the test cases), the production of log-files (which include the test run) and the integration of configuration management all tests should be reproducible.
- **Definition of Standardised Test Documentation (see Fig. 2)**  
All considerations and hypothesis which are important for the definition and creation of the test case should be documented. The documentation of all the testing activities should be standardised on a company level.  
It should be defined protocol, report, documentation forms.

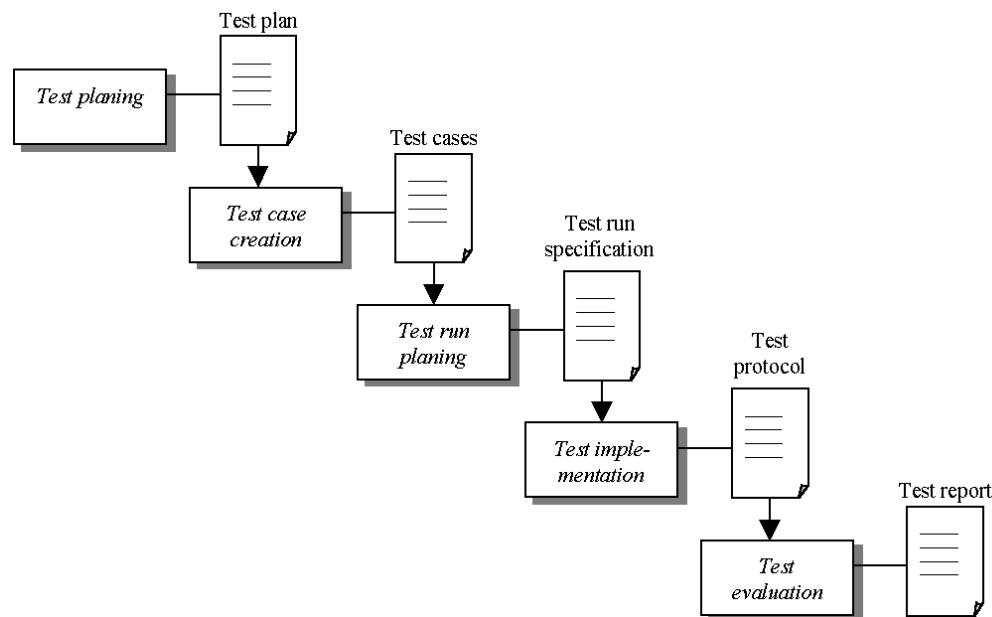


Fig. 2: Standardised Test Documentation for the test process [6]

## Measurement of Results

Intermediate results within the PIE are difficult to measure, as it aims at an increase in the overall quality of the software development process. This was the reason for us to perform metrics with our subcontractor. Therefore we use the Goal Question Metric Paradigm (GQM). Goal-oriented measurement according to the GQM approach provides the means for deriving metrics from precisely defined measurement goals, and for interpreting the collected data in

the context of the measurement goals.

A GQM goal states the object to be measure, the purpose of the measurement task, the quality aspect analyzed, the point of view from which the analysis is performed and the context of the measurement [5]. First of all a GQM plan was prepared, to receive from the GQM method metrics. A GQM plan consists of the goal, a set of questions divided into different subgoals and, for each question a set of metrics [5]. Derivation of questions and metrics from a GQM goal is a demanding process. The gathering of the necessary information is supported by interviewing project members. The role of the person who is interviewed for a particular GQM plan has to match the viewpoint description in the GQM goal. With the metrics which we have create with this method, we have measured our effectiveness of the test process and the configuration & change process.

The previous part described the possibilities of process improvement by implementing well-defined and structured methods. The following part describes the implementation of process improvements by the DTK.

### **The Implementation of the improvement Actions**

Software systems, like the Relay-Master, assemble a lot of elements. These elements (source files, databases, modules, documents, test plans) have a great number of interrelationships. One feature of configuration & change management is to handle this large amount of elements (Fig.3). Changes are necessary in the development and maintenance process of software systems. Together with a large number of elements irregular and unchecked changes could be a potential source for errors and inconsistencies.

Within the software life cycle process it could be taken into consideration to reuse parts of the software or extend the software life cycle. Obviously that different persons or parts of the organisations are responsible to maintain or develop the further software process. Problems can be avoided only if the information's and interdependencies of the whole system are stored effectively.

Software development is labour intensive and costly. Testing as one part of the development process consumes the majority of developers effort. A important problem in software testing is to determine the steps which are necessary to implement test plans and strategies.

In the following parts we describe the consideration that has been done by the DTK for the ESSI Project N°24,078, IMPACTS2. First we start to introduce to managing configuration & change procedures for the software development process. Furthermore we describe the testing process that is established during the evaluation of measurements and process improvement criteria.

For the implementation of configuration management it is essential to consider certain preparations and measures such as company software development policies and configuration management plans. Otherwise the installation of a configuration management system could fail. In the next part we describe the considerations and plans of establishing configuration management into the software development process of the DTK.

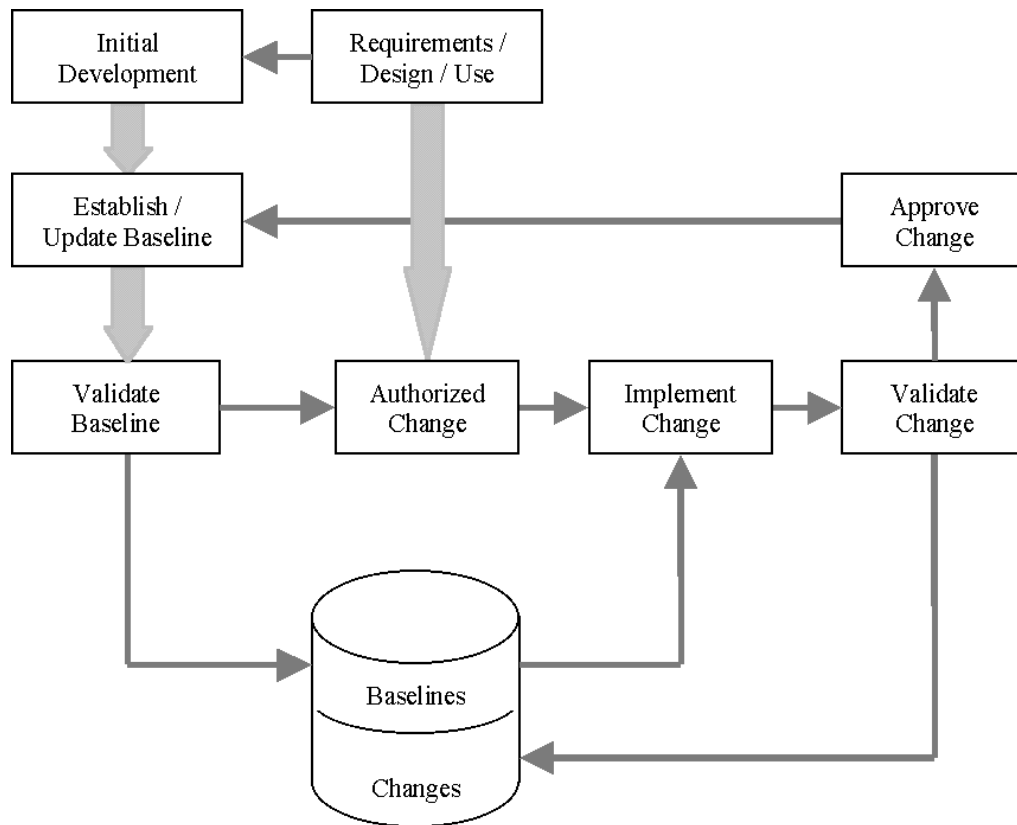


Fig.3: Configuration Management Overview [4]

The main rules of initialising configuration management into the development process are:

- Development of a configuration management plan
- Fixing common development policies
- Establishing requirements for version control

These rules are necessary because the process of implementing CM into the company's development activities is difficult and wrong decisions and implementations decrease programming staff effectiveness. Configuration management implicates the establishment of an appropriate process environment.

In the next part we discuss the above mentioned rules and give a short introduction to our considerations for the implementation of configuration management in our software development process.

### ***Configuration management plan***

Some effort is required imbedding CM into the software development process [9]. Consequently the CMS should be

- applicable to the environment,
- acceptable for developers and their tools,
- measurable improving the software development process,
- improving the efficiency and quality,
- improving the actual support for the quality system.

According to these points, the implementation of a configuration tool is highly complex and requires a lot of process experiences and detailed knowledge about the software development process in the company. To implement a configuration plan solving the problems above, a lot of standards have been established for use within a large segment of industry (e.g. IEEE Standard for Software Configuration Management Plans (IEEE Std 828-1990), DoD Software Development Plan Item Description (DID) associated with DoD-STD-2167A). These standards help a CM administrator to establish a well defined and effective CM. Without going into the depth of the standards contents the following steps are necessary [2], [9]:

- performing change control
- using different configuration identifications schemes
- using status accounting reports and their implementation requirements
- writing "procedures" within the configuration process

By following these steps the administrator is able to manage the configuration process properly. These plans contain a various amount of information about the development process and specifications. Furthermore the administrator is able to perform activities during the development process -- called "*procedures*" -- which describe all processes involved within the development process. Guidelines could be introduced for configuration management plans but are out of keeping in this paper.

### ***Common Development Policies***

Software development policies and procedures differ widely from organisation to organisation, but share a common goal: improving the quality and time-to-market of the software under construction. The following section establishes some important policies, which must be included into the configuration management process [1]:

- **Documentation of Changes** — All changes to sources must be recorded.
- **State Transitions** — The system must track the progress of each source file through the official approval stages.
- **Released Configuration** — All the versions that went into the building of a Release – and only those versions – must be marked with a defined label
- **Work on Bugfixes** — Fixes to a past release must be performed in isolation, starting with the exact configuration of versions that went into that release.
- **Freezing Certain Data** — Public header files may not be changed until further notice.
- **Enforcing Quality Standards** — Source files may not be checked-in unless they pass the quality metrics.
- **Requirement tracing** — Each source code module should have a pointer to an associated design document.

### ***Version Control***

The basic requirement for a software configuration management system (SCM) is *version control* – maintaining multiple versions of software development objects, termed *elements*. An element is a file or directory for which a SCM maintains multiple versions. The versions of an element are logically organised into a hierarchical *version tree*, which can include multiple *branches* and sub-branches[1].

The following features should be implemented for version control:

- The SCM should automatically assign integer **version numbers** to versions.

- Each version can also have one or more **user-defined version labels** (for example, REL1, REL2\_BETA,REL2).
- The SCM should support **multiple branching levels**.
- The SCM should use a **checkout-edit-checkin** model to manage the growth of elements' version trees.
- For each version of an element, **attributes** should be defined, to insure the process for bug tracking, quality control and requirement tracing.
- **Hyperlinks** should be established to connect two objects. inheritance, dependencies between elements and merging between versions should be supported.
- The SCM should be supported **Triggers** which implements "monitors" that tracks development work. For user-defined operation the SCM *fires* automatically user-defined trigger actions. Triggers can help to document the development process, improve communications within the development group, and implement process-management policies.

### ***Test environment***

The process to establish test environments is complex and difficult [6]. The test process in the scientific way, for example, is recognized by "*search for errors*". The management, on the other hand, confines the test process as "*prove that it works*". Therefore the test manager has the task to satisfy both of these directions. Another point is the consideration of given budgets and schedules, prepared by others, by people who are filled with the optimism of the developer — "*Everything is going fine*" or the urgency of management — "*Get this program out of the door. Now.*". To fulfill these different goals the test manager must establish a lot of process improvement. A well-defined software development process, consideration about planning, designing efficient test cases, executing the tests in an efficient way using automation and tools, supports the test manager in solving these problems.

For the DTK, the following tasks are necessary to establish a test environment with solving the rules mentioned above:

**1. Creation of a test manual**

This document describes general procedures for implementing a test structure.

It contents:

- detailed information about test concepts
- descriptions to organizing and coordinating the test process
- the test plan
- involved persons and organizations
- used procedures
- used test tools
- documentation requirements

**2. Using GQM methods to determine metrics**

These metrics are recognized by a lot of test tools. A external company will be measure these results and compare this with the previous measurements

**3. Implementing the contents of the test manual**

- Creating a test plan
- Assigning the test tasks to the test staff
- Selection of a test tool
- Involving the test tool into the test environment
- Development of first functions and add this to the test process

**4. Using Static Tests** for proving the size of modules, complexity requirements and the enforcement of development policies.

- Proving the first test results
- Modules and function which are not passed these tests will be returned to the developer with a detailed error report.

**5. Modules or Function with a higher complexity** as described in the test plan concepts will be forwarded to a code reading session.

- In the code reading session the test crew are generate test patterns for critical combination task supplementary to the normal specifications.
- Generating test pattern described in the specification.

**6. Using Dynamical Test**

- Proving the first results
- Modules or functions which are not passed these test will be returned to the developer with a detailed error report.
- Modules with errors will be tested again with all test patterns.

**7. Functional Tests of the customer of the products.**

In the previous section we described the process improvements that we recognized for the test process in our company. But these tasks could not satisfy the entire test process. Obviously the software development process influence the test process to a high degree. A good solution to solve this problem is the combination of the development and test process. The V-Model gives best result in managing this task by preparing test cases during the specification and design process (Fig. 4) [6]. Solving the test process with these features and planning the rules given above will decisively improve the software test process.



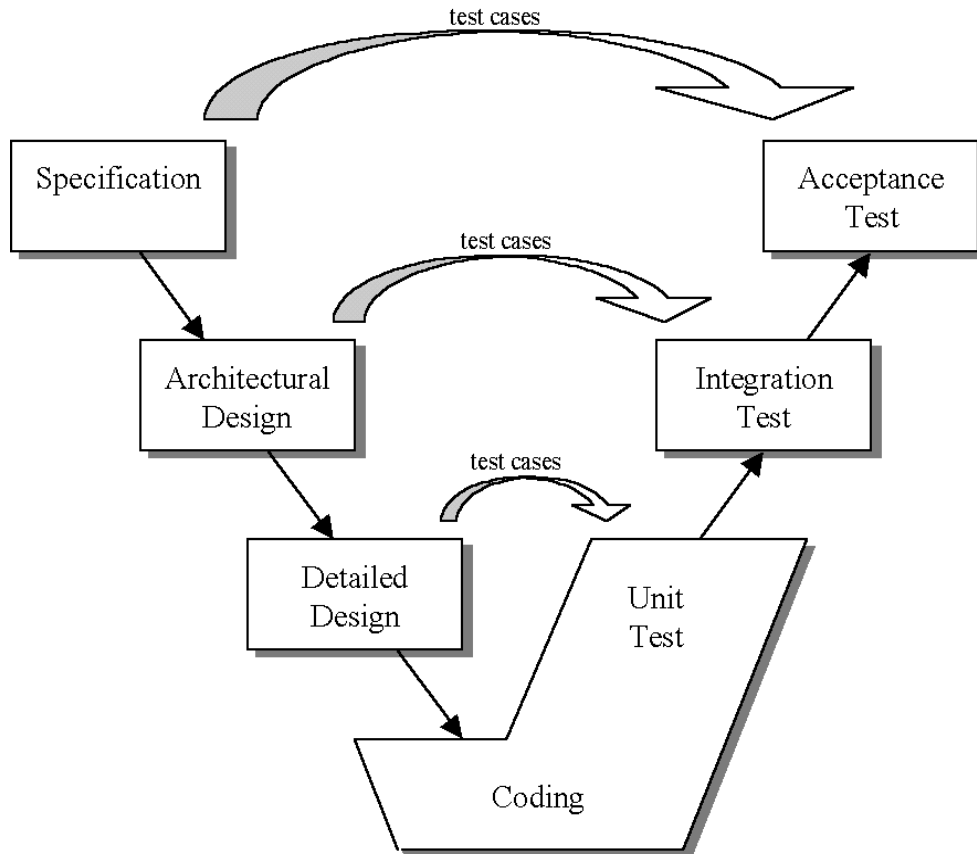


Fig.4: Testing process involved into the development V-Model

## Conclusion

This paper describes early results from ESSI Project N°24,078, IMPACTS2, in which test as well as configuration management methodologies and tools are evaluated with respect to their impact on process improvement in the course of software development.

It has been shown that configuration management is particularly well-suited for the software development process. It has been discussed, that in order to use the features of configuration management the software development process is less error-prone. One step implementing configuration & change management is the necessity of common development policies. These policies and requirements should be understood as valuation criteria's for establishing CM into the software development process. Obviously this rules are not common for all companies. For managing a software project life cycle as shown in Fig.5 these rules support the production considerably. It is recognised that the implementation of CM into the company's development activities is difficult. Therefore a configuration management plan should be established for a well-defined and documented process. Configuration management implicates the establishment of an appropriate process environment.

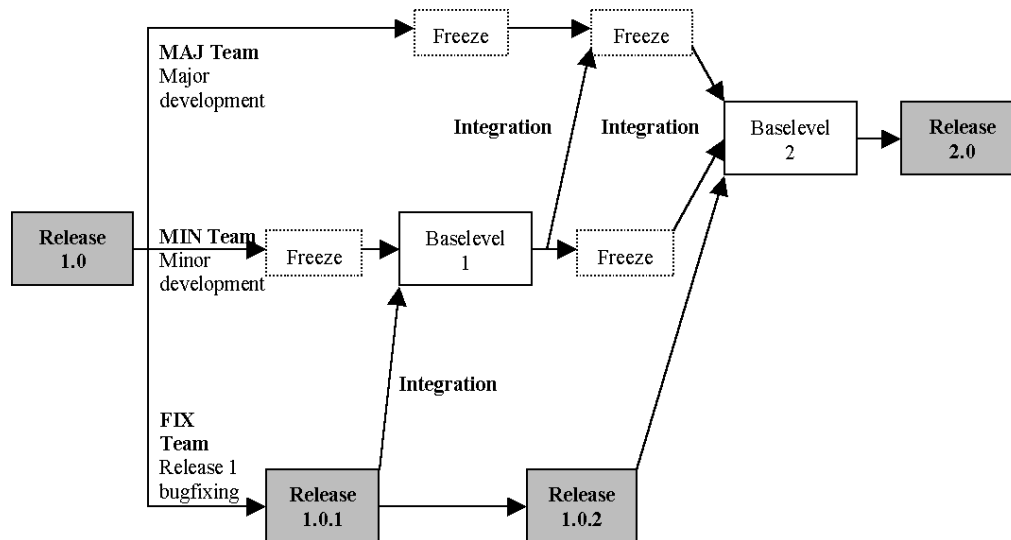


Fig. 5: Project plan for version and release improvement [1]

Furthermore the paper introduces into the implementation of test process improvements. We have learned that software test is more than debugging code or to make an acceptance test on the end of the developing process. But to use only a tool to have an automatic unit test is not enough. The most important things are the acceptance of all people who are involved in the developing process and the discipline of these people. Everybody knows, for example, that he has to document the tests he had made but only when he knows the reasons of this work and who must understand his document he can do it well. Also the creating of an executable test process is a very important point. In this process every involved person has to know which position, which interfaces and which task he has.

The DTK will be able to measure the improvement of configuration & change management and test processes by analyzing metrics which has been performed by our subcontractor. These metrics, determined by a GQM method defining measurement goals, should measure our effectiveness establishing the new processes. Final results will be expected in the near future.

This paper reflects the results of the PIE project at the time of preparation of the paper. This has taken place considerably earlier than its actual publication and presentation. As the project did only commence in March this year, detailed results of the project could not yet be reported. The authors will be able to present more details on the conference itself and in later publications in the course of the project.

## References

- [1] PureAtria, ClearCase *Concepts Manual*, Document Number 4000-002-C, PureAtria, 1996
- [2] Software Engineering Institute, *Configuration Management Plan – The Beginning to your CM Solution*, Software Engineering Institute, 1996
- [3] Alex Lobba, *Team Development Overview*, SilCom 1996
- [4] Watts S. Humphrey, *Software Configuration Management in: Managing the Software Process* p113-246, Addison Wesley, Software Engineering Institute, 1989

- [5] H.D. Rombach, *Practical benefits of goal-oriented measurement*. In Proceedings of the Annual Workshop of the Centre for Software Reliability, pages 217-235. Elsevier, Sept. 1990.
- [6] Bernd-Uwe Pagel, *Software Engineering*, Addison Wesley, 1994
- [7] Thomas C Royer, *Software Testing Management*, PTR Prentice Hall, 1993
- [8] W. Rigg et.al., *Ovum Evaluates: Configuration Management Tools*, Ovum Ltd., UK (1995)
- [9] ANSI/IEEE Std 828-1990, *IEEE Standard for Software Configuration Management Plans*, IEEE 1990

## CV Authors

### CV Harry Debler

Harry Debler studied Technical Informatics at the School of Engineering Lübeck Germany and finished his studies in 1985.

From 1985 on he has been involved in many different industry projects dealing with software engineering and process improvement. He worked 7 years as a development engineer in the sections of Information Technology and Transportation. In this time he was engaged with designing of safety critical software for railroad control systems, developing computerized test equipment, network design (DATEX-P), realisation of tools for configuration management.

In the last years he worked as a Quality Assurance Engineer at the DTK. In this position he was mainly involved in projects dealing with Quality-, Inspection- and Certification. He was working on large SW-test-projects for example for the German flight control DFS. Also he was engaged with development of quality management system guidelines for use in software development departments, Implementing quality reviews, assessing existing quality assurance activities in large companies. Applying DIN/ISO/EN norms to software development, preparation for Bootstrap-Quality assessment.

### CV Marcus Begemann

Marcus Begemann studied Technical Informatics at the School of Engineering Lübeck Germany and finished his studies in 1992.

From 1992 on he has been involved in different industry projects dealing with software engineering and process control. He worked as a development engineer in the sections of Information Technology, Transportation and Sensor Technology. He was engaged with designing of safety critical software for railroad control systems, developing computerized test equipment, realisation of methods and tools for test of relay-based circuits and software.

In the last years he worked as project leader for software engineering and test projects at the DTK. In this position he was mainly involved in projects dealing with interlocking system technology, quality management in the environment of safety-technology. He was the project leader of large SW-test-projects for example for the safety kernel of the train control system of the Spanish high speed train (AVE). Also he was engaged with development of guidelines for railway systems use in examination authorities.

## **Panel Chair: Dr Miklos Biro**

### **Experts Panel**

Christophe Debou, Amor Dominguez, Alec Dorling, Richard Messnarz, Bernhard Posch, Hakan Wickberg

## **Titel: Key Success Factors in SPI and Collaboration Opportunities with Central and Eastern Europe**

Contents:

- How could an East - West collaborative enterprise partnership look like
- How important are skills and human resource transfer to create a win-win partnership and to gain success
- How could we exploit the funding and cooperation options offered by the EU as presented by Dr Klaus Woelken the day before
- What are outsourcing requirements of big Western firms which must be satisfied before giving any contract to an east firm (ideas about a risk management scenario)
- Further issues as proposed by the audience during the tutorials and the conference

Discussion Procedure:

For each of the topics each panelist gives a small comment. Then the audience can contribute themselves with know how and can ask questions. As we believe that the audience are experts as well (like all speakers) their contributions have the same weight.

All discussions are recorded and two weeks after the conference a summary is distributed to all attendees.

### **PLEASE ACTIVELY CONTRIBUTE**

If you see any interesting issue, do not hesitate to leave a note at the registration desk for Dr Richard Messnarz who will insert this issue into the panel discussion.