



**PROCESS IMPROVEMENT EXPERIMENT
FINAL REPORT**

InCoMM

**Experimental Introduction of a
Configuration Management Model for an
Open Control System**

ESSI Process Improvement Experiment No. 27546

Roland Beeh, ISG

02.03.00

Version 1.32

Table of Contents

1. Executive Summary	3
2. Background Information.....	3
2.1. Objectives	5
3. Starting scenario	5
3.1. Experiment context	5
3.2. Company Context	6
3.3. Baseline project context	7
4. Experiment description.....	8
4.1. Overview	8
4.2. Phases of the experiment.....	9
4.3. Consultancy during the experiment.....	11
5. Resulting scenario.....	11
5.1. Technical	11
5.2. Business impact.....	12
5.3. Organisation impact	13
5.4. Culture impact	13
5.5. Skills impact.....	14
6. Key Lessons learned.....	14
6.1. Technological point of view	14
6.2. Business point of view	15
6.3. Strengths and weaknesses of the experiment	15
7. Conclusions and Future Actions	16
8. Glossary.....	18
9. References	20
Annexes	21
Figures	21
Tables	26
CM-Tool Overview.....	30

1. Executive Summary

This report describes the final state of the PIE project InCoMM (Experimental Introduction of a Configuration Management Model for an Open Control System) at the end of execution. Within the scope of this PIE the software development process of ISG was improved by the introduction of a CM model according to the guidelines of the V-Model /1/.

The key lesson learned is the realisation of the necessity of configuration management (CM) in the software development within an open and modular software system. This necessity derives not only out of the open and modular design of the software but also out of the way of further developing the existing software parallelly by about 20 software engineers. As ISG delivers source code and documentation as a commercial of the shelf software (COTS) supplier to the customers, a further reason for the necessity of an efficient CM is the continuously development of versions and variants.

Therefore the results of the project are of importance for all software houses, further developing one existing software product parallelly by different software developers as well as for software houses, providing COTS systems to the market. The two initial project phases, the tailoring phase, necessary for the adaptation of CM to ISG's special requirements and the phase of setting up the CM activities, necessary for the integration of CM to ISG's software development process, were completed up to mid-term of the project. This included the introduction of the CM-Tool ClearCase from Rational. The finally executed phase of experimental work with CM revealed strengths and weaknesses of the set up CM activities as well as of the appropriateness of ClearCase for ISG's needs.

With the set up CM activities the project objectives could be achieved. On basis of the COTS system the co-operation with customers and partners could be improved. The improvement of the interaction between Quality Assurance (QA) and CM resulted in an decrease of the number of bugs identified by customers. Increasing the number of reused software-elements the development of versions and variants can be done more effective.

The set-up of a working CM system resulted in an increase of on-time deliveries and on faster responses on support requests of customers. This mentioned facts lead to an improved customer-supplier relationship and therefore to a consolidation of ISG's position on the market..

2. Background Information

The ISG develops and sells software components for the assembly of real time kernels for numerical controls (NC). The software developed by ISG is used for the numerical control of various machine tools such as milling machines, lathes, wood-working machines, textile-working machines, and robot systems. The ISG's customers are located in various countries and are either NC vendors or machine tool vendors, who design their own NC using software components of ISG. ISG provides also NC software to partners of public funded research projects (funded by regional and national institutions). Within such research projects technological improvements of the NC functionality are the objectives. The partners, provided with NC software in this research projects, are research institutes of universities.

As ISG delivers source code and documentation, the customers and partners are enabled to extend and configure the control system according to their own technological requirements. This comprises the adaptation of ISG's NC software to a particular operating system (e.g. VxWorks, OS/9, AIX, etc.), to the particular control system environment (PLC, HMI, Drive-Interface, etc.) as well as to the machine specific conditions (number of axis, kinematics, dynamic behaviour, etc.). According to the contract, the customers and partners are required to feed back any newly designed or changed software modules to the ISG.

This fact forced ISG to introduce a CM method for tracking and documenting software throughout its life cycle and to ensure that changes can only be made in a controlled manner. Therefore ISG introduced within the InCoMM-PIE the efficient CM Model, known as a sub-model of the V-Model /1/. V-Model is a Lifecycle Process Model, originally developed to regulate the software development process within the Federal Administration of the Federal Republic of Germany. The V-Model is subdivided into the sub-models (fig. 1 of the annexes)

- Project Management (PM),
- System-Development (SD),
- Quality Assurance and
- Configuration Management.

This subdivision shows that within the V-Model the CM field of activity is regarded as an independent field of activity which equates with the fields PM, SD and QA.

The sub-model CM contains the set up of the four CM activities CM planning, product and configuration administration, change management, and CM services (data backup, CM reporting, etc.). This procedure ensures that all products are uniquely identifiable, that interrelations and deviations of different versions or variants of a configuration remain evident and that any product changes can be made only in a controlled manner. Therefore ISG means by the term CM all methods, tools, and CM aids, which support the development and maintenance of a software product as a succession of controlled revisions and supplementation based on safe process results. In this context, the ISG placed the emphasis on the controlled aspect of the CM activities, which were expected to achieve the objectives of InCoMM.

The set-up of all CM related activities within the InCoMM-PIE were executed following the technical rules called "Tailoring", defined within the V-Model's documentation. Beside this formal Tailoring an analyse of the current software development process was executed in order to find out, which CM relating activities and products were already existing at ISG. Furthermore ISG's special requirements on an effective CM were discussed and worked out. Basing on this summarised requirements and the analyse of the current software development process a comparison with the results of the formal Tailoring was possible.

According to the V-Model CM comprises the activities

- CM Planning (CM-1),
- Product and Configuration Administration (CM-2),
- Change Management (CM-3) and
- CM Services (CM-4)

as shown in figure 2 of the annexes. The activities CM-1 and CM-4 are considered as accompanying activities, initialised once for a project. CM-2 and CM-3 are the main activities within the field of CM. This activities need ongoing effort during the software development process.

The product and configuration administration (CM-2) comprises the administration of products (source modules, documents, test patten, etc.), configurations (assemblage of several products) and rights (user authorisation). The administration comprises the initialisation (registration and assignement of identification) as well as the archivation of products, configurations and rights.

The change management (CM-3) monitors the change procedure. This includes the recording and evaluation of change requests, the decision about change procedure and initialisation of change and the

recording of change reports. Changes can be done either due to bug-fixing, or enhancement or due to functional extension of the existing software.

2.1. Objectives

The number of existing software elements at ISG is continuously increasing due to functional extensions of the existing NC software construction kit and due to branching out (physical copying of software elements) new customer specific variants out of this construction kit. The objective to increase the number of reused software-elements is therefore of an utmost organisational and business importance. This goal was achieved by changing the branching methodology. So far the complete NC construction kit was duplicated physically at the beginning of a functional development. This led to an enormous number of parallel existing identical software elements. Introducing a CM-Tool (ClearCase from Rational) that support the definition of views, the number of branched software elements could be reduced on the number of software elements actually affected by a development. In this way functional developments can be merged into the NC construction kit with little effort. The redundancy of software elements can be avoided. The measurement of this objective was executed by evaluating the reduction of software elements between the beginning and the end of the InCoMM-PIE.

The improvement of the co-operation with customers and partners was one of the main business goals. This improvement bases on an increase of on-time deliveries and on faster responses on support requests from customers. Due to the use of a clear identification system and the possibility to get a survey of software elements relating to a specific customer or to a specific variant of the NC by the employment of an appropriate CM tool, this aims were achieved. The measurement of this kind of goals was carried out by recording time values (e.g. time necessary for compilation of customer specific NC variants) and by evaluating questionnaires handed out to customers and partners as well as to ISG's software developers.

The improvement of CM planning and the introduction of a clear identification system lead to a more efficient application of quality assurance (QA) activities. Tests can be coordinated with the progress of development activities. In this way only configurations of the NC that are ready for analytical QA measures (e.g. tests or reviews) are provided from the system development to the QA field. The improvement of CM planning made the bug tracking easier as the first appearance of a bug within the NC system can be clearly identified. Due to this coordination between QA and CM as well as the improvement of bug tracking the number of bugs delivered to customers or partners could be reduced. This reduction was measured by an ongoing analysis of the ratio of bugs internal found and bugs reported by customers or partners.

The objectives are summarised in table 1 of the annexes.

3. Starting scenario

3.1. Experiment context

The purpose of the experiment was the optimisation of the CM related activities of the software development process at ISG. This was done by the introduction of an established and precisely described CM method the so-called CM sub-model of the V-Model. To ensure the suitability of this method for ISG's demands the required CM products and activities, provided within this method, were selected in a so-called Tailoring phase. Due to an adaptation of the selected activities and products to the already existing CM activities and products within ISG's software development process, the necessary extensions and optimisations of ISG's CM activities were executed.

As an effective CM is impossible without the support of an appropriate CM Tool, a CM Tool (ClearCase from Rational) was introduced to the software development process within this PIE. The project had therefore the two main focuses: Introduction of a CM method and establishment of a suitable CM tool.

Before applying for the InCoMM PIE an investigation of the current software development process within the ISG was executed. This exposed several deficits in the analytical, organisational as well as constructive fields of quality assurance. By dividing these deficits into clearly outlined problems, the ISG intends to overcome these deficits on a step by step basis. Besides the introduction of project and quality assurance plans and a further development of the test strategy, a further step towards improving the software development process was the introduction of a configuration management method as a part of the constructive quality assurance field.

The core of the InCoMM PIE team consisted of two software engineers of ISG and a research engineer of the subcontractor FISW Steuerungstechnik GmbH. This team was responsible for the evaluation and adaptation of CM methods and tools to ISG's software development process. During an experimental phase, while this CM method and tool were set up to the baseline projects, the software engineers, involved to this baseline projects, were also part of the InCoMM PIE team. Fourteen of ISG's twenty-one software developers were involved in development activities for this baseline projects in different periods of time and scopes. So all of this software developers came into contact with the InCoMM project and the process improvement activities in the field of CM.

The experience of the experiment team members in software engineering reaches from experience gained within university projects to the experience of several years of software engineering in an industrial environment. Experience in process improvement was restricted to some team members, who carried out some small projects in the field of QA.

3.2. Company Context

The ISG develops and sells software components for the assembly of numerical controls (NC). The software developed by ISG is used for the numerical control of various machine tools such as milling machines, lathes, wood-working machines, textile-working machines, and robot systems. The ISG's customers are located in various countries in Europe and are either NC vendors or machine tool vendors, who design their own NC using software components of ISG.

The type of contract the ISG usually makes with its customers, is based on a conception, that supports above all the configuration and the extension of the functionality of the ISG's "open" control system by the customers on their own. Therefore the ISG delivers source code and documentation to their customers, which enables them to extend and configure the control system according to their own technological requirements. According to the contract, the customers are required to feed back any newly designed or changed software modules to the ISG.

Due to the continuously increasing number of versions and variants of software elements, relating to ISG's NC system the CM related effort in the daily work of software development is permanently increasing. This situation encouraged ISG's software developers to optimise the software development process in the field of CM. So the CM related process improvement activities as well as other process improvement activities executed already before were initiated by the software developers on their own.

ISG's organisational structure has a flat hierarchy. So the communication style within the company is uncomplicated and direct between software developers as well as between software developers and the management. This fact was and is very helpful for unbureaucratic process improvement activities parallel to daily work activities.

3.3. Baseline project context

The open control system, which was used as baseline project for this PIE, is the main product of the ISG. The software, which is completely written in the high level programming language "C", is used for the numerical control of different machine tools. This control system is developed in a modular way, that allows the ISG and its customers to configure and extend it in an easy way. Due to these configurations and extensions, there were seven different customer variants and three variants used internally by the ISG for research and development purposes at the beginning of the InCoMM project, which need to be maintained. The existing versions of the control system differ among other things from the operating system (e.g. VxWorks, OS/9, AIX, etc.) the control system is running on.

Each of these variants contains about 450 source code files with approximately one million lines of code, which requires about 45 megabyte of memory capacity. For the documentation of the basic modules of the controls system as well as the technology specific extensions, about 2000 documents in English and German language exist. In addition to this there are currently about 2000 NC part programs in use for testing the control system's functionality. About seventeen of ISG's twenty-one software engineers are developing and maintaining all these software elements belonging to the control system.

For the purpose of the experimental character of the InCoMM project ISG used its so called central development variant (CDV) of the NC as baseline project for InCoMM. That means this CDV was taken under control of the CM system, which was set up within the PIE. Furthermore two development projects were carried out within the InCoMM PIE. The control systems for both projects were branched out of ISG's CDV. So all CM related activities within this development work were executed according to CM guidelines and with the newly set up CM tool ClearCase.

One of the mentioned projects comprises the development of a control system for laser-based machining and a control system for electrical discharge machining. Both control systems are basing on ISG's existing control system which is extended by new technological specific software modules. The control software for this systems is extended in a way, to run on different real time OS. This extension was a main goal of the baseline project and resulted in additional demands on the CM due to the additional variants of the NC to be managed. The used real time OS are VenturCom and ReadySys, which are both basing on Windows NT with software extensions, as well as VxWin, which is a solution with VxWorks and Windows NT both running on one CPU using. This second solution requires a passive additional hardware, that reacts on normal interrupts from the ISA-Bus in order to generate a Non Maskable Interrupt instead. All of this OS specific variants of the NC are running on ISA IPCs with Intel CPUs.

The second project contains the development of the so called three dimensional tool geometrical compensation. So far ISG used a two dimensional tool radius compensation for NC of milling machines. Due to new functional requirements of machine tool builders the existing compensation needed to be extended by further compensation algorithms in different software modules. The reuse of existing software elements for this project is of a high importance. Therefore this development activity was very suitable to be done under the control of the CM.

The NC variant is running on the real time OS TwinCat (Windows NT with software extension) on an ISA IPC with Intel CPU.

For both projects an effort of about one person year (PY) was estimated for the works related to CM as there were the specification phase, the coding phase, the integration phase and the test phase. In comparison to other developments at ISG these projects have a long planning period.

Since the experiment was attached to the baseline project there was no turnover in the baseline project team.

4. Experiment description

4.1. Overview

No process within ISG's software development was affected in a negative way by the execution of the experiment. As intended with the InCoMM-PIE the main improvement was obtained in the field of CM. The execution of the PIE had also positive side effects on the processes of software testing, system integration, software maintenance and PM. In this fields, the better administration of software elements due to the introduction of CM as well as the growing awareness for the necessity of process improvement came into play.

The main focus of the InCoMM PIE was on the introduction of a CM method into the software development process of ISG. This was done according to the CM sub-model of the V-Model /1/. As mentioned in chapter 2 this sub-model comprises four main activities. Within the CM-Planning (CM-1) the procedural framework (branching method, version number system, etc.) for CM was defined. The installation and set-up of the CM tool Clear Case to manage all software elements and configurations of the NC system was the main activity of CM-2. All activities concerning the change management (CM-3) are realised within the already existing intranet system LOTUS NOTES. The existing bug-tracking system was extended step by step in order to fulfil all demands ISG has on the change management. So all sequences within the change management (fig. 3 of the annexes) are represented within this change management system. CM Services (CM-4) as there are data backup activities or CM reporting were implicitly realised by introducing ClearCase to the infrastructure of ISG software development environment.

Realising the necessity of improving the field of CM ISG was looking for an established and precisely described CM method. V-model fulfils this requirements and was therefor considered to contain an appropriate CM method for ISG's purposes.

The common classification of CM-Tools described in literature and brochures distinguishes

- Version oriented tools (VOT),
- Developer oriented tools (DOT) and
- Process oriented tools (POT).

As the ISG has not yet defined formal and restrictive processes for software development POT are considered as unsuitable for ISG at the current situation. VOT do not support the feature to save different views of configurations. In order to fulfil the ISG specific demands on CM, using a DOT is necessary. The option of these tools, allowing to define views of configurations for individual developers or developing teams, who are accessing software elements of different variants and versions, software development can be optimised within a modular software system.

For the period of the InCoMM project the CM tool ClearCase from RATIONAL was chosen to be evaluated more detailed during this PIE. Different configurations for development purpose or bug-removal activities can be defined with this tool in a suitable way by defining different views. This definition of views is supported by assigning attributes to the individual software elements. Furthermore this tool offers the ISG enough possibilities, to describe ISG-internal existing but not yet formulated processes with the aid of mechanisms provided by the tool.

The features of ClearCase are summarised in table 2 of the annexes.

4.2. Phases of the experiment

The executed workpackages (WP) of the InCoMM PIE can be distinguished into two categories of WP. The first category of WPs comprises the phases of CM introduction to the software engineering process of ISG. These WPs are:

- WP-2 Tailoring,
- WP-3 Set up of CM activities and
- WP-4 Experimental Work with CM.

Each of these three key phases of the experiment concluded with a milestone and each of these phases depended on the respective phase before.

The main objective of WP-2 was the adaptation of CM to ISG's special requirements. To attain this main objective the formal requirements for an effective CM of ISG's control system had to be defined. Therefore the objective of the formal Tailoring (WP-2.1) was to find out these requirements according to the Tailoring procedure of /1/. The objective of the Tool evaluation phase (WP-2.2) was to find an appropriate CM tool, fulfilling ISG's requirements as well as standardised evaluation criteria described within /1/ as "Functional Tool Requirements". Within the following phase (WP-2.3) an analysis of the current software development process was executed in order to find out, which CM relating activities and products were already existing at ISG. Furthermore ISG's special requirements on an effective CM were discussed and worked out. Basing on this summarised requirements and the analysis of the current software development process a comparison with the results of the formal Tailoring was possible. The objective of WP-2.3 was to find out, which of the CM activities and products are relevant for ISG.

The main goal of WP-3 was to integrate the CM to ISG's software development process for the baseline projects. This integration was executed according to the Tailoring results elaborated in WP-2.

Within the first phase of WP-3, the set up of "CM Initialisation" (WP-3.1) a guideline for identifying versions and variants of ISG's modular control system was composed. Furthermore the installation work for the CM tool ClearCase from Rational was part of WP-3.1. Beside the installation and the configuration of the server application the baseline project concerning software elements were integrated in the ClearCase repository. The access to this data is done via the ClearCase client applications according to pre-defined views.

The set up of the "Configuration Administration" (WP-3.2) contained the optimisation of the directory structures of the existing source modules of the baseline project. This is done in order to identify the different configurations of the modular control system uniquely. So the source modules are organised in a way to identify application (operation system specific; machine tool specific; etc.) specific configurations of the modular control system without the support of a CM tool. CM tools are usually not available during the set up of machine tools in a shop floor environment.

The set up of the "Change Management" (WP-3.3) was executed by extending the existing bug-tracking within the internal used intranet system LOTUS NOTES. The sequences within the change management as proposed within /1/ are integrated within this system (fig. 3 of the annexes). The responsibilities for the decisions about change procedures and for the initialisation of changes are assigned to the circle of ISG's project manager as 'change control board'. This is important as within this group the know-how is available, whether changes have any side effects on other developments or projects.

Within WP-3.4, the set up of "CM Services" the data back-up activities as well as the CM reporting activities had to be optimised for the baseline project. As the data back-up activities at ISG were already executed in a sophisticated way this activities were just extended in a way, to bring the new InCoMM-PIE specific data (ClearCase repository) also into this back-up system. Information about all software elements (change reports) of the control system are stored and available via the CM tool ClearCase. New attributes for the software elements (e.g. according to specific customer information or according to phases of the software development process), necessary for an effective reporting, are defined and assigned within ClearCase.

WP-4 described the experimental work and development of the baseline project under the control of the set up CM system. So the main objective of WP-4 was to prove the suitability of V-Models CM-Model in ISG's software development process. Furthermore this experimental phase contained the opportunity to evaluate the performance and suitability of the CM tool ClearCase under conditions, the software development process at ISG requires.

The second category of WPs with long term WPs contains

- WP-0 Project Management,
- WP-1 Co-operation with other ESSI projects,
- WP-5 Measurement Activities,
- WP-6 Training Activities and
- WP-7 Dissemination Activities.

WP-0 and WP-1 covered the complete period of the experiment and had a pure organisational character. WP-5, WP-6 and WP-7 were accompanying WPs for the WPs of the first category. This WPs took several months.

Beside the postponement of the official end date from 31.12.1999 to 29.02.2000 no important changes from the 1st version of the workplan were done. The postponement was necessary as ISG started projects with new customers. With this projects the internal dissemination of the InCoMM-PIE was enforced. Therefore the CM part of this projects is carried out according to the existing results of InCoMM. Due to this the InCoMM team members are strongly involved in this new projects. All WPs and corresponding deliveries of deliverables and reports originally planned for the time period between 01.05.1999 and 31.12.1999 were postponed for 2 months.

A delay of the first category of WPs (WP-2, WP-3 and WP-4) with the corresponding Milestones (MS-1, MS-2 and MS-3) resulted out of the time-consuming tool evaluation within WP-2 as well as out of the mentioned postponement. Although ISG expected the tool evaluation to be time-consuming, the estimated effort and time for this purposes had to be limited in the project plan on the instruction of the EC. A comparison of the original and of the final project plan is shown in figure 4 of the annexes.

Training activities were executed within the scope of WP-6. All software developers, related to the baseline project, were informed about the goal and the way how to introduce the CM activities to the software development process of ISG. This two-hour training was done by members of the core team of the InCoMM-Pie. All this software developers were also trained about the use of ClearCase within an in-house training. This one-day training was executed by a trainer from Rational, specialised on teaching the use of ClearCase. Furthermore two members of the InCoMM-Pie team participated in a special two-day course for administration of the ClearCase server application. This training took place at Rational's training facilities in Munich.

4.3. Consultancy during the experiment

External consultancy was used by FISW Steuerungstechnik GmbH. In this way ISG could use the experience and knowledge of FISW Steuerungstechnik GmbH in the field of software engineering for automation systems. The support of FISW Steuerungstechnik GmbH was particularly used for defining and initialising the configuration management (CM) activities. Due to the strong involvement of FISW Steuerungstechnik GmbH by the measurement activities, FISW Steuerungstechnik's software engineers could continuously judge the current state of InCoMM and could give advice towards the success of InCoMM.

FISW Steuerungstechnik GmbH and ISG co-operate in several research and development activities as well as in public projects. Among other things FISW Steuerungstechnik GmbH supports ISG in carrying out tests with ISG's numerical control and in developing new test strategies to reach a highly thorough examination. Furthermore FISW Steuerungstechnik GmbH was already involved in the currently established assessment of ISG software development process (e.g. by evaluating ISG's software error data base). Out of the past collaborations FISW Steuerungstechnik GmbH gained a lot of insights to ISG's software development process.

The effort expended for the consultancy of FISW Steuerungstechnik GmbH amounted to 100 person days (PD). This effort was split up into two parts with different labour rates. 50 PD of consultancy were invested for 'Expert Advice' and 50 PD were invested for 'Normal Project Work'.

5. Resulting scenario

5.1. Technical

The main methodology-related results were the possibility to identify uniquely each software element of the baseline project at any time and the ability to reproduce each modification of this software elements as change activities are controlled and reproducible procedures. Apart from this the fundamental understanding for the necessity of a clearly defined CM within the software development procedure was a further result, not quantitative ascertainable.

The objective to increase the number of reused software-elements has been approached by the introduction of the CM tool ClearCase. Due to this the branching methodology has been improved in a way, that only those software elements need to be branched as actually affected by the respective development. During the execution of WP-4 this influenced not only the work within the baseline projects but also the projects, executed already according to the current results of the InCoMM-PIE. The increase of the number of reused software elements respectively the decrease of branched software elements for development purposes for this projects is illustrated in figure 5 of the annexes. Since the introduction of ClearCase in April 1999 the number of parallel existing branches increased from 27 to 175 in February 2000 (increase of about 650 %) at the end of WP-4. In spite of this increase the increase of the number of source files could be limited to about 160 %. Using views for customer specific variants, which are defined with the aid of the CM tool ClearCase, the time necessary for the compilation of deliveries as well as the time required for responses on support requests could be reduced.

The improvement of CM planning is realised with the aid of the introduction of ClearCase as well as due to the utilisation of a magnetic drawing board. Particularly during the execution of WP-4 this board turned out to be an ideal discussion medium for development planning regarding the CM related activities as there are branching and merging. In this way the responsible for the several development activities can co-ordinate and plan development and integration steps as well as QA steps. The results of this planning are converted into activities within the ClearCase tool. Due to the improvement of CM

planning the time effort for integrating parallel existing development branches could be significantly reduced within WP-4 to 10 % of the initially needed time.

Within the year (Jul 97 - Jun 98) before the InCoMM-PIE begun 25,73 % of all found bugs were detected by customers. Since the begin of the CM optimisation in July 1998 this percentage has reduced to 17,93 % until May 1999. The current percentage of bugs (Jan 2000), found by customers, is 24,25 %.

The experimental work (WP-4) using the set-up CM activities and in particular the CM tool ClearCase led partly to the reduction of the bugs. The number of integration errors arising during merge activities could be reduced, the certainty for the software developer to have merged all necessary files in the right way could be increased. A quantitative statement about the concrete reduction of integration errors since the set-up of CM activities and the use of ClearCase is not possible, as this number was only recorded, when the InCoMM-PIE was already at the end. A significant reduction of this integration errors is expected, due to the statements of software developers and project managers of ISG, who were intensively dealing with merge activities during WP-4.

The InCoMM-PIE had positive side effects on several processes beside the CM. In order to keep the effort for NC-software testing within acceptable time and economical limits, tests need to be executed in the form of regression tests [3]. For this kind of test a test version of the NC-software is compared with an already tested version as reference. Therefore the optimisation of the version control, as it is realised within the InCoMM-PIE, led to a direct improvement of the test procedures. This improvement reflected mainly during WP-4 in the improved reproducibility of test procedures.

The PM profited from the InCoMM-PIE in different ways. First of all the CM planning serves the PM as information source. The PM can trace back the stage of development of the different project and can therefore draw conclusion from the stages of development to the current project status. This allows an ongoing cost-benefit analysis of the development projects. Secondly the PM starts to accept and plan the effort for CM activities since the begin of the execution of the WP-4. So far the CM effort was part of the daily work of each software developer at ISG and therefore not ascertainable. Now the effort necessary for CM is a calculable working package during the software development process at ISG. Already at an early time after the initiation of WP-4 other projects at ISG, which did not belong to the baseline project, started to plan the CM work in the same way it was done within the InCoMM-PIE. As basis for planning purposes about 10 % of the overall development effort is calculated for CM work. The technical impacts are summarised in table 3 of the annexes.

5.2. Business impact

The main business impact of the InCoMM-PIE is the possibility to support more customers faster and in an easier way with NC-software. The acquisition of new customers for ISG's NC construction kit was only possible on the condition of a working CM as it is set up with the InCoMM-PIE. The administration of several parallel existing customer specific variants is only possible with an effective CM. A reduction of the time necessary for responses on support requests could not be detected neither within the baseline projects nor within the projects, applying CM already according to the InCoMM-PIE results. The reason for this is, that the respective project managers had already personal CM organisation structures. However the time necessary for delivery a new version of the NC to a customer could be reduced within WP-4 to 25 % of the originally needed time.

The execution of the InCoMM-PIE led to a quality improvement of the software development process (as mentioned in chapter 5.1) and a quality improvement of the NC-software (e.g. due to test optimisation) itself. Due to this ISG's position in the NC market could be consolidated.

So far the CM effort was part of the daily work of each software developer, as mentioned already in chapter 5.1. With the initiation of the InCoMM-PIE these efforts became obvious and became part of the planning of the PM. About 10 % of the overall development effort is calculated for CM work as result of the InCoMM-PIE. Thus an organised CM seems to involve additional effort at first sight. The evidence of a monetary benefit of an organised CM reflects in the reduction of time necessary for CM related activities (integrating parallel existing development branches, providing new customer versions, etc.) as well as in the more reliable and therefore faultless execution of CM activities.

The business impacts are summarised in table 4 of the annexes.

5.3. Organisation impact

With reference to /1/ and /4/ InCoMM introduced CM to the software development process at ISG to be an independent field of activity which equates with fields like QA, SD and PM (fig. 1 of the annexes). As this field was not defined so far, new roles and responsibilities were created at ISG. The newly established CM manager is responsible for the co-ordination of CM related activities. The CM manager also organises the interaction between the different fields of activities (CM, QA, SD and PM). Furthermore one responsible for the administration of ClearCase was established at the time, ClearCase was introduced within the InCoMM-PIE. Since it came to the decision, that ClearCase will be introduced as standard CM Tool for ISG, a second administrator for ClearCase was established. The tasks of this administrators are to maintain the tool and to provide the software developers with necessary views.

With the introduction of a more formalised change management (WP-3.3) the complete software development process at ISG is influenced. Bug removal activities and functional developments on behalf of a customer were already a controlled and reproducible process at ISG. Enhancements and functional developments within the scope of ISG internal projects were hardly registered and therefore not reproducible although this changes represent a large part of all change activities. Within WP-4 such kind of changes were considered and treated in the same way as bug removal activities. This involved changes in the way of work of all software developers at ISG.

5.4. Culture impact

Starting with the application for the InCoMM-PIE and particularly within WP-4 CM became more and more a point of discussion within ISG. The discussions about CM were not only dependent on the PIE on its own but it was a growing sensibility for this field of software development. In this way CM became almost as important as QA within ISG's software development process.

The main consequences according to the introduction of CM at ISG were and are based on the results of daily discussions among ISG's software developers (InCoMM-PIE team members and non-members) and on fast and pragmatic CM related changes of the software development process. The more formal introduction of a CM model as described in the V-Model was slowed down by the missing acceptance of the management for process improvement activities. The acceptance was missing as at first sight process improvement activities do not reveal any monetary benefit. Therefore the InCoMM-PIE work was frequently interrupted by customer projects. The missing general acceptance of the management prevented also an analysis of the management with the subject CM. The resulting mixing of CM, QA and PM tasks by the management resulted in indistinct formulated expectations of the InCoMM-PIE.

5.5. Skills impact

Due to the InCoMM-PIE the main knowledge was accumulated in the field of CM in general. This increase of knowledge was not restricted to the team members of the InCoMM-PIE but existed also more and more among the software developers of ISG. This knowledge was transferred from the InCoMM-PIE team members to the software developers not only due to the training activities but rather due to the discussions about CM during the daily work. The software developers gained first experiences with an organised CM by the adaptation of the CM to the baseline projects in WP-4.

Further knowledge arose from the use of ClearCase. The software developers involved in the baseline projects were trained by a professional ClearCase trainer of Rational in an in-house training. They gained also practical experience with this tool during their daily work in WP-4. Two administrators were trained in a special two-day ClearCase administration course at Rational's branch office in Munich.

The temporary knowledge about the CM tool market situation was of less importance as the CM tool market is continuously in motion. This motion results not only out of the technical further developments of the tools but also out of the permanent take-overs of the respective software houses. Therefore the CM tool overview in the annexes describes just the current state of December 1998. Especially according to the information about company names and company addresses.

General knowledge about the V-Model /1/ was gained during the complete project. This knowledge was limited to the field of CM but comprised also some knowledge about the fields PM and QA. This fact will be helpful for following process improvement activities in future.

The way of working of the software developers has changed as they started working according a more formalised CM method with the begin of WP-4. Due to the introduction of CM planning, change management and the CM tool ClearCase this changes were initiated. This new facts in the software development process and the growing acceptance of CM by the software developers led also to a more disciplined way of working within the baseline projects and within projects, executed already according to the current results of the InCoMM-PIE.

6. Key Lessons learned

6.1. Technological point of view

The main lessons ISG's software engineers learned during the project was the necessity of software developing under the control of CM. The software engineers accept to do all changes and developments at software elements only with the support of an appropriate CM tool (ClearCase). In this way they can guarantee, that the development and change process is a controlled and reproducible activity.

A further lesson learned from a software-engineering point of view was to make a distinction of the changes (executed at the existing NC software system) between different categories. Bug removal activities, enhancement, functional extensions and customer related development activities are the defined categories.

The growing understanding for the interaction between CM, QA and PM was also important from a technological point of view. As already mentioned in chapter 5.3 CM became an independent field of activity in the software development process at ISG. The resulting advantages for PM, that can use CM as information source (see also chapter 5.1), and QA, that is supported in regression testing (see also chapter 5.1), are other main lessons learned.

6.2. Business point of view

From a business point of view it was important to realise and accept the necessary effort for CM within the software development process. This is important for project planning as well as for cost estimation for development work.

Dealing with process improvement activities in a particular field as CM comprises also activities in other fields of possible process improvement activities. The completion of the project proposal and the project programme as well as the execution of the InCoMM-PIE project at ISG revealed a lot of possibilities for corrective actions within the software development process. In particular in the fields PM and QA some defects were realised. In both fields improvement activities has been successfully executed parallelly to the InCoMM-PIE activities.

Independent from CM a majority of ISG's software developers and project responsables recognised the advantages of process improvement. This lead to the conviction, that high quality of the product can only be achieved by high quality in the product development process.

The implementation of process improvement, in particular within the scope of an EC funded project, can be used as an advertisement for customer acquisition by the management. This is independent of the technical knowledge and understanding of the process improvement activity.

A further lesson learned during the execution of the InCoMM-PIE was the opportunity to improve the co-operation with customers and partners due to the introduction of a working CM system. Providing the software (source and documentation) with higher quality in shorter time as well as the reduction of time required for responses on support requests led to this improvement. The mentioned acknowledgement by the customers and partners for this improvement resulted in a realisation of CM's importance within the software development process.

6.3. Strengths and weaknesses of the experiment

The strengths of the experiment can be divided into general strengths and CM related strengths.

The general advantage of the experiment itself as well as of the completion of the project programme and the proposal was the growing consciousness for the software development process. This software development process at ISG was so far only considered to be restricted to coding (SD). Thus improvement activities were also restricted to the field of SD. For the fields of CM, QA and PM were neither responsibilities assigned nor detailed efforts estimated for project planning. This change for a greater awareness for this field is therefore particularly worth mentioning.

The benefit of an effective CM within the software development is already mentioned several times. The main CM related strength of the experiment is the execution following the V-Model /1/. Although the V-Model is written in a pure functional ("bureaucratic") way it contains detailed advises and helpful approaches for the introduction of CM. Especially the division of CM activities into sub-activities (see also chapter 2) was helpful for the understanding and introduction of CM.

The InCoMM-PIE was postponed in April 1999 for two months. This postponement resulted in a re-planning of all project internal milestones, reports and deliverables. The reason for this postponement was the involvement of InCoMM team members in projects with new customers. The advantage of this involvement was the realisation of this new projects according to the results of the InCoMM-PIE already existing at this time. So the additional experience, gained out of this projects, could be directly fed back to the InCoMM-PIE and could be used for the set up of the CM to the baseline projects. This fact emphasised ISG's software developers in general and the management in particular the importance of a real application of the methods and procedures, introduced within a process

improvement activity.

For the administrative part of the InCoMM-PIE this postponement resulted in shifting of all milestones and paper deliveries in the project plan and also in an additional PPR, that has to be provided at the end of the project. The main weakness of the experiment was the difficulty to prove the management the monetary benefit of an effective CM. As long as this does not succeed any process improvement activity will be slowed down by the management (see also chapter 5.3) and will only be supported by the software developers.

Therefore an alternative approach for the execution of the experiment - if it would be possible to repeat the experiment - would be a more detailed analysis of the software development process in terms of costs. A comprehensive consideration of all cost factors (software development effort, increase of market value, effort for bug removal activities, advertising appeal, etc.), that are influenced by the introduction of an effective CM, would reveal the necessity of CM from a business point of view. Such an analysis in advance of the project would have led to an acceptance of the project by the management.

7. Conclusions and Future Actions

The aimed project objectives of the InCoMM-PIE were reached. Conclusive quantitative statements are mentioned in the annexes (fig. 5, table 3, table 4) as well as in a separate project deliverable relating to measurement activities.

The evaluation of the suitability of V-Models CM-Model for ISG's software development process within WP-4 led to the two main results, which are not quantitative ascertainable:

- the possibility to identify uniquely each software element at any time and,
- the ability to reproduce each modification of this software elements as change activities are controlled and reproducible procedures.

Further objectives, described with the aid of numerical dimensions, were reached and could be measured after the conclusion of WP-4 (see also chapter 5) due to this two mentioned main results. The increase of the number of reused software elements is for example only possible, if the existence of all software elements is known and if all software elements can be uniquely identified.

The execution of the PIE and the experimental work with CM (WP-4) had and still has positive side effects on PM and QA. This conclusion describes the most important outcome of the InCoMM-PIE. The side effects resulted not only in methodological changes in the fields of PM and QA at ISG but they resulted also in a fundamental understanding of the tasks and responsibilities of this fields. This understanding was strongly developed during the execution of WP-4, when the set-up CM activities were applied in the baseline projects at daily work.

Therefore future actions will also conclude improvement activities of the fields PM and QA. A new proposal for an IST project is already submitted to the EC. This project will deal -in case of acceptance- with the improvement of test execution and test management.

A further "future action" as result of the InCoMM-PIE was already initialised while the project was still running in the experimental work phase (WP-4), when it came to the decision, that ClearCase will be introduced as standard CM Tool for ISG. This decision was done on basis of the experience gained out of the use of ClearCase within the baseline project work when ClearCase's suitability for ISG's needs was proven.

The execution of future actions can be done according to the guideline of the V-Model if suitable for the particular task.

8. Glossary

BO	Business Objective
CDV	Central Development Variant
CM	Configuration Management
CM-x	Configuration Management Activity No. X
COTS	Commercial Of the Shelf
CPU	Central Processing Unit
DOT	Developer Oriented Tool
EC	European Community
HMI	Human Machine Interface
InCoMM	Experimental Introduction of a Configuration Management Model for an Open Control System
IPC	Industrial Personal Computer
ISA	Industry Standard Architecture
ISG	Industrielle Steuerungstechnik GmbH
IST	Information Society Technologies
NC	Numerical Control
OS	Operating System
PD	Person Day
PIE	Process Improvement Experiment
PLC	Programmable Logic Controller
PM	Project Management
POT	Process Oriented Tool
PPR	Periodic Progress Report
PY	Person Year

QA	Quality Assurance
SD	System Development
TO	Technical Objective
V-Model	Lifecycle Process Model (developed to regulate the software development process within the Federal Administration) /1/
VOT	Version Oriented Tool
WP-x	Work Package No. X

9. References

- /1/ N.N.: Entwicklungsstandard für IT-Systeme des Bundes - Vorgehensmodell, Teil 1: Regelungsteil, Allgemeiner Umdruck Nr. 250/1. R. Oldenbourg Verlag, München, Wien 1998.
- /2/ Tichy, W.F.: Configuration Management. John Wiley & Sons, Chichester, New York, Brisbane, Toronto, Singapore 1994.
- /3/ Beeh R., Bürger T.: Quality Assurance for NC-Software with the support of Configuration Management, in: *Proceedings of the EuroSPI'99 Conference*, pp. 9.12-9.23, Pori School of Technology and Economics, Pori, Finland, October 1999.
- /4/ Frühauf, K.: Unterlagen zum TAE-Lehrgang Software Konfigurationsmanagement. Technische Akademie Esslingen, Ostfildern 1998.

Annexes

Figures

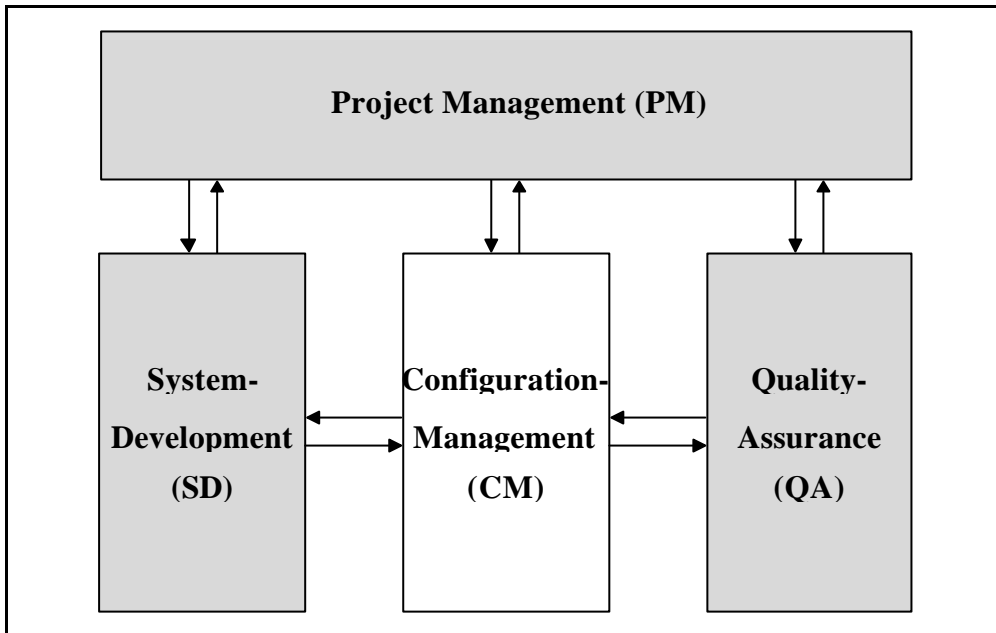


Fig. 1: Sub-models of the V-Model /3/

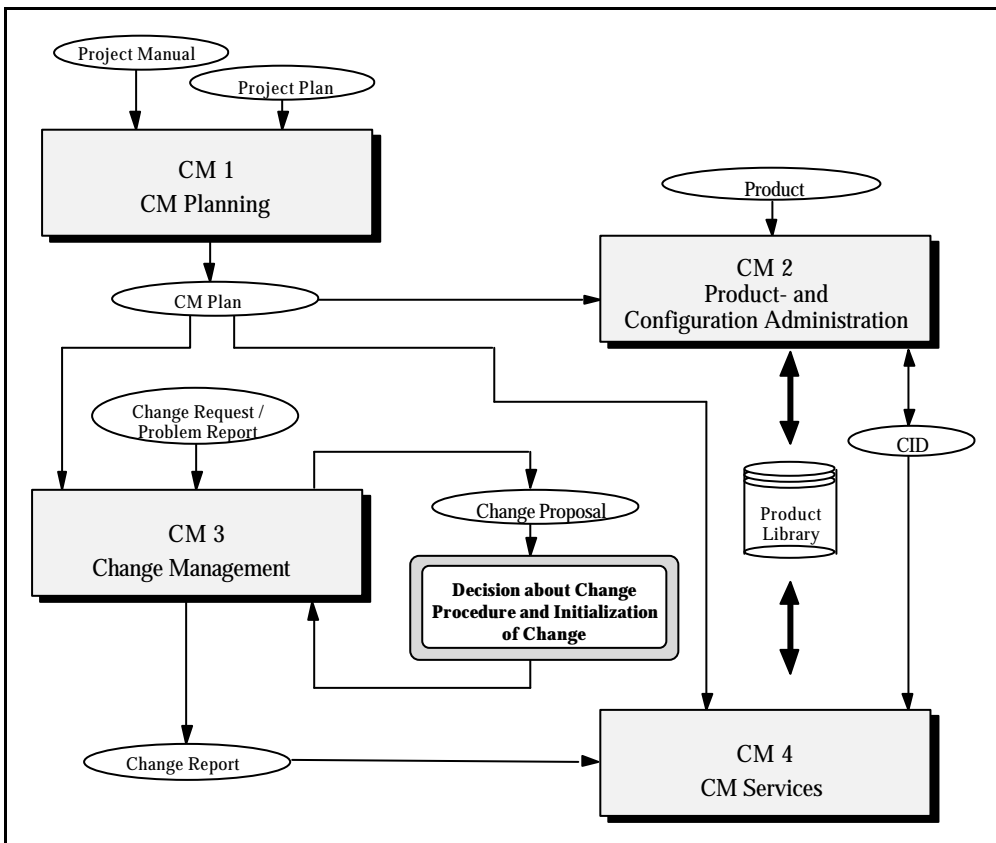


Fig. 2: Activities within the Configuration Management /1/

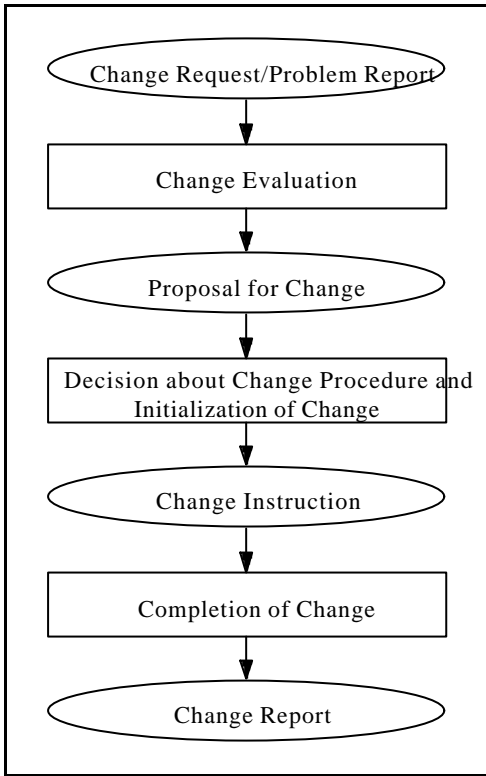


Fig. 3: Sequences within the Change Management

Original Project Plan:

	07/98	08/98	09/98	10/98	11/98	12/98	01/99	02/99	03/99	04/99	05/99	06/99	07/99	08/99	09/99	10/99	11/99	12/99	01/00	02/00	
WP-0																					
WP-1																					
WP-2																					
WP-3																					
WP-4																					
WP-5																					
WP-6																					
WP-7																					
	MS-1 →				MS-2 →				MS-3 →				End →								

Final Project Plan:

	07/98	08/98	09/98	10/98	11/98	12/98	01/99	02/99	03/99	04/99	05/99	06/99	07/99	08/99	09/99	10/99	11/99	12/99	01/00	02/00	
WP-0																					
WP-1																					
WP-2																					
WP-3																					
WP-4																					
WP-5																					
WP-6																					
WP-7																					
	MS-1 →				MS-2 →				MS-3 →				End →								
					EXT-D1 →				EXT-D3 →												



Key phases of the InCoMM -PIE (see also chapter 4.2)



Long term WPs of the InCoMM -PIE (see also chapter 4.2)

WP-X Workpackage No. X (see also chapter 4.2)

MS-X Milestone No. X

EXT-D1 1st presentation at international event ('6th European Conference on SW-Quality', Vienna, April 12th to 16th, 1999)

EXT-D3 2nd presentation at international event ('EuroSPI '99 Conference', Pori, October 25th to 27th, 1999)

Fig. 4: Original and final project plan

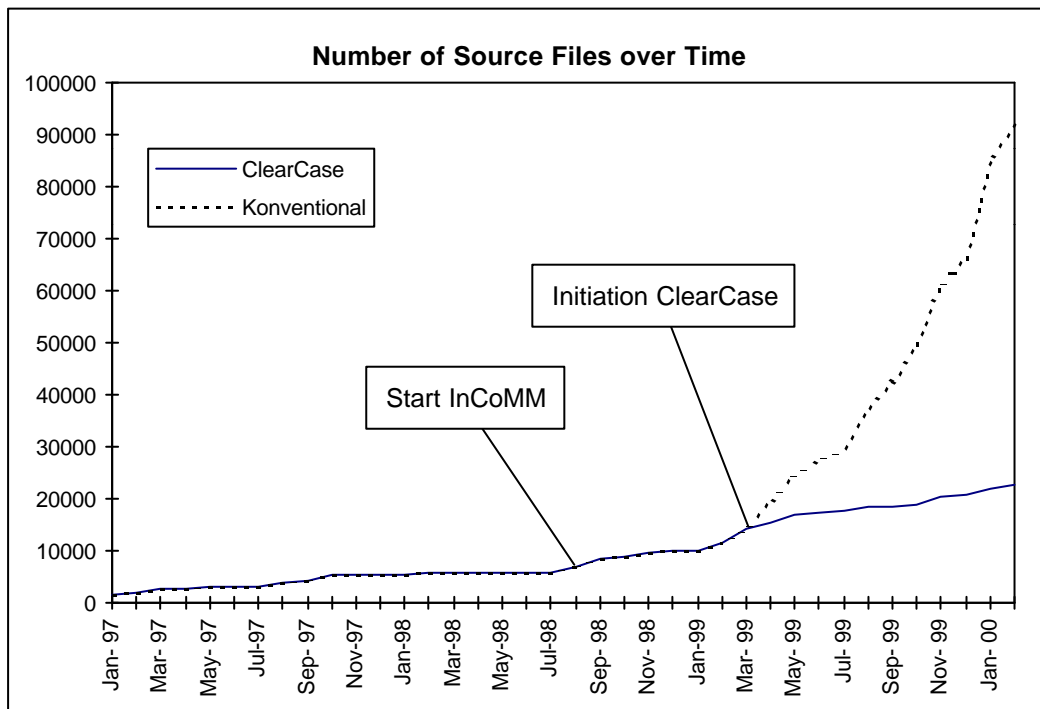
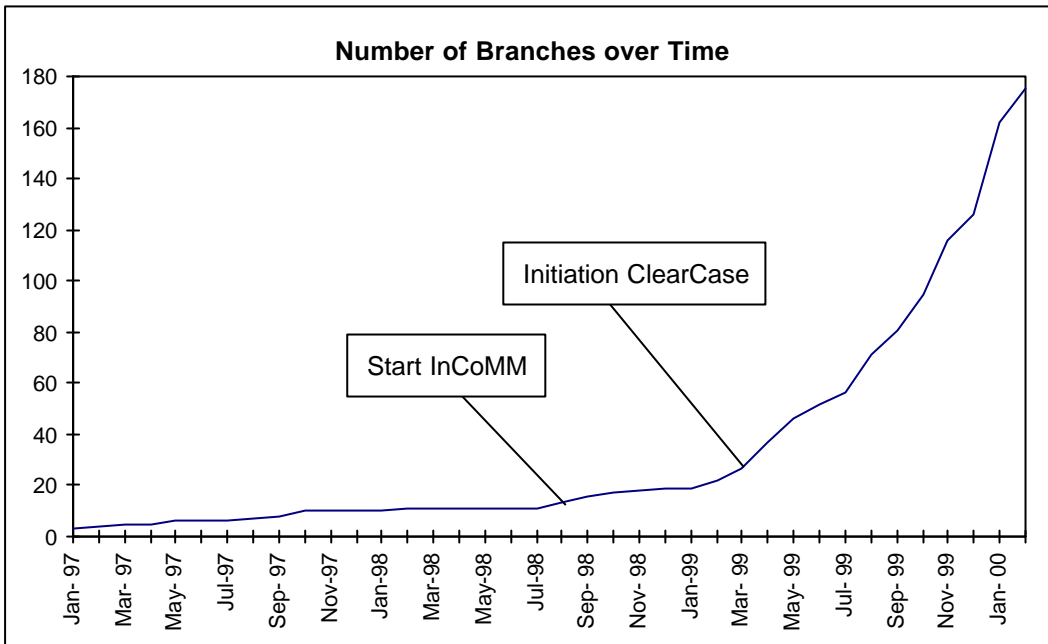


Fig. 5: Increase of the number of reused software elements

Tables

	Objective:	Attainment:	Measurement:
TO	Increase of the number of reused software elements	Change of branching methodology	Evaluation of the reduction of software elements
TO/BO	Improvement of quality	Introduction of <ul style="list-style-type: none"> – CM planning – a clear identification system 	Ongoing evaluation of error data base
BO	Improvement of co-operation with customers and partners in <ul style="list-style-type: none"> – on-time deliveries and – faster responses on support requests 	Use of <ul style="list-style-type: none"> – a clear identification system – an appropriate CM tool 	Evaluation of questionnaires Time measurements

Table 1: Objectives of the InCoMM PIE

Feature	Description	Used by ISG
Build Support	Generating executable files with knowledge about required source files, include files and precompiler settings.	Not yet; planned
Configuration Support	Definition of product versions and product variants (consisting out of several software elements).	Definition of views (with the aid of attributes and labels) including source modules, documentation and test patterns
Customisation Support	Adaptation of the tool to user specific requirements	Definition of different triggers; creating branches at check out event; refusing check in without comment or with wrong file naming
Library/Repository Support	Support by database and tools	In use
Process Support	Support of processes (e.g. for development, bug removal, test).	Not yet, planned
Release Management Support	Definition of product versions; definition of processes for product delivery.	Definition of views (with the aid of attributes and labels) including source modules, documentation and test patterns. Processes are not yet defined.
Reporting/Query	History; file compare; generating protocols of ClearCase tools (e.g. merge)	In use
Security/Protection Support	Administration of access rights	In use
Team Support	Support of parallel development even in developer teams	Definition of customer specific and developer specific views; reserved and unreserved checkouts; identification of hijacked files; use of triggers; support of merging files
Tool Integration	Integration in development environment	Integration in Microsoft developer studio, Windows explorer and ClearCase explorer

Table 2: Features of the CM tool ClearCase

Technical Item:	Attainment:	Actual results:
Increase of the number of reused software elements	Change of branching methodology and use of ClearCase	The increase of the number of software elements could be limited to 160 % since April 1999 in spite of an increase of the number of branches of 650 %.
Improvement of CM planning	Co-ordination and planning development and integration steps with the aid of a magnetic drawing board and ClearCase	A significant reduction of time effort for integrating parallel existing development branches to 10 % of the initially needed time could be reached.
Reduction of the number of delivered bugs to customers	Optimisation of CM planning and use of ClearCase features (e.g. for bug tracking)	A reduction of percentage of bugs detected by customers from 25,73 % (Jul 98) to 17,93 % (May 99) and finally 24,25 % (Jan 2000) could be reached.
Improvement of test execution	Definition of a clear identification system (version numbering and naming)	A significant reduction of time effort for testing could not be detected. The reproducibility of test procedures could be improved.
Improvement/introduction of integration planning	Co-ordination and execution of different projects parallely	Reduction of software elements redundancy and therefore reduction of the number of reused software elements (see above)
CM planning as information source for PM	CM allows PM to trace back project statuses. PM starts to plan CM effort	CM has been reduced to about 10 % of the overall development effort

Table 3: Technical results from the experiment

Business Item:	Attainment:	Actual results:
Improvement of customer support	Use of a great number of parallel existing branches of the NC construction kit for different customers and development purposes. Possible due to the use of an appropriate CM tool.	The number of parallel existing branches increased from 27 to 175 since April 1999. A reduction of the time necessary for responses on support requests could not be detected, as personal (no ISG standardised) CM organisation structures of the respective project managers were already existing. The time necessary for delivery a new version could be reduced to 25 % of the originally needed time.
Quality improvement	Quality improvement of software development process leads to a quality improvement of the product	A reduction of percentage of bugs detected by customers from 25,73 % (Jul 98) to 17,93 % (Jan 99) and finally 24,25 % (Jan 2000) could be reached.
CM as part of the development process	Integration of CM in the software development process as an independent field of activity with its own responsibility and effort	CM has been reduced to about 10 % of the overall development effort

Table 4: Business results from the experiment

CM-Tool Overview

Name of Tool	Company Name and Contact Address in Germany	Kind of Contact	Tool Category
<i>CCC/Harvest</i>	PLATINUM Technology Solutions GmbH Adenauerring 9, D-81737 München	in-house	POT
<i>ClearCase</i>	RATIONAL Software Corporation Keltenring 15, D-82041 Oberhaching	in-house	DOT
<i>CONTINUUS</i>	Software GmbH Arnulfstr. 27, D-80335 München	fair	POT
<i>in-Step</i>	microTOOL GmbH Voltastr. 5, D13355 Berlin	fair	POT
<i>MKS Source Integrity</i>	Mortice Kern Systems GmbH Martinstr. 42, D-73728 Esslingen	fair	DOT
<i>PVCS</i>	INTERSOLV GmbH PO box 1313, D-85729 Ismaning	publicity event	VOT
<i>PVCS Process Manager</i>	INTERSOLV GmbH PO box 1313, D-85729 Ismaning	publicity event	POT