



Chapter 11

Case Study



Review

- Object-oriented design is based on data abstraction and every basic system component is a module supporting data abstraction.
- A class defines a possible set of objects.
- An object-oriented life-cycle approach applies object-oriented principles to every phase of the development life cycle.
- A common process framework (CPF) characterizes an organization's approach to software development and maintenance.
- The component process assembly process model is commonly used as an OO process model.
- Object-oriented metrics are different due to factors such as localization, encapsulation, information hiding, inheritance, and object abstraction.
- Metrics used for object-oriented projects are:
 - ◆ Number of scenario scripts
 - ◆ Number of key classes
 - ◆ Number of support classes
 - ◆ Average number of support classes
 - ◆ Number of subsystems



Objective

- Study the implementation of configuration management in small enterprises

For Aptech Centre Use Only



Business Motivation and Objectives

- The Experiment
- Expected Impact and Experience

For Aptech Centre Use Only



The Experiment

- ICONMAN aims to implement lightweight configuration management procedures for very small enterprises
- The procedures will be based on the current software engineering practice of the partners
- Use of configuration management will address documentation and error reporting
- The project will define and test a set of software metrics



Expected Impact and Experience

- Three companies will implement procedures for revision and configuration management
- There will be procedures to handle all communication with the customers that relates to product development

Business Sector	Software Industry
Application Area	Oil-reservoir simulation, statistical analysis, actuarial software
Keywords	Configuration Management, Software metrics
Technologies/Methodologies/Tools	Configuration Management tools



Lessons Learnt

- Configuration management is a complex activity.
- Implementing configuration management is an iterative process.
- Configuration management should be tested in a controlled, but real-life environment in a small scale organization.
- Difficult to identify quantitative data to measure process performance.
- Existence of an operative configuration management system has shown to make a positive impact on customer relations.
- Existence of an operative change request database has proven to be valuable in planning product releases.
- The existence of an operative configuration item library has proven to simplify the process of reconstructing earlier releases.



Background Information

- The sub-contractors proposed to base the configuration management procedures on the SPICE Support Process Category (SUP).
- The project partners agreed and used a process that is called *SUP-2 – Perform Configuration Management*.



Using SUP-2 – The Purpose

- Establish and maintain the integrity of all the products of a software project throughout the project's software life cycle

For Aptech Centre Use Only



The SUP-2 Process Tasks

- Establish configuration management library system
- Identify configuration items
- Maintain configuration descriptions
- Manage change requests
- Control changes
- Build product releases
- Maintain configuration item history
- Report configuration status



The ICONMAN Project Objectives

- Formalize and document internal procedures for configuration management in each company
- Choose and implement configuration management software
- Produce an internal "Users Guide" for configuration management describing both the routines and the supporting tools
- Develop metrics to measure the impact of configuration management
- Develop a network consisting of the participating software companies and the subcontracted researchers
- Utilize the network for exchange of experiences



Involved Companies in the experiment

- Event AS
- Technical Software Consultants (TSC)
- Aktuar Systemer



Scenario before the experiment

- Event AS

- ◆ All software development is based on one fundamental software library.
- ◆ No formal procedures for handling error reports and requests for changes from the users
- ◆ Technical platform consists of Windows 3.11-workstations in a Novell Netware 4.x - based LAN.

- TSC AS

- ◆ No formal or documented routines for change management of their main product, FrontSim were in use.
- ◆ Procedures for handling error reports were only partially implemented.



Scenario before the experiment Contd...

- ◆ No formal procedures for handling customer feedback
- ◆ Technical environment consists mainly of Unix workstations.
- Aktuar Systemer AS
 - ◆ The company had implemented a very rudimentary version control system.
 - ◆ Simple forms for error reports and change orders were used.
 - ◆ Co-ordination and assignment of priority to various change orders was weak.
 - ◆ No formal test procedures were implemented except for a limited integration test were used.
 - ◆ The technical platform consists of Windows95-workstations in a Novell Netware 4.x - based LAN.



Work Plan

- First Phase
- Second Phase
- Third Phase

For Aptech Centre Use Only



First Phase

- Lasted 6 months
- Configuration management procedures were developed
 - ◆ Management of change requests
 - ◆ Metrics to measure the outcome of the experiment
- Appropriate tools were selected and introduced together with the procedures.



Second Phase

- Lasted 7 months
- Consisted of
 - ◆ Training of the developers
 - ◆ Refining the configuration management procedures based on feedback from the developers
 - ◆ Starting the process of collecting metrics data



Third Phase

- Lasted 5 months
- Collection of metrics data continued
- Experiment was evaluated



List of Deliverables

Deliverable Reference	Availability: (I)nternal (R)estricted (P)ublic	Description of the deliverable (Title)
D1	I	Report on definition of CM
D2	I	Report on Implementation of selected tools
D3	R	Periodic progress report (first 6 months)
D4	R	Report on training on the job
D5	P	Mid-term report
D6	R	Report on first estimate of the defined metrics
D7	I	Report on refined CM procedures
D8	R	Periodic progress reports (6 to 12 months)
D9	P	Final Periodic Progress Report
D10	R	Consolidated cost statement
D11	P	Final Report



Expected Outcomes

- Improved procedures for handling feedback from customers
- Improved documentation procedures
- Improved testing procedures
- Configuration and change management procedures



Work Performed

- Organization
- Technical environment
- Training
- Role of the consultants
- Phases of the experiment
- Internal dissemination



Results and Analysis

- Objectives were met
- The outcome of the experiment corresponded well with the expected ones



Objectives after experiment

- Technical
- Business
- Organization
- Culture
- Skills



Conclusion

- The implementation of configuration management in the three organizations has been successfully completed.



Future Actions

- Use of all routines and tools has to be consolidated.
- Further metrics to extend the control over the development process have to be implemented.
- Documentation of the development process has to be further improved.



Summary

- The ICONMAN project's main objective was to implement configuration management in three small software development companies and to assess the effect of this measure from a business and a technical point of view.
- Configuration management is a complex process and highly integrated with change management, both with respect to error correction and handling customer requests.
- Implementing configuration management is an iterative process, and requires continuous refinement.
- Configuration management in small companies dealing mainly with the development of one product has a large impact on the way the whole business is organized.
- Formalized routines for configuration management will help small companies in planning their production process.