Enabling reuse
with a configuration language

Jacqueline Floch  SINTEF
Bjørn Gulla  Telenor

Fourth International Conference on Software Reuse
Garex Products

- Voice Switching Systems for the professional communication control market (Air Traffic Control, Sea Traffic, Rescue Coordination, Police, ...)

- Highly customized systems

- Design and supply both HW and SW

- Few employees; numerous installations world-wide!
Reuse factors

- Generic software architecture
- Standardized component interface and protocols
- Highly configurable components
- Specific and well-known domain
- High stability of technical staff
Enabling reuse with a configuration language - 4th ICSR

Software architecture enforced by the runtime support system (TST)

TST provides support for FSM-based systems:

- FSM interpretation
- communication
- composition
- instantiation
- configuration

Enabling reuse with a configuration language - 4th ICSR
Problems

• Lack of visibility of the overall system structure
• Lack of representation of variability
• Complexity of system configuration on TST
• Lack of tool support for system modelling
• Complexity of system composition and building
• Partial configuration binding
• Evolution
The PROTEUS project

- ESPRIT project 1992 - 1995, funded by the European Commission

- Objective: to provide methodological and tool support for system evolution

- Main result: Proteus Configuration Language and associated PCL tools

- In Norway two companies Garex and Stentofon participated as “application” providers.
PROTEUS Configuration Language

- PCL is a language for modelling systems and system building
- The PCL model is based on the family notion
PROTEUS Tools

Enabling reuse with a configuration language - 4th ICSR
PROТЕUS approach at Garex

**OOram role model**

**PCL system model**

**Logical view**

**Physical view**

**MSC-based extractor**

**Message sequence charts**

**System building process**

**Configuration file**

**Executable system**

**Software source code**

**Reverse engineering**

**Forward engineering**

Enabling reuse with a configuration language - 4th ICSR
A comprehensive PCL model of a Garex application (500 kloc + 10 kl configuration) was developed:

• PCL model provides overall system visibility

• System and component variability can be described

• Makefiles for building particular system instances can be generated automatically

• TST configuration files can be generated
Evaluation results

+ Tasks can be done at a higher abstraction level
+ All system knowledge including variability is represented in one place
+ Concise system building specification is incorporated in system model
- The models for TST configuration generation are complex: limited support for instance description
- The implementation of interactive and partial binding is not satisfactory
- The PCL toolset is not a commercial product
Conclusions

• Garex wish to move from implementation oriented reuse towards design oriented reuse

• PCL is a promising approach for supporting reuse, allowing flexible component composition and configuration

• PCL is currently *not* used for component reuse classification and selection

• We need more experience in order to:
  - learn about maintenance of PCL models
  - develop guidelines for using PCL
Enabling reuse with a configuration language - 4th ICSR

PCL main concepts

Family entity
- attributes
- classification
- interface
- parts
- physical
- relationships

Defined by

attribute type def.
classification def.
relation def.

Selects

Version descriptor
- attributes
- parts

Inherits from

Tool entity
- attributes
- inputs
- outputs
- scripts

Inherits from

Enabling reuse with a configuration language - 4th ICSR
Enabling reuse
with a configuration language

Jacqueline Floch
Bjørn Gulla

SINTEF
Telenor

Fourth International Conference on Software Reuse
Garex Products

• Voice Switching Systems for the professional communication control market (Air Traffic Control, Sea Traffic, Rescue Coordination, Police,...)

• Highly customized systems

• Design and supply both HW and SW

• Few employees; numerous installations world-wide!
Reuse factors

- Generic software architecture
- Standardized component interface and protocols
- Highly configurable components
- Specific and well-known domain
- High stability of technical staff
Software architecture enforced by the runtime support system (TST)

TST provides support for FSM-based systems:

- FSM interpretation
- communication
- composition
- instantiation
- configuration

Enabling reuse with a configuration language - 4th ICSR
Problems

• Lack of visibility of the overall system structure
• Lack of representation of variability
• Complexity of system configuration on TST

• Lack of tool support for system modelling
• Complexity of system composition and building
• Partial configuration binding
• Evolution
The PROTEUS project

- ESPRIT project 1992 - 1995, funded by the European Commission

- Objective: to provide methodological and tool support for system evolution

- Main result: Proteus Configuration Language and associated PCL tools

- In Norway two companies Garex and Stentofon participated as “application” providers.
PROTEUS Configuration Language

- PCL is a language for modelling systems and system building
- The PCL model is based on the family notion
Enabling reuse with a configuration language - 4th ICSR
PROTEUS approach at Garex

OOram role model

PCL system model

Logical view

Physical view

MSC-based extractor

System building process

Message sequence charts

Configuration file

Trace

Software source code

Executable system

PCL Reverse

Enabling reuse with a configuration language - 4th ICSR
Achievements

A comprehensive PCL model of a Garex application (500 kloc + 10 kl configuration) was developed:

- PCL model provides overall system visibility
- System and component variability can be described
- Makefiles for building particular system instances can be generated automatically
- TST configuration files can be generated
Evaluation results

+ Tasks can be done at a higher abstraction level
+ All system knowledge including variability is represented in one place
+ Concise system building specification is incorporated in system model
- The models for TST configuration generation are complex: limited support for instance description
- The implementation of interactive and partial binding is not satisfactory
- The PCL toolset is not a commercial product
Conclusions

• Garex wish to move from implementation oriented reuse towards design oriented reuse

• PCL is a promising approach for supporting reuse, allowing flexible component composition and configuration

• PCL is currently *not* used for component reuse classification and selection

• We need more experience in order to:
  - learn about maintenance of PCL models
  - develop guidelines for using PCL
Enabling reuse with a configuration language - 4th ICSR

PCL main concepts

Family entity
- attributes
- classification
- interface
- parts
- physical
- relationships

Tool entity
- attributes
- inputs
- outputs
- scripts

Version descriptor
- attributes
- parts

Selects

Defined by:
- attribute type def.
- classification def.
- relation def.

Inherits from:
- classified

Enabling reuse with a configuration language - 4th ICSR