

Intrinsic Flexibility and Robustness in Adaptive Systems: A Conceptual Framework

Stephan Kubisch, Ronald Hecht, Ralf Salomon,
Dirk Timmermann
{stephan.kubisch,ralf.salomon}@uni-rostock.de

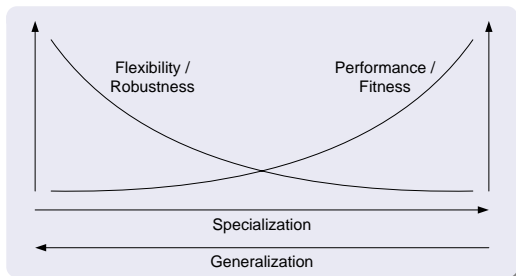
University of Rostock,
Institute of Applied Microelectronics and Computer Engineering



July 25th, SMCals/06, Logan, Utah

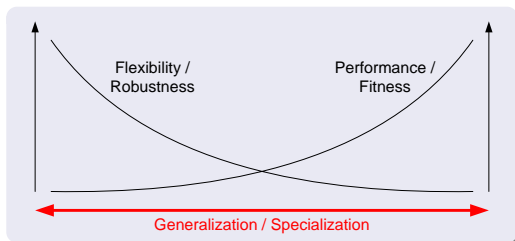
Outline

- 1 **Introduction**
 - Motivation
 - Objectives
- 2 **The Plan**
 - Version Control Systems
 - Reconfiguration in Hardware
 - The Conceptual Framework
- 3 **The Outcome**
 - New System Properties
 - Preliminary Results
 - Target Applications/Challenges
- 4 **Summary**



Flexibility, Robustness, and Performance

- Performance increases with specialization
- Robustness increases with generalization
- Performance and robustness are opposite
- Archetype nature: achieve high fitness & exploit a niche at best ⇒ But why became the dinosaurs extinct?



Goals

- 1 Get along with uncertainty in technical systems
- 2 Performance in dynamic environs \Rightarrow need for adaptability
- 3 Robustness in case of failure or abrupt changes

Conclusion: We need mechanisms to specialize **and** to generalize!

Source of Inspiration: CVS

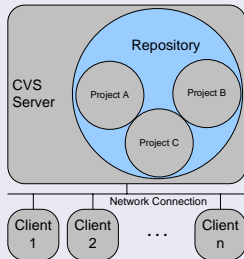
1970 SCCS - diffing function

1985 RCS - multiple but exclusive users

1990 **CVS** - multiple simultaneous users

200x Subversion, monotone...

CVS is well known, approved, and widely used



Source of Inspiration: CVS

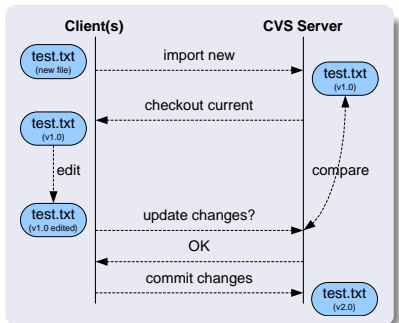
1970 SCCS - diffing function

1985 RCS - multiple but exclusive users

1990 **CVS** - multiple simultaneous users

200x Subversion, monotone...

CVS is well known, approved, and widely used



Why is CVS interesting for our goals?

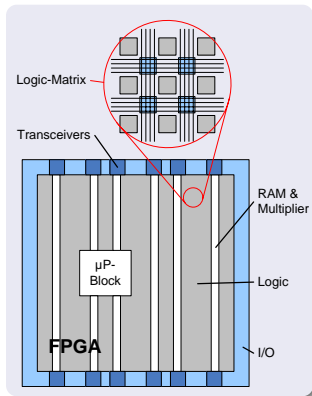
Feature	in CVS	in the Framework
Version Numbering	Distinguish changes and different releases	Temporal order, incremental logging
Branching	Isolate changes for separate lines of development	Adaptation in different directions
Role-back	Go back to main line of development	Memory, checkout of former revisions
Multiple users	Concurrent editing	Redundancy

⇒ Rich set of available features



Adaptation in Hardware: FPGAs

- Rich in logic resources & features
- Wired during Synthesis-Flow
- Partial Dynamic Reconfiguration (PDR)

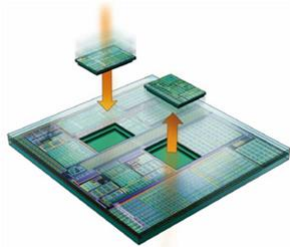


Adaptation in Hardware: FPGAs

- Rich in logic resources & features
- Wired during Synthesis-Flow
- Partial Dynamic Reconfiguration (PDR)

PDR

- Reconfigure distinct parts @ runtime
- Increased flexibility
- In-field updates & bug-fixes
- Applications allowing no interrupts
- Self-x attributes



Managing all this: Operating System Support

- No new OS \Rightarrow modify existing one
- Linux is Open Source
- Modifications to support dynamic reconfiguration in hardware and software
- hardware resources managed like memory \Rightarrow going virtual
- Existing CVS features can be used



Overview

The new conceptual Framework

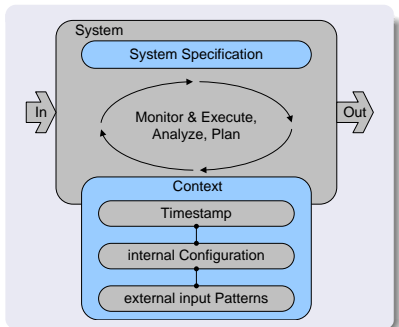
- 1 System context to be tracked & stored in a CVS library
- 2 System's own specification
- 3 Redundant system models allow for adaptation
- 4 System-wide observer
- 5 Coordination by meanings of a modified Linux operating system

The System Context

- Triplet of information
 - Current input patterns to recognize certain conditions
 - Internal configuration parameters
 - Timestamp as index

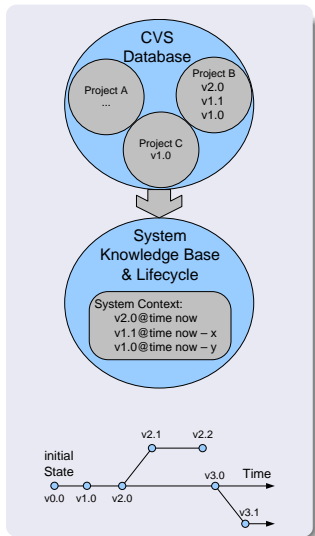
The System Specification

- System's own functional scope, parameters, and constraints
- Policies for specialization and adaptation mechanisms



System Context in a CVS

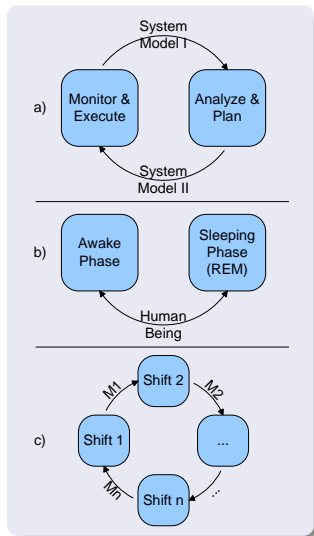
- ... stored on occurrence of changes
- Version no. & timestamp \Rightarrow temporal order
- Represents lifecycle and knowledge base
- Estimate future or cyclic events
- Proactive preparations \Rightarrow proprioception
- System's history affects its future
- Obsolete entries may be deleted



Adaptation requires Redundancy

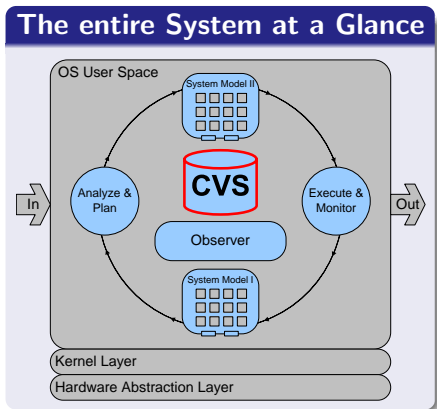
- Two concurrent system models
- Flexible shift work in a loop
- 2 Shifts
 - Analyze & Plan
 - Monitor & Execute
- Human sleep: Awake ↔ REM-phase

Generally, nature shows no redundancy.
But it can be applied in technical systems.



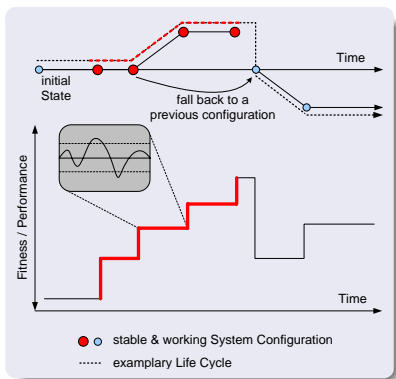
The Conceptual Framework

- Combine the aforementioned parts
- 2 redundant system models in shift work
- Observer estimates/ calculates potential benefits
- Learning algorithm connected to observing entity
- CVS is central system library and knowledge base



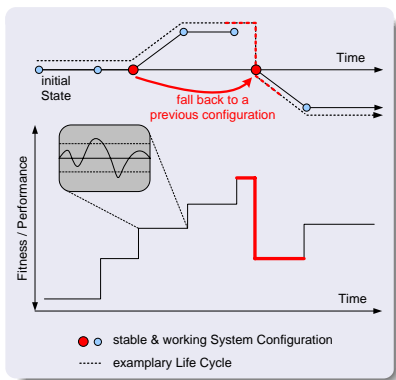
Property 1: Performance Maximization...

- ... by intrinsic flexibility
- React on changes and specialize
- Adaptability by using a learning algorithm
- Fitness increase is the benefit
- But the more specialized the less flexible



Property 1: Survivability/Availability...

- ...by intrinsic robustness
- withstand and endure changes
- Fall-back capability of CVS-Lib allows generalization
- Lowering the risks of uncertainty
- Maybe fitness decreases



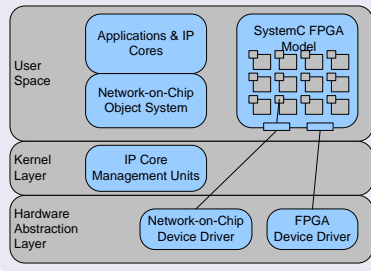
Ingredients & Modifications

- Network-on-Chip object system
- SystemC model of FPGA
- Kernel-Mechanisms for IP-core management
- FPGA & NoC device drivers

Results

- OS support for runtime reconfigurable systems
- Swapping of IP-cores \Rightarrow hardware scheduled like virtual memory
- Multiple system models in OS user space

The Operating System Architecture



” Grand Challenge”

- Reach missions objectives without human intervention
- Self governance



Target Applications/Challenges

"Grand Challenge"

- Reach missions objectives without human intervention
- Self governance

Outer Space Missions

- Reach missions objectives in unknown environs
- Improve/not downgrade within functional scope
- Self management



Target Applications/Challenges

"Grand Challenge"

- Reach missions objectives without human intervention
- Self governance

Outer Space Missions

- Reach missions objectives in unknown environs
- Improve/not downgrade within functional scope
- Self management

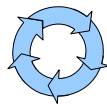
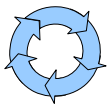
Telecommunication

- Dynamic environment
- Adapt to, e.g., bandwidth demands

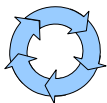


Summary

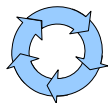
- Focus on flexibility and robustness in dynamic environs



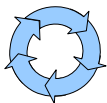
Summary



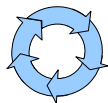
- Focus on flexibility and robustness in dynamic environs
- Conceptual framework for adaptive systems



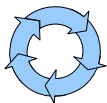
Summary



- Focus on flexibility and robustness in dynamic environs
- Conceptual framework for adaptive systems
- CVS-library is the central entity in the framework

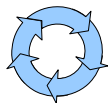


Summary



- Focus on flexibility and robustness in dynamic environs
- Conceptual framework for adaptive systems
- CVS-library is the central entity in the framework

⇒ Combining techniques for specialization **and** generalization



Thank you! Any questions?