Software engineering self-adaptive web-based information systems - Current research topics

vangel.ajanovski@finki.ukim.mk

Faculty of Computer Sciences and Engineering (FCSE)
Ss Cyril and Methodius University, Skopje, Macedonia

Introduction and concepts

Origins of "self-adaptive"

- Earliest reference in IEEE databases
- Proceedings of the self adaptive flight control systems symposium, 1959

Definition on Self-Adaptive (Software) Systems (abrev. SAS):

- "*Self adaptive software is software that monitors its own operation, detects faults and opportunities, and repairs or improves itself in response to faults and changes. It effects the improvement by modifying or re-synthesizing its programs and subsystems, using a feedback control-system like behavior.* "
  - From: "*Results of the Second International Workshop on Self-adaptive Software", IWSAS 2001, Lake Balaton, Hungary*

This presentation gives an overview of the results of a series of workshops on SAS with regard to Software Engineering

- intended as a special (and still general) case - on web-based information systems as part of two courses

SEAMS (International Symposium on Software Engineering for Adaptive and Self-Managing Systems)

"*The ICSE workshop series on Software Engineering for Adaptive and Self-Managing Systems (SEAMS) attempts to consolidate interest within the software engineering community on self-adaptive, self-managing, self-healing, self-optimizing, self-configuring, and self-organising systems. The aim of the SEAMS symposium/workshop series is to discuss progress and challenges in this important area of software engineering.*"

Workshop is hosted as part of ICSE (International Conference on Software Engineering)

ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems

- 2006 - Shanghai, China
- 2007 - Minneapolis, USA
- 2008 - Leipzig, Germany
- 2009 - Vancouver, Canada
- 2010 - Capetown, South Africa

ICSE Internation Symposium on Software Engineering for Adaptive and Self-Managing Systems

- 2011 - Hawaii, USA
- 2012 - Zürich, Switzerland
- 2013 - San Francisco, USA

Dagstuhl Seminars

Organised at Schloß Dagstuhl, Wadern, Germany
Related Events and Results:

- Software Engineering for Self-Adaptive Systems: Research roadmap - 2008
- Software Engineering for Self-Adaptive Systems: 2nd Research roadmap - 2010
- *Software Engineering for Self-Adaptive Systems: Assurances - 2013*

Research Roadmap – Classification of Modeling Dimensions

Goals - the objectives the system under consideration should achieve

- Evolution, Flexibility, Duration, Multiplicity, Dependency

Changes - the cause of adaptation

- Source, Type, Frequency, Anticipation

Mechanisms - the system reaction towards change

- Type, Autonomy, Organization, Scope, Duration, Timeliness, Triggering

Effects - the impact of adaptation upon the system

- Criticality, Predictability, Overhead, Resilience

Research Roadmap - Adapt the Requirements

Introduction of uncertainty. Whereas in traditional requirements:

- Standard requirement: The system will do this ...
- Quasi-adaptive: For 1<*parameter*<10 the system will do this ... for *parameter*>=10 the system will do that ...

A new requirements language is needed with constructions like:

- “The system might do this ... ”
- “But it may do this ... as long as it does this ... ”
- “The system ought to do this ... but if it cannot, it shall eventually do this ...”

Deal with uncertainty in several aspects:

- Mapping uncertainty to architecture - run-time adaptation, reconfigurability
- Managing uncertainty - the boundary between invariants and flexibility, managing risk
- Requirements reflection - observe and monitor the requirements themselves as objects in run-time
- Online goal refinement - change of goals not just in design-time, but also in run-time
- Traceability from requirements to implementation - changing in run-time

Research Roadmap - Engineering SAS

Introduce the notion of feedback

- Many natural processes involve feedback mechanisms for self-control
- Watt's steam engine - first self-regulating mechanical system

Control Loop Model phases:

- Collect, Analyze, Decide, Act

Increase the focus on the control loop and introduce the loop mechanism as a first-class system component
Control loops and Software Engineering

- Modeling
- Architecture
- Design
- Middleware
- Verification and Validation
- Reengineering
- Human Computer Interaction

Research Roadmap - System Assurance

Proof that the set of stated functional and non-functional properties are satisfied

- In a system where many properties and functionalities can change

Important aspects to investigate:

- Dynamic Identification of Changing Requirements
- Adaptation-Specific Model-Driven Environments
- Agile Run-Time Assurance
  - Proof-carrying code (PCC)
- Liability and Social Aspects

2nd Research Roadmap - Design space for adaptive solutions

Important design space dimension clusters:

- Representation in runtime - of the problem and system - internal representation of the system itself
- Observation - what to monitor, when and how
- Control - decide on the change for the adaptation, control the feedback, organization of multiple control loops
- Identification - how to identify the instantiation of the system at the very moment, instantiations accross goals
- Adaptation Mechanisms - explicit/implicit, internal/external, how to support adaptation

2nd Research Roadmap – Software Engineering Processes

Need for new software processes for self-adaptive systems.

Traditionally, the process is divided on the time axis:

- development-time, deployment-time, run-time

Some cases wll have a shift:

- development and deployment will happen in run-time
- human role will move from operational to strategical

The important challenges

- Full understanding of the nature of a system that is changing - full life-cycle
- Implement some framework to establish new processes:
  - Library of process elements for self-adaptive software systems built as a framework that the software engineer uses to instantiate the software process by using:
    - reusable process elements - activities, roles, artifacts
    - best-practices, conrete processes
    - process analysis, tuning, specific components

- Process elements defined based on SPEM framework (Software & Systems Process Engineering Metamodel Specification)
- Automation is a key question - process should be based on models using run-time able modeling tools

2nd Research Roadmap - From Centralized to Decentralized Control

This area discusses organization issues for multiple control mechanisms - dependency and inter-control
All adaptation control is based on several sequencial steps

- monitor, analyze, plan, and execute (MAPE)
- the series loops and forms a feedback mechanism

Several patterns are possible:

- hierarhical control (higher level - longer timescale)
- master/slave (A and P at master, M and E at slave)
- regional planner (M, A and E are decentrilized and are in control of local hosts, P happens in a regional wide central)
- fully decentralized (each host - whole MAPE loop, communication peer-to-peer M-M, A-A, P-P, E-E)
- information sharing (similar to above but only peer M-M communication happens)

Investigation on pattern applicability, completeness and final QoS analysis

2nd Research Roadmap - Practical Run-time Verification and Validation

Fundamental questions, perform the verification and validation without hindering the system:

- what to validate?
- where to measure the aspects to validate?
- when to validate these aspects?

Viability zone (set of states where the system is satisfactory) is highly dependent on context which changes
Most important tasks to achieve:

- Tracing the evolution of requirements
- Control complexity expected at run-time
- Dynamic context monitor

Links

http://self-adaptive.org/
- contains sections for all the mentioned conferences, symposiums and seminars, with all the publications

References

Software Engineering for Self-Adaptive Systems-- A Research Roadmap

- B.H.C. Cheng et al. (Eds.): Self-Adaptive Systems, LNCS 5525, pp. 1–26, 2009. Springer-Verlag Berlin Heidelberg 2009

Software Engineering for Self-Adaptive Systems-- A Second Research Roadmap

- Rogério de Lemos, Holger Giese, Hausi A. M¨ller, Mary Shaw, et al, Dagstuhl Seminar 10431 on Software Engineering for Self-Adaptive Systems in October 2010.

Results of the Second International Workshop on Self-adaptive Software

- R. Laddaga, P. Robertson, H. Shrobe (Eds.): IWSAS 2001, LNCS 2614, pp. 281–290, 2003. Springer-Verlag Berlin Heidelberg 2003