

## Welcome to SENG 480B / CSC 485B / CSC 586B Self-Adaptive and Self-Managing Systems

Dr. Hausi A. Müller  
Professor  
Department of Computer Science  
University of Victoria

<http://courses.seng.uvic.ca/courses/2013/summer/seng/480b>  
<http://courses.seng.uvic.ca/courses/2013/summer/csc/485b>  
<http://courses.seng.uvic.ca/courses/2013/summer/csc/586b>

## Announcements

- Thu, May 30
  - Assignment 1 due
  - Presentations of Part III after break
- Fri, May 31
  - Assignment 2 handed out
- Mon, June 3 – Tue, June 11 (inclusive)
  - Congress of the Humanities and Social Sciences
  - No classes
- Fri, June 28
  - Midterm in class

<http://uviccongress2013.ca/>



## Reading Assignments Autonomic Computing

- Kephart, J.O., Chess, D.M.: The Vision of Autonomic Computing. IEEE Computer 36(1):41-50 (2003)
- IBM Corp.: An Architectural Blueprint for Autonomic Computing, Fourth Edition (2006)  
<http://people.cs.kuleuven.be/~danny.wevns/csd/IBM06.pdf>
- Kluth, A.: Information Technology: Make It Simple. The Economist (2004)  
[http://www.economist.com/surveys/displaystory.cfm?story\\_id=E1\\_P\\_PDSPPG&CFID=17609242&CFTOKEN=84287974](http://www.economist.com/surveys/displaystory.cfm?story_id=E1_P_PDSPPG&CFID=17609242&CFTOKEN=84287974)

3

## Specific Challenges in ULS System Monitoring and Assessment

- The effectiveness of ULS system design, operation, evolution, orchestration, and control has to be evaluated.
- There must be an ability to monitor and assess ULS system state, behavior, and overall health and well being.
- Challenges include
  - Defining indicators
  - Understanding why indicators change
  - Prioritizing the indicators
  - Handling change and imperfect information
  - Gauging the human elements

Design and evolution  
Orchestration and control  
→ Monitoring and assessment

4

## Unprecedented Levels of Monitoring

- To be able to observe and possibly orchestrate the continuous evolution of software systems in a complex and changing environment, we need to push the monitoring of evolving systems to unprecedented levels.

5

## Runtime Check Monitors

- Monitor assertions and invariants
- Monitor frequency of raised exceptions
- Continually measure test coverage
- Data structure load balancing
- Buffer overflows, intrusion
- Memory leaks
- Checking liveness properties

6

## Monitor, Assess, and Manage System Properties

1. Govern and enforce rules and regulations
2. Monitor compliance
3. Assess whether services are used properly
4. Monitor and build user trust incrementally
5. Manage tradeoffs
6. Recognizing normal and exceptional behaviour
7. Assess and maintain quality of service (QoS)
8. Monitor service level agreements (SLAs)
9. Assess and monitor non-functional requirements



Define indicators  
Understand why indicators change

## Midterm Question

- Which type of system is more difficult to maintain and evolve? And why?
  - Traditional information systems
  - Self-adaptive systems



8

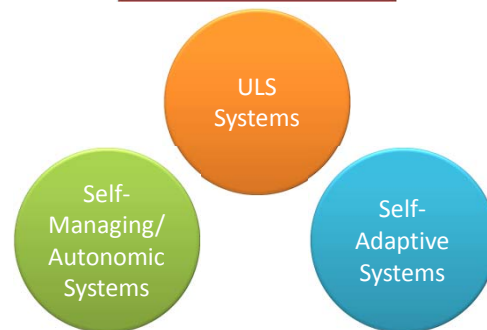
## Midterm Question

- Which type of system is more difficult to maintain and evolve? And why?
  - Traditional information systems
  - Self-adaptive systems



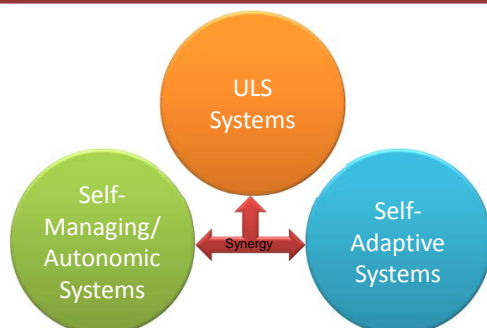
9

## Related Systems



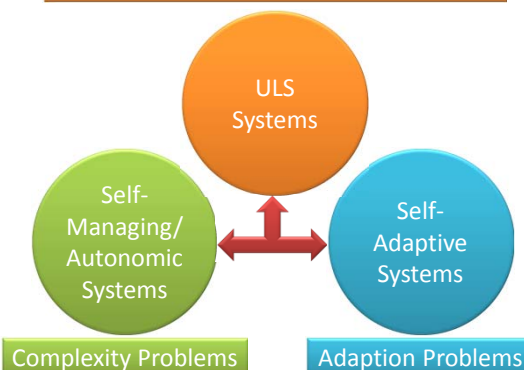
10

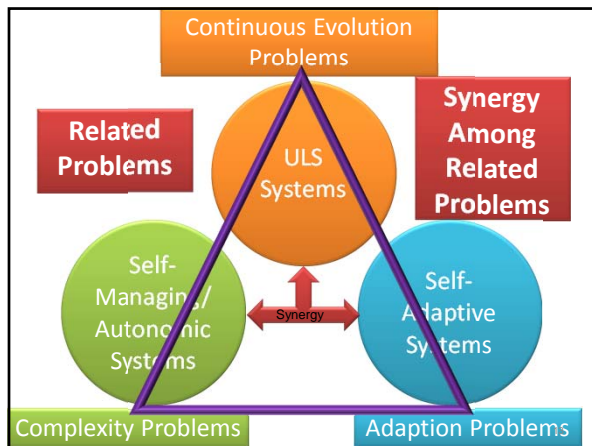
## Synergy Among Related Systems



11

## Continuous Evolution Problems





### The Continuous Evolution Problem

Devices, environments, infrastructure, web, services, business goals, user expectations, ...

all evolve over time

all evolve over time

14