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

1

Quiz 5

- Are you sitting next to the same person you did on Fri?
- Did you look up any term or resource related to this course since Fri?

- This course involves a lot of reading! How much reading have you done so far?

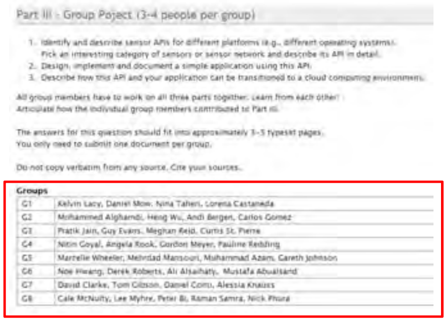
2

Reading Assignments

- ULS Book Section 1-3 on-line at
  - http://www.sei.cmu.edu/uls/the_report.html
- Murray (Ed.): Control in an Information Rich World Report of the Panel on Future Directions in Control, Dynamics, and Systems, SIAM (2003)
  - Chapters 1 & 2
  - http://www.cds.caltech.edu/~murray/cdspanel/report/cdspanel-15aug02.pdf

3

Assignment 1 — Part III



Part III - Group Poject (3-4 people per group)

1. Identify and describe sensor APIs for different platforms (e.g., different operating systems). Pick an interesting category of sensors or sensor network and describe its API in detail.
2. Design, implement and document a simple application using this API.
3. Describe how this API and your application can be transitioned to a cloud computing environment.

All group members have to work on all three parts together. Learn from each other! Articulate how the individual group members contributed to Part III.

The answers for this question should fit into approximately 3-5 typeset pages. You only need to submit one document per group.

Do not copy verbatim from any source. Cite your sources.

Groups

G1	Kelvin Lacy, Daniel Mow, Nina Taheri, Lorena Castaneda
G2	Mohammed Alghamdi, Heng Wu, Andi Bergen, Carlos Gomez
G3	Pratik Jain, Guy Evans, Meghan Reid, Curtis St. Pierre
G4	Nitin Goyal, Angela Rook, Gordon Meyer, Pauline Redding
G5	Marrelie Wheeler, Mehrdad Mansouri, Muhammad Azam, Gareth Johnson
G6	Noe Hwang, Derek Roberts, Ali Alsaihaty, Mustafa Abualsaud
G7	David Clarke, Tom Gibson, Daniel Cornu, Alessia Knauss
G8	Cale McNulty, Lee Myhre, Peter Bi, Raman Samra, Nick Phura

4

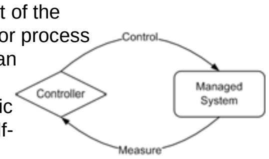
What have you learned so far?

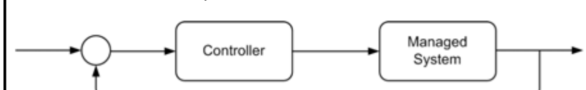
- Self-adaptive systems (SAS)
- Situational awareness (SA)
- Context—life cycle, management
- The three I's
  - Instrumented, Interconnected, Intelligent—smart(er)
- Smart systems revolution
- Something profound is happening
  - *Confluence* of sensors, networks, (mobile) devices, clouds and apps
- Internet of Things (IoT)
- Continuous evolution
- Managing trade-offs
- Feedback—positive, negative, bipolar
- Feedback loops in natural and engineering systems

5

Feedback Systems

- **Merriam-Webster's Online Dictionary**
  the return to the input of a part of the output of a machine, system, or process (as for producing changes in an electronic circuit that improve performance or in an automatic control device that provide self-corrective action)

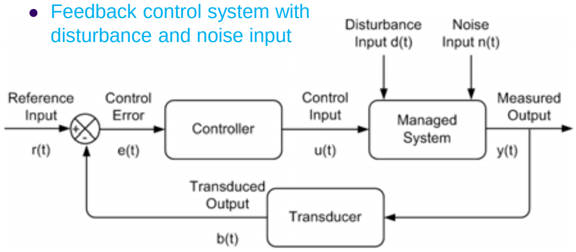




6

Realization of a Dynamic Architecture

- Feedback control system with disturbance and noise input



Hellerstein, Diao, Parekh, Tilbury: Feedback Control of Computing Systems. John Wiley & Sons (2004)

7

Feedback is ubiquitous
in natural and
engineered systems

8

Feedback loops in natural and engineering systems

- Autonomic Nervous System (ANS)
  - Separates normal day-to-day internal processes from exceptional, stressful situation processes
- Homeostasis
  - Property of a system that regulates its internal environment to maintain a stable condition (equilibrium)
  - Carbon-water, ice-albedo, climate, financial markets
- Autonomous vehicles
  - Quadcopters, blimps, drones, robots
- Controllers in engineering
  - Centrifugal governor, cruise control, ABS, guidance and flight control, Mars Curiosity, industrial process control, printing press, NC machines, computer networks
- Feedback equations
  - Fractal generators—ferns and grasses
  - Julia and Mandelbrot sets

9

Control in an Information Rich World

- Control in an Information Rich World Report of the Panel on Future Directions in Control, Dynamics, and Systems
  Edited by Richard Murray
  - Chapters 1 & 2
  - http://www.cds.caltech.edu/~murray/cdspanel/report/cdspanel-15aug02.pdf
- Quadcopters
  - http://www.quadcopter.org
  - http://www.geekosystem.com/quadcopters-james-bond-theme/
  - http://www.openideo.com/open/usaid-humanity-united/ideas/quadcopter-signal-network

10

James Bond Theme Played by Quadcopters

http://www.geekosystem.com/quadcopters-james-bond-theme/

11

Julia and Mandelbrot Set Explorers

- http://aleph0.clarku.edu/~djoyce/cgi-bin/expl.cgi
  - http://math.bu.edu/DYSYS/explorer/
  - http://mbharris.co.uk/fractals/mandelbrot-set-and-julia-set-explorer-in-html5.html
- http://www.javaview.de/vgp/iterate/juliaSet/PaJuliaSet.html
- YouTube: search for *Mandelbrot video* and enjoy

Types of Feedback

- **Negative feedback**
  - Stabilizes operation; regulates within a set and narrow range
  - Classic examples
    - Thermostat control
    - Homeostasis
- **Positive feedback**
  - Increase, accelerate, or enhance output created by a stimulus that has already been activated
  - Classic example
    - Audio feedback—sound from loudspeakers enters a poorly placed microphone and gets amplified, and as a result the sound gets louder and louder
    - Blood platelet accumulation, which, in turn, causes blood clotting in response to a break or tear in the lining of blood vessels
    - Release of oxytocin to intensify the contractions that take place during childbirth
- **Bipolar feedback**
  - Either increase or decrease output
  - Bipolar feedback is present in many natural and human systems
  - Feedback is usually bipolar in natural environments producing synergic and antagonistic responses to the output of system

13

Ultra-Large-Scale (ULS) Systems

- Premise
  - ULS systems will place an unprecedented demand on software acquisition, production, deployment, management, documentation, usage, and evolution
- Needed
  - A new perspective on how to characterize the problem
  - Breakthrough research in concepts, methods, and tools beyond current hot topics such as SOA (service-oriented architecture) or MDA (model-driven architecture)
- Proposal
  - New solutions involving the intersections of traditional software engineering and other disciplines including fields concerned with people—microeconomics, biology, city planning, anthropology

14

ULS Sources

- **Scale Changes Everything**
  by Linda Northrop
  Director, Product Line Systems Program Software Engineering Institute
  OOPSLA 2006 Presentation, Oct 24, 2006
- **Ultra-Large-Scale Systems**
  The Software Challenge of the Future
  by Linda Northrop et al.
  SEI Technical Report, June 2006
  http://www.sei.cmu.edu/uls

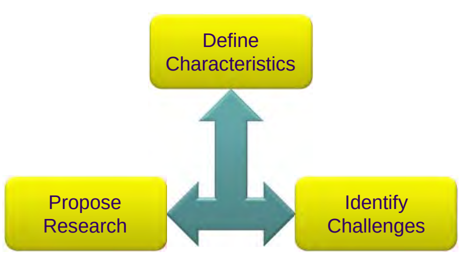
15

ULS Research Agenda

- Describes
  - the characteristics of ULS systems
  - the associated challenges
  - promising research areas and topics
- Is based on new perspectives needed to address the problems associated with ULS systems.

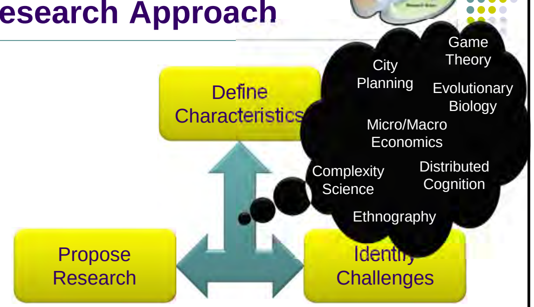
L. Northrop. Scale Changes Everything. OOPSLA 2006

Research Approach



L. Northrop. Scale Changes Everything. OOPSLA 2006

Research Approach



L. Northrop. Scale Changes Everything. OOPSLA 2006

Characteristics of ULS Systems

- Ultra-large size in terms of
  - Lines of code
  - Amount of data stored, accessed, manipulated, and refined
  - Number of connections and interdependencies
  - Number of hardware elements
  - Number of computational elements
  - Number of system purposes and user perception of these purposes
  - Number of routine processes, interactions, and "emergent behaviours"
  - Number of (overlapping) policy domains and enforceable mechanisms
  - Number of people involved in some way
  - ……

19

What is an ULS System

- A ULS System has unprecedented scale in some of these dimensions
  - Lines of code
  - Amount of data stored, accessed, manipulated, and refined
  - Number of connections and interdependencies
  - Number of hardware elements
  - Number of computational elements
  - Number of system purposes and user perception of these purposes
  - Number of routine processes, interactions, and "emergent behaviours"
  - Number of (overlapping) policy domains and enforceable mechanisms
  - Number of people involved in some way

ULS systems will be interdependent webs of software-intensive systems, people, policies, cultures, and economics.

Scale Changes Everything

- Characteristics of ULS systems arise because of their scale
  - Decentralization
  - Inherently conflicting, unknowable, and diverse requirements
  - Continuous evolution and deployment
  - Heterogeneous, inconsistent, and changing elements
  - Erosion of the people/system boundary
  - Normal failures
  - New paradigms for acquisition and policy

These characteristics may appear in today's systems, but in ULS systems they dominate. These characteristics undermine the assumptions that underlie today's software engineering approaches.

Today's Approaches

- **The Engineering Perspective**—for large scale software-intensive systems
  - largely top-down and plan-driven
  - requirements/design/build cycle with standard well-defined processes
  - centrally controlled implementation and deployment
  - inherent validation and verification
- **The Agile Perspective**—proven for smaller software projects
  - fast cycle/frequent delivery/test driven
  - simple designs embracing future change and refactoring
  - small teams and retrospective to enable team learning
  - tacit knowledge

Today's approaches are based on perspectives that fundamentally do not cope with the new characteristics arising from ultra-large scale.

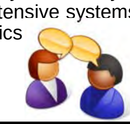
From Buildings to Cities

- Designing a large software system is like building a single, large building or a single infrastructure—power, water distribution

Ruins under Rome: In Rome's Basement, National Geographic, 2006

ULS Systems Operate More Like Cities

- Built or conceived by many individuals over long periods of time (Rome)
- The form of the city is not specified by requirements, but loosely coordinated and regulated—zoning laws, building codes, economic incentives (change over time)
- Every day in every city construction is going on, repairs are taking place, modifications are being made—yet, the cities continue to function
- ULS systems will not simply be bigger systems: they will be interdependent webs of software-intensive systems, people, policies, cultures, and economics



24

New Perspectives Are Needed

"The older is not always a reliable model for the newer, the smaller for the larger, or the simpler for the more complex…Making something greater than any existing thing necessarily involves going beyond experience."

Henry Petroski
Pushing the Limits: New Adventures in Engineering

The mentality of looking backward doesn't scale.

Change of Perspective

- From satisfaction of requirements through traditional, top-down engineering

The system shall do this … but it may do this … as long as it does this …

- To satisfaction of requirements by regulation of complex, decentralized systems

How? **With adaptive systems and feedback loops ☺**

26

ULS Sources

- **Scale Changes Everything**
  by Linda Northrop
  Director, Product Line Systems Program
  Software Engineering Institute
  OOPSLA 2006 Presentation, Oct 24, 2006
- **Ultra-Large-Scale Systems**
  The Software Challenge of the Future
  by Linda Northrop et al.
  SEI Technical Report, June 2006
  http://www.sei.cmu.edu/uls

27