Food for Adaptive Thinking…

Working lunch

Session Chair: Betty Cheng

Breakout groups

- Stimulate discussions
  - Learn about capabilities and needs from other disciplines
  - Learn more about the utility and limitations of your area
- Identify challenge problems
- Stimulate new research investigations

Breakout groups

1. Assurance
   - Discussion Leaders: Rogerio and Jeff M.
   - Structural, behavioral, NFR (security, fault tolerance)
2. Discipline-inspired adaptation
   - Discussion Leader: Betty
   - Biology, AI, control theory, robotics
3. Technology-enabled adaptation
   - Discussion Leaders: David G, Marin L.
   - (e.g., patterns, model-based development, process, architectural styles/frameworks, languages, etc.)
4. Application-driven adaptation
   - Discussion Leaders: Steve F., Hausi
   - (e.g., web, servers, grid computing, etc.)

Process

- Decide on a breakout group:
  - You work in the topic area OR
  - You want to learn about the area (bringing your expertise)
- Discussion leader(s) will select a table
- Identify a scribe for each table
- Each group will produce 1-3 slides
- Elected group member to present after lunch - 5 minutes.

Breakout Product

1. Describe 3 key challenges for the area
2. What existing work can be leveraged?
3. Anticipated near-term results
4. Ideal long-term results

Discipline Inspired

The Betty Posse

Challenges

- Decision Making
- Knowledge Representation / Model
- Scope:
  - Global vs. Local
  - Centralized vs. Decentralized
- Common / Standard Evaluation

Near-term Solutions

- AI Techniques
- Decision Theory
- Multi-attribute Utility Theory
- Evolution

Ideal Solutions

- Automatic generation
  - goals
  - requirements
  - design models
  - code
- Clear mapping between artifacts (goals to requirements to models to code)

Technology challenges

- Understanding the level of adaptations
  - code level, component level, system level
  - Tuning, structure change

Technology challenges

- Multidimensional aspects of adaptability
  - Composition of different concerns
    - Perspective on performance, reliability
  - Solutions:
    - Domain specific approaches
    - Separation of concerns
    - Composition of aspects (utility function)

Technology challenges

- What are the enabling technogies
  - SE tchnologies: architectures, AOP, languages, etc
  - Other domains technologies: AI learning, Biology, performance models, reliability models
  -

Technology challenges

- How to deal with expected/unexpected changes in the environment
  - Planning in advance for anticipated changes
    - Design the system accordingly
  - Solve the unexpected changes through evolution\

Application-Driven

- Undoing an adaptation
  - Short-term
    - Application-specific approaches
    - Logging and compensating actions
  - Long-term
    - Integrate into requirements and check adaptation meets requirements

Application-Driven

- Centralized vs distributed control
  - Short-term
    - Map out specific application scenarios
  - Ideal
    - Define spectrum of control models to match requirements

Application-Driven

- Evolution of non-functional requirements
  - – Short-term
    - Model variability of requirements
  - – Ideal
    - Include variability into architecture

Application-Driven

- Making legacy systems adaptive
  - Short-term
    - Wrapping
  - Long-term
    - SOA

Application-Driven

- Effective and adaptive monitoring
    - Short-term
        - Facilities to control intensity
    - Long-term
        - Intelligent control

Assurances

- what are the differences for the provision of assurances:
  - traditional systems;
    - difficult to achieve in high assurance systems [Littlewood keynote];
      - e.g., nuclear power stations;
  - self-adaptive systems;
    - order of magnitude much higher;
    - e.g., run-time testing;
- what should be adapted? and why?

Assurances

- methodology for supporting self-adaptability;
  - design-time;
    - what needs to be modified?
    - arguments based on model checking, testing, code inspections, etc.
  - run-time;
    - what should be incorporated?
    - design of emergent behaviours:
      - guarantees that the emergent
- domain specific solutions;
  - industry want guarantees on systems/solutions

Assurances - Challenges

- – real-time constraints;
  - how to ensure end-to-end arguments;
- – adaptive assurance;
  - how to generate assurances at run-time;
    - – there is a difference between design-time and run-time assurances;
  - level of assurance defined on the goals of the self-adaptive system;
- – oracle for assurances:
  - based on domain knowledge;