Self-adaptive Python based middleware

Md. Muhaimeen Ashraf
July 7, 2010

University of Waterloo, ECE Dept.

ECE 750-T11 Group 2
First presentation
Outline

1. Introduction
2. Problem description
3. Proposed approach
4. Expected results
5. Evaluation criteria
6. Expected results and contributions
7. References
Introduction

Software systems
- increasing in complexity
- deployed in heterogeneous environments
- unpredictable runtime behaviors
- need to adapt to changing conditions and requirements

Self-adaptive system
- able to modify its structure and behavior dynamically in response to changes in its execution environment
- parameter and compositional adaptation

Middleware
- layer of services separating application from operating system, network, and other components
- can intercept and modify communication between components
Problem description

- making a self-adaptive Python based middleware
- most literature talks about Java and .NET
- fill-in gap for Python based self-adaptive system
- Python is used widely (Django web framework)
- use dynamic adaptation derived from concepts in:
  - control theory, and
  - artificial intelligence

Related work:
- not much in literature using Python
- PyMX is a Python analog to Java's JMX
- StarMX - noteworthy reference for Java based system
Proposed approach

- create three-tier web application
- focus on Python based middleware
- apply dynamic adaptation techniques from disciplines of control
  - theory and artificial intelligence

Control theory:
- interact with World using sense-response-act loop:
  - feedback, adaptation, and reconfiguration

Artificial intelligence:
- Rule-based reasoning
- Artificial neural networks
- Evolutionary computation
Evaluation criteria

Autonomic characteristics:
- Stability and Robustness
- Degree of autonomy
- Control scope
- Self-* properties
- Management logic expression
- Runtime modification

Quality attributes:
- Performance
- Flexibility
- Scalability
- Usability
- Reusability
- Extensibility
Expected results / contributions

- outcome of the project will be a self-adaptive Python based middleware to changing run-time conditions
- project is experimental in nature
- results would help software developers choose optimal dynamic adaptation techniques from control theory and artificial intelligence in making Python based middleware
- contribute as implementations or case studies to evaluate models proposed by researchers in designing self-adaptive systems using control theory and artificial intelligence
References