Designing and Implementing Adaptive Distributed Systems

Richard Schlichting, Principal Investigator

Matti Hiltunen, Research Associate

The University of Arizona

rick/hiltunen@cs.arizona.edu

http://www.cs.arizona.edu/ftol/adaptive

Motivation and Outline

An adaptive system changes its behavior as a result of changes in the execution environment.

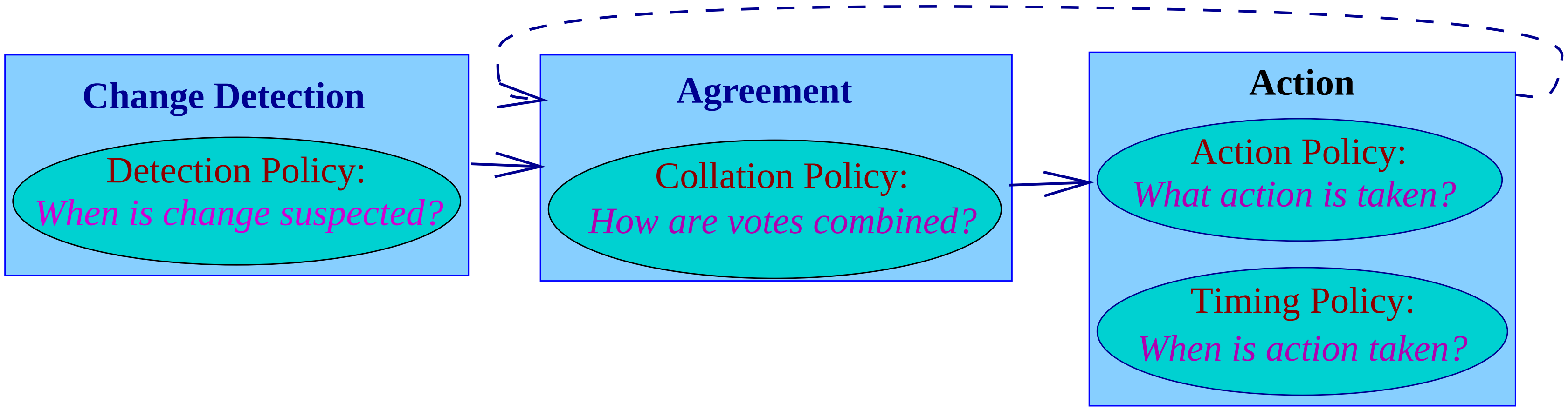
Examples: Processor or link failures, changes in communication patterns or frequency, and changed user requirements.

Objective: To develop fundamental new techniques for designing and implementing adaptive computing systems on distributed platforms.

Research components:

1. Conceptual model for adaptive systems.
2. Implementation using micro-protocol approach.

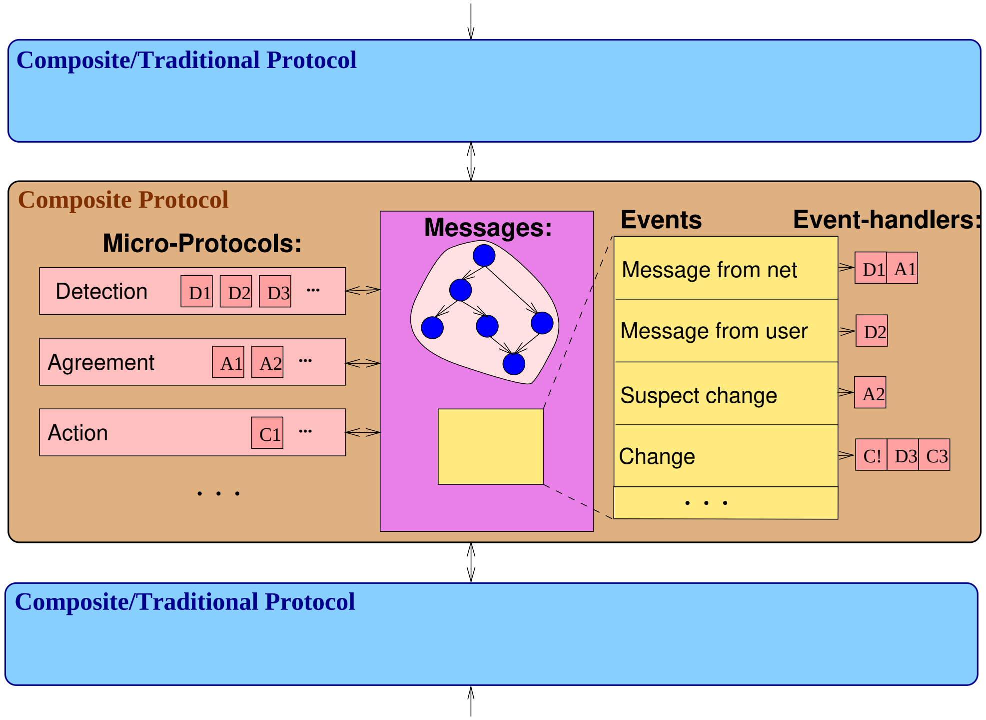
Conceptual Model



Micro-protocols

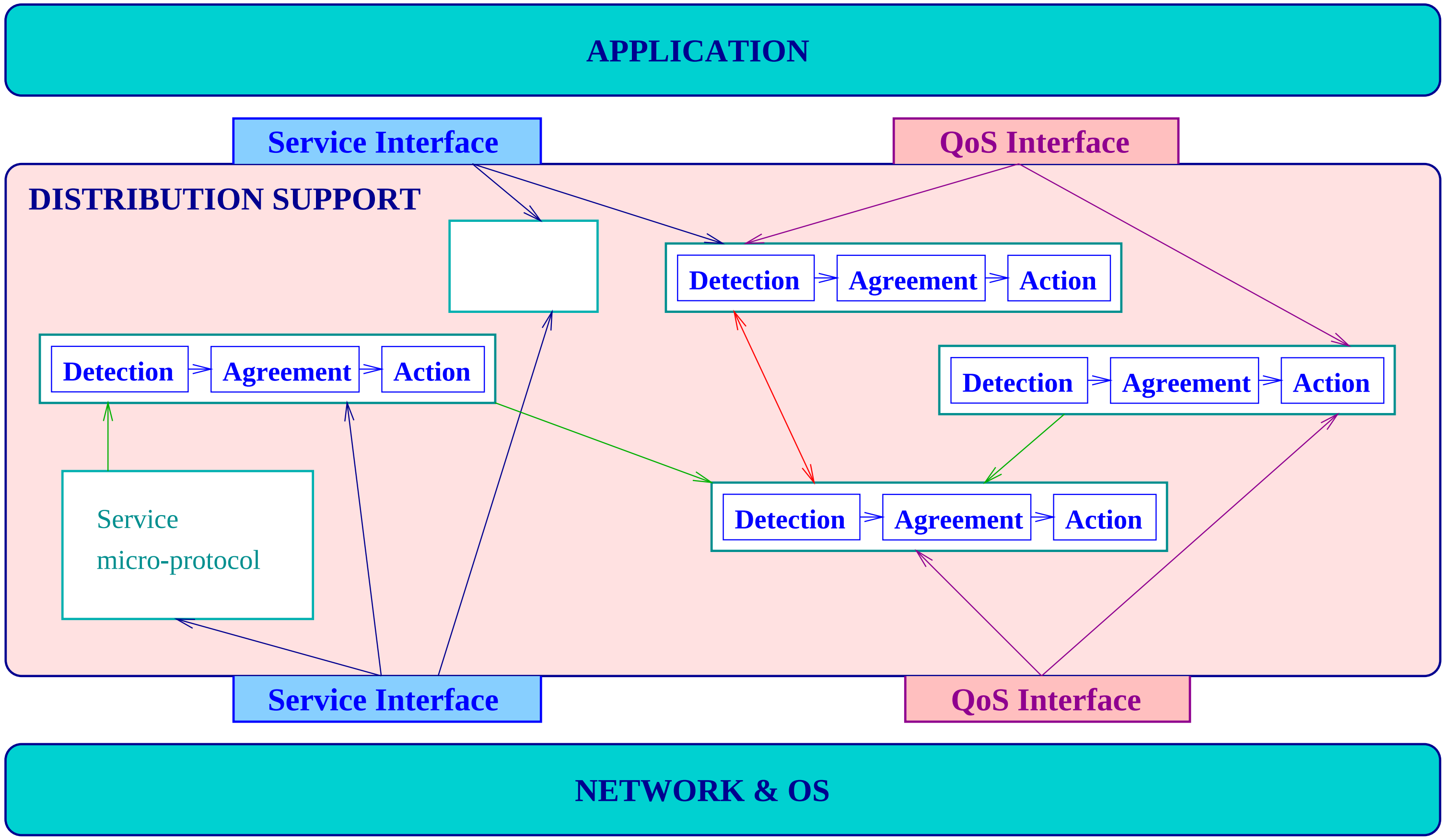
Developed for the construction of configurable fault-tolerant distributed services.

Concepts: Composite protocols, micro-protocols, and event-driven model.

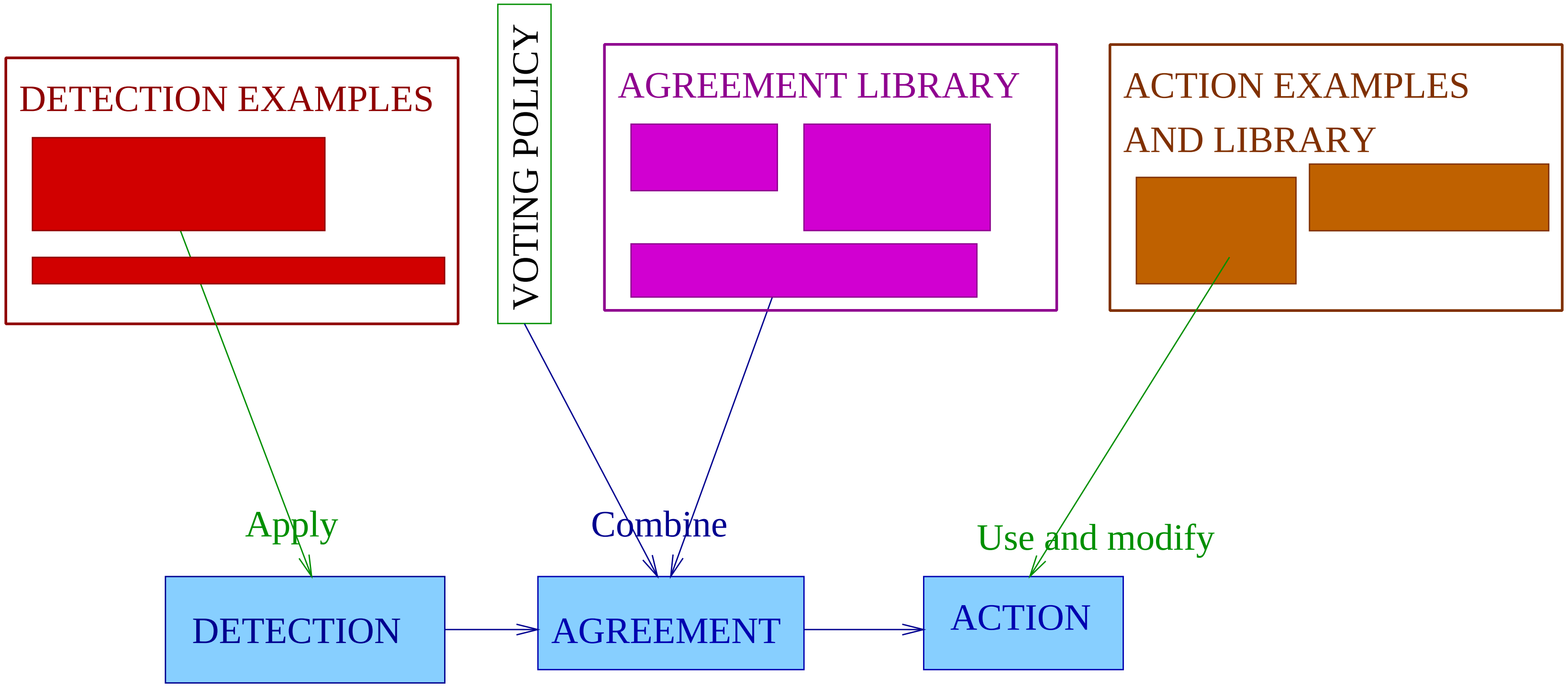


Features: Modules for different phases, interaction between phases through events, powerful adaptations (dynamic event/handler binding, etc.).

System components



Constructing an Adaptation



Software Prototypes

Testing and validation platforms.

x-kernel prototype:

- Composite protocol implemented as an *x*-kernel protocol object.
- Compatible with the standard distribution of *x*-kernel.
- Port planned to MK 7.2 from Open Group RI.

C++ prototype:

- Object-based implementation with base classes **CompositeProtocol** and **MicroProtocol**.
- Distributed architecture simulated; distributed version planned.

Deliverables and schedule

- Primary deliverables: Model, agreement and action libraries.
- Platforms: Currently C++ on Solaris, shortly MK 7.2 (Open Group).
- Conceptual work on:
  - a framework for using adaptations in layered distributed systems,
  - using adaptations as a method to reduce the cost of fault-tolerance.