Position Statement: Supporting Coordinated Adaptation in Networked Systems

Patrick G. Bridges Wen-Ke Chen
Department of Computer Science
The University of Arizona
Tucson, AZ 85721

Matti A. Hiltunen Richard D. Schlichting
AT&T Labs - Research
180 Park Avenue
Florham Park, NJ 07932

While adaptation is widely recognized as valuable, adaptations in most existing systems are limited to changing execution parameters in a single software module or on a single host. Our position is that the true potential of adaptation can only be realized if support is provided for more general solutions, including adaptations that span multiple hosts and multiple system components, and *algorithmic adaptations* that involve changing the underlying algorithms used by the system at runtime. Such a general solution must, however, address the difficult issues related to these types of adaptations. Adaptation by multiple related components, for example, must be coordinated so that these adaptations work together to implement consistent adaptation policies. Likewise, large-scale algorithmic adaptations need to be coordinated using graceful adaptation strategies in which as much normal processing as possible continues during the changeover. Here, we summarize our approach to addressing these problems in Cactus, a system for constructing highly-configurable distributed services and protocols [2].

When multiple related system components can adapt to changes in the system state, the adaptations performed by these components must be coordinated to achieve a consistent adaptation policy. To achieve this, we have implemented an *adaptation controller* architecture that is responsible for making adaptation decisions for related adaptive components. Adaptation policies are specified on a component-by-component basis using sets of fuzzy logic rules, and then composed along with rules to coordinate the actions of different components to form a single controller. The challenge, of course, is designing a set of fuzzy rules that reflect the best adaptation strategies for a given situation.

Even when coordinated adaptation decisions are made, large-scale algorithmic adaptations still present a difficult challenge. Without special provisions, for example, an adaptive system may be unable to process normal application traffic while it is changing between different algorithms. To alleviate this problem, we have designed and implemented a *graceful adaptation protocol* that coordinates changes across hosts and gracefully switches between algorithmic alternatives on each host. This protocol uses agreement, barrier synchronization, and message tagging to ensure that hosts reach consistent adaptation decisions and change between alternative algorithms with minimal disruption.

These techniques are being prototyped using Cactus, a design and implementation framework for constructing configurable services in networked systems. The graceful adaptation protocol and adaptation controller are currently being prototyped separately using different versions of Cactus. The controller is being implemented using the C version of Cactus 2.0. The initial focus is on coordinating layers for a test configuration consisting of a streaming video application layered on a configurable transport protocol, CTP [3]. Initial experimental results suggest that the controller is indeed successful in coordinating adaptation between multiple components. An initial version of the graceful adaptation protocol has been completed using the C++ version of Cactus 1.1. Preliminary experimental results using an adaptive group communication service suggest that the protocol does indeed provide a graceful transition from one adaptation aware module to another, and demontrate the overall value of adaptive strategies. Further details on the graceful adaptation protocol can be found in [1].

*This work supported in part by DARPA under grant N66001-97-C-8518 and NSF under grants ANI-9979438 and CDA-9500991.

References

[1] W.-K. Chen, M. Hiltunen, and R. Schlichting. Constructing adaptive software in distributed systems. In *Proceedings of the 21st International Conference on Distributed Computing Systems*, Phoenix, AZ, Apr 2001.

[2] M. Hiltunen, R. Schlichting, and G. Wong. Cactus system software release. http://www.cs.arizona.edu/cactus/software.html, Dec 2000.

[3] G. Wong, M. Hiltunen, and R. Schlichting. CTP: A configurable and extensible transport protocol. In *Proceedings of the 20th Annual Conference of IEEE Communications and Computer Societies (INFOCOM 2001)*, Anchorage, Alaska, Apr 2001.