Specifying Weak Sets

Jeannette M. Wing David C. Steere

September, 1994
CMU-CS-94-194

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Submitted to the 15th International Conference on
Distributed Computing Systems

Wing is supported in part by the Avionics Lab, Wright Research and Development Center, Aeronautical Systems Division (AFSC), U. S. Air Force, Wright-Patterson AFB, OH 45433-6543 under Contract F33615-90-C-1465, ARPA Order No. 7597. Steere is supported in part by the Advanced Research Projects Agency (Hanscom Air Force Base under Contract F19628-93-C-0193, ARPA Order No. A700; and the Wright Laboratory, Aeronautical Systems Center, Air Force Materiel Command, USAF, under grant number F33615-93-1-1330), IBM Corporation, Digital Equipment Corporation, Intel Corporation, and Bellcore.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Wright Laboratory or the U. S. Government.

Keywords: Search, Distributed Systems, Formal Specification, Specification of Iterators, Larch Shared Language, Dynamic Sets

Abstract

We present formal specifications of a new abstraction, *weak sets*, which can be used to alleviate high latencies when retrieving data from a wide-area information system like the World Wide Web. In the presence of failures, concurrency, and distribution, clients performing queries may observe behavior that is inconsistent with the stringent semantic requirements of mathematical sets. For example, an element retrieved and returned to the client may be subsequently deleted before the query terminates. We chose to specify formally the behavior of weak sets because we wanted to understand the varying degrees of inconsistency clients might be willing to tolerate and to understand the tradeoff between providing strong consistency guarantees and implementing weak sets efficiently. Our specification assertion language uses a novel construct that lets us model *reachability* explicitly; with it, we can distinguish between the existence of an object and its accessibility. The specifications were instrumental in understanding the design space, and we are currently implementing the most permissive of the specifications in several types of Unix systems.