Cronus: A Distributed Computing Environment

Dr. Stephen Vinter

ABSTRACT

Cronus is a distributed computing environment that supports the development and operation of applications that use diverse hardware and software resources in an internetwork environment. Primarily funded by the Rome Air Development Center for the SDI program, Cronus was developed to make the integration of large distributed applications, such as command and control systems, feasible to build, maintain, and operate.

One measure of its feasibility is its ability to operate in heterogeneous environments that include different network architectures, host types, operating systems, and programming languages. Its portability and interoperability make it possible to incorporate existing systems and software at low cost, and to exploit the unique capabilities of each system (e.g., sophisticated user interfaces, multiprocessors, and database systems). Further, Cronus augments rather than replaces existing operating systems, making it possible to retain use of all existing software.

A second measure of its feasibility is the support it provides for building distributed applications. Cronus supplies a set of distributed operating system services, including global naming, monitoring and control, and distributed files, that are commonly needed by applications. Automated support is also available for building distributed applications. Tasks such as message conversion and parsing, access control, and replication are implemented by a set of development tools that customize each task to the application component. Software development is greatly simplified through the adoption of an object-oriented architecture and programming paradigm. This approach provides a unifying framework for defining a global, coherent view of the heterogeneous environment.

Last, Cronus provides a range of mechanisms to support applications that evolve, scale, and are capable of surviving component failures. Components are able to evolve in isolation because the architecture requires and enforces strict modularization. The ability of applications to scale is achieved by making it possible to partition and distribute resources among many different sites without added complexity within the application. And survivability is achieved by a set of fault tolerant mechanisms that include replication and dynamic location of available resources.

Cronus is in use in 10 testbeds on over 100 hosts. The types of hosts currently supported include Sun, Vax (running Ultrix, BSD Unix, and VMS), Masscomp, and Symbolics. Our applications work has included the integration of databases, knowledge bases, and expert systems. This talk will address the capabilities of Cronus, some of its existing applications, and plans for future development.