Technical Overview of the CHORUS Family of Operating Systems

The Integration of Real-time, Distributed Processing and UNIX

The CHORUS technology integrates inter-process communication into the heart of its real-time operating system. It is this integration that gives CHORUS its essential properties: real-time; transparent distribution of, and access to, programs and files over a network or bus; modularity; and the efficient integration of UNIX and other open system standards into full real-time distributed operating systems.

The CHORUS family of operating systems is centered around the small real-time distributed CHORUS **Nucleus** which provides a minimal set of essential generic services. The other members of the family are built from the Nucleus and inherit its real-time distributed computing capabilities. The **CHORUS/MIX** operating system incorporates the CHORUS Nucleus as a base on which it provides a UNIX interface that is transparently extended to distributed processing and to use in real-time environments.

The key characteristics of the CHORUS family of operating systems are:

— *Real-time*: CHORUS systems are built on the real-time CHORUS Nucleus and have all the functional characteristics and performance of classic real-time executives.

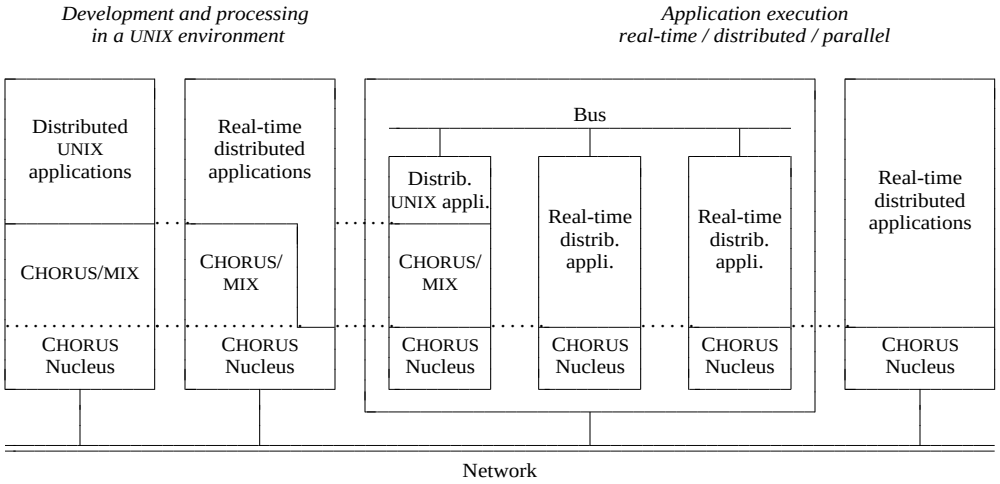
— *Controlled transparent distribution* of processing and of data: the management of distribution can be entirely taken care of by CHORUS systems, yet controlled by network administrators. In particular, CHORUS permits dynamically reconfiguring the system and applications.

— *Modularity*: CHORUS systems are composed of a set of communicating modules that can be assembled and reconfigured dynamically depending on the hardware capability and application needs.

— *Openness*: Both the CHORUS Nucleus and CHORUS/MIX support the open communications standards OSI and TCP/IP. In addition, CHORUS/MIX provides all the functionality of a standard UNIX system (X/Open). The CHORUS Nucleus can also host other operating system interfaces.

— *Compatibility* with UNIX is provided by CHORUS/MIX: application programs, utilities, and files used under UNIX are supported without modification (binary compatibility) by CHORUS/MIX.

The immediate domains of application for CHORUS products are those for which the limits of traditional operating systems are preventing the development of fully satisfactory solutions. This includes especially the development of advanced real-time, industrial and technical systems (process-control, command stations, computer-aided automation, test and measurement, aerospace, telecommunications, and military systems) and specialized systems (dedicated servers, multiprocessors, parallel machines, and multicomputers).



The CHORUS Nucleus and subsystems

Structure of CHORUS systems

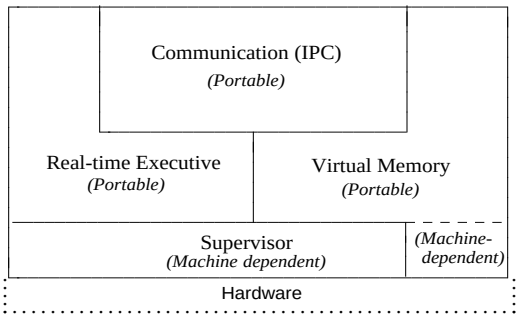
CHORUS systems are composed of a small real-time distributed *Nucleus* (on the order of 50K bytes) and one or several *subsystems* offering application programs standard operating system interfaces. The first of these subsystems, the UNIX subsystem, is included in the CHORUS/MIX product.

The Nucleus

The CHORUS Nucleus is built around a *real-time executive* that supplies the minimum set of functions that are needed to support the smallest system configurations. The real-time executive provides multi-tasking (CHORUS *threads*), schedules threads by their priorities, manages the preemption of running threads by interrupts and provides primitives for synchronization and for low-level local communication between threads.

The real-time executive is augmented with:

- A low-level hardware supervisor permitting the dynamic association of interrupt handling code to interrupts.
- An Inter-process communication (IPC) facility for communicating by exchanging *messages* between *ports* attached to groups of threads in a given address space (CHORUS *actors*). This IPC facility offers a uniform vision of communication that is transparent to the distribution (location) of the communicating threads.
- A distributed *virtual memory* manager for those applications having such a need and running on processors equipped with the necessary hardware.



Structure of the CHORUS Nucleus

Subsystems

The basic services of the CHORUS Nucleus can be enriched by adding *subsystems* that supply complete operating system interfaces to application programs. A subsystem is a set of system servers that use the Nucleus's generic services to provide higher-level services.

The first subsystem available is the UNIX subsystem used by the CHORUS/MIX system to provide a standard UNIX (X/Open) environment extended to be a distributed system that respects real-time constraints for local operations.

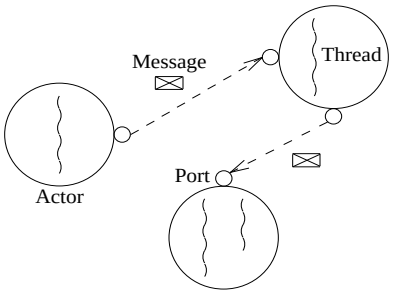
The CHORUS Nucleus

The CHORUS Nucleus offers the functionality and performance of a classic real-time executive.

Actors and Threads

The CHORUS Nucleus permits the execution of *threads* (execution contexts) that share the resources of a given *actor* (address space). The threads communicate between themselves independently of their location by using the CHORUS IPC. Privileged "system" threads may be installed directly into the Nucleus address space.

Threads and actors are used in exactly the same way to implement system functions as to support user processes.



Actors, threads, ports and messages

Thread management

The CHORUS Nucleus provides calls to create and delete threads, to suspend and resume their execution, etc....

The scheduling of threads is preemptive. Each thread receives a priority level at its creation; when several threads are ready to be executed, the Nucleus chooses the thread with the highest priority; for threads of equal priority, the threads are scheduled in FIFO (first-in, first-out) order. The Nucleus reschedules each time an interrupt or exception is produced and each time a thread makes a system call to the Nucleus that changes the current thread (i.e., suspend, etc.).

Structuring of threads

The CHORUS Nucleus groups threads into *actors* that provide to each of their threads a private execution context (protected address space).

Communication and synchronization

The CHORUS Nucleus offers two modes of communication and synchronization:

— The threads of a given actor share that actor's address space. They can communicate directly among themselves by using shared memory. CHORUS includes synchronization mechanisms that can be used to ensure the proper coordination of threads:
  - mutual exclusion semaphores,
  - execution control (suspend / resume) of a thread.

— The Nucleus also provides a message-based communication mechanism that can be used between any thread – in a given actor or in various actors, regardless of which machine they may be on in a network. The principle characteristics of this communication service are:
  - the communication interface is independent of the localization of the threads and thus permits configuration changes without having to modify the code of the threads,
  - messages are exchanged between *ports* or *groups of ports*, with addressing modes specially adapted to distributed applications and without reducing the performance of local communication,
  - communication may be asynchronous or synchronous (Remote Procedure Call – RPC).

  The exchange of messages is equally a method of synchronizing threads, whether the threads are located in different actors or in the same actor.

Memory management

The CHORUS Nucleus offers mechanisms for allocating and deallocating physical memory.

When the hardware permits and the applications need it, the Nucleus can also manage *virtual memory*. Specified pages can be locked into physical memory in order to guarantee the response time of threads operating under strong real-time constraints.

Interrupt handling

The CHORUS Nucleus allows connecting user-written routines to machine interrupts; these routines can be installed and replaced dynamically and do not need to be linked into the Nucleus. This ability is useful for debugging and maintenance as well as for adapting to hardware reconfigurations.

The interrupt routines can trigger (signal or start) threads which are then scheduled by the Nucleus according to their relative priorities.

The CHORUS/MIX system

CHORUS/MIX supplies a standard UNIX environment by integrating a UNIX subsystem with the real-time CHORUS Nucleus. The CHORUS/MIX UNIX services conform to the X/Open specifications and have been extended to the distributed environment (distribution of programs as well as files) and to real-time in a manner that is completely transparent to UNIX application programs.

Compatibility with UNIX

CHORUS/MIX is compatible with UNIX at three levels:

— *system call compatibility:* CHORUS/MIX is compatible with UNIX system calls:
  - at the source code level for all programs written for UNIX and following the X/Open base level standard,
  - at the executable binary code level for certain reference UNIX machines serving as porting bases;

— *file compatibility:* CHORUS/MIX offers total compatibility with UNIX files: files created under UNIX can be used just as they are in the CHORUS/MIX environment;

— *device driver compatibility:* CHORUS/MIX offers functional compatibility with UNIX device drivers, reducing considerably the cost of porting existing UNIX drivers to a CHORUS/MIX system.

Signal management

CHORUS/MIX offers the UNIX set of functions for managing signals, but extends them to manage real-time processes and distributed processes. This extension is transparent and permits the remote creation of processes on any machine, while respecting the inheritance rules of UNIX (environment, open files, etc.), and permits "manipulating" those processes at a distance.

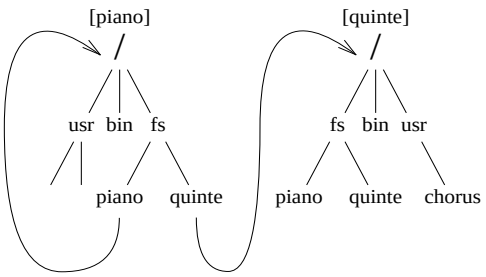
CHORUS/MIX provides the traditional UNIX communication mechanisms between processes and extends them to a distributed environment in a manner that is transparent to applications. Furthermore, CHORUS/MIX provides a mechanism for message-based communication between processes (IPC) that is particularly well adapted to distribution.

File management

CHORUS/MIX offers the UNIX set of file management functions and extends them, transparently, to supply a true *distributed file system*:

— *The automatic interconnection of local file trees* permits accessing any file located on any machine on the network;

— *Symbolic links* permit masking the distribution of files and their placement, to "personalize" naming and to be able to reconfigure systems in a manner transparent to applications and users.



Interconnection of file trees

Utilities

CHORUS/MIX supports the UNIX utilities defined by the X/Open Portability Guide (for program development, the shell, for file manipulation, etc.) and extends them to distribution. So, for example, the file and device manipulation utilities offered by CHORUS/MIX permit manipulating distributed files and devices as if they were local UNIX files and devices, without being concerned with their location.

Development environment

The integration of the standard UNIX environment and its extension allow CHORUS/MIX to offer a software development environment that is particularly rich, including utilities such as:

- editors,
- configuration management systems,
- symbolic debuggers extended to debug distributed processes,
- *CHORUS Simulator* on a native UNIX,
- *Communication Debugger* (under development),
- etc.

CHORUS open operating systems

CHORUS systems support international open standards, not only in terms of the system interface (UNIX with the CHORUS/MIX product), but also in terms of communications* (OSI/TOP, TCP/IP).

CHORUS/MIX supports the traditional UNIX protocols that permit exchanges between UNIX machines.

Because of their modularity and clear organization, all systems built using the CHORUS technology are open to extension and evolution: to support future open system standards and new machine architectures, to adapt to particular user needs, to integrate new functionality, etc.

---

* The CHORUS support for communication standards is implemented using the OSIAM protocol management software from Marben.

CHORUS is a registered trademark of Chorus systèmes.
CHORUS/MIX is a trademark of Chorus systèmes.
UNIX is a registered trademark of AT&T.
Reorder part number: CS/TR-89-10.5
Specifications are subject to change without notice.
© 1989 Chorus systèmes. Printed in France.