UNICOS/mk System Administration

Jim Harrell, Cray Research, Inc., 655-F Lone Oak Drive, Eagan,
Minnesota 55121

ABSTRACT: *UNICOS is being reorganized into a microkernel-based system. The purpose of this reorganization is to provide an operating system that can be used on all Cray architectures and provide both the current UNICOS functionality and a path to the future distributed systems. UNICOS/mk will initially be supported on the CRAY T3E platform. The CRAY T3E system, besides being an MPP system, has a new I/O subsystem (referred to as the GigaRing) and the associated I/O nodes, and a new System Workstation (SWS).*

The reorganization of UNICOS does not make any substantive changes to the UNICOS user interface. The UNICOS filesystems and binaries are running on our parallel vector processor test platforms. The system administration interface has undergone some modifications, which have resulted mainly from the differences in hardware. These changes and a few simplification efforts are discussed in this paper. The discussion is focused on mainframe-related changes.

1 Introduction

The UNICOS/mk system will be released soon on CRAY T3E systems. This system is a modularized version of UNICOS and is composed mainly of code from UNICOS. A key design goal was to have the look and feel of UNICOS, and on parallel vector machines to be binary compatible. (There is no plan to release the parallel vector version of UNICOS/mk at this time, however.)

The user interface for UNICOS/mk is an extension of the UNICOS user interface. The UNICOS/mk system administration interface is also an extension of UNICOS. There are changes to the administrative interface and this is a discussion of some of those changes.

There were two major reasons for changing parts of system administration. The first is the differences in the MPP hardware. In addition, the CRAY T3E also introduces a new I/O subsystem and a new System Workstation (SWS) that combines the functions of the Operator Workstation (OWS) and the Maintenance Workstation (MWS) currently used on Cray systems.

The second reason for changes in system administration was the opportunity to simplify a few areas. One area in particular that has been simplified is configuration and installation.

The discussion in the paper focuses on mainframe-related issues. The changes related to the I/O subsystem and SWS are not addressed directly, but only as they relate to mainframe issues.

2 Areas of Unchanged System Administration

There is much in system administration that did not change. A few of the areas that are not affected are as follows:

- NQS
- User creation
- File system and file administration
- Network administration
- Disk and tape administration

In the case of devices, some changes are required because of differences in the new devices provided with the new I/O subsystem. The scope of the changes are the same as would be expected with any new devices.

Obviously, the areas not changed cover a great deal of system administration and should give an impression of the commitment to maintaining the same interfaces. It is also true that using code from UNICOS made it easier to maintain the interfaces because the code instantiated the interfaces.

3 Accounting

An important area of system administration is accounting. Our original plan was to initially provide only UNIX-style accounting, which is limited in capabilities and granularity of reporting. However, we were able to port all of UNICOS accounting and it will be available with the initial releases.

We added a feature called remote CPU time, which was added to support the MPP system organization. The feature allows the tracking of system time used in servers that are not on the same processing element (PE) as the user process that made the request. It was felt that this feature was needed in order to correctly account and bill for system usage.

The multitasking integrals used in UNICOS accounting were deactivated because a PE does not have multiple processors. The SDS integrals were deactivated because there is no support for SDS in UNICOS/mk at this time.

Device accounting was removed because it is not being used in UNICOS and was scheduled for removal in a future release of UNICOS.

4 Hardware Error Management

Hardware error management is being reorganized. The hardware errors are being processed by a server called the error manager (EM). The EM processes all the hardware error interrupts and writes console and *syslog(*8) messages. The *syslog* system log is being used to coordinate all the various log functions. We expect that over time this will be a convenient and consistent way of tracking system messages.

5 Scheduling

The *nschedv* (8) command has been simplified because the swapping and memory management on the CRAY T3E system is much simpler than the UNICOS model in shared memory systems. In the future, as new capabilities are added to UNICOS/mk, further modifications to the *nschedv* command may be made to support memory management changes.

A higher level of scheduling, referred to as political scheduling, is supported in UNICOS through the fair-share scheduler. Political scheduling is based on the interests of the system user community and the allocation of resources based on user privileges.

In UNICOS/mk, this mechanism will be modified to support a multidimensional share hierarchy that allows several different independent share trees to coexist. This modification was thought necessary because on an MPP platform, some sites will want the capability of allocating shares in command PEs in a different hierarchy than in the application PEs where multi-PE applications are run.

These different trees would allow different share policies to be implemented on the different styles of usage. This new capability will also be useful in the future on distributed systems where different trees could be allocated on different machines within the system.

6 Configuration and Installation

There are currently many configuration mechanisms within UNICOS, almost as many as the number of subsystems. The complexity of MPP systems, including such things as the organization of the torus and PEs, and the requirements for new configurable information such as different clock speeds and memory sizes for PEs, required a way of reducing the difficulty of configuring MPP systems.

In order to accomplish this the configuration and installation of UNICOS/mk has been modified extensively, including changes that are not necessarily technical. The major technical goal of installation and configuration work is to provide a consistent mechanism to configure UNICOS/mk.

In addition, we also wanted to improve the maintainability of the configuration information. This is being accomplished by organizing the kernel and servers so that configurable information is automatically available to the installation tools, thus making the installation correctly configure the system.

The installation tools are composed of the following components:

- Configuration and Installation Tool (CIT)
- PArameter Configuation Tool (PACT)
- Configuration Tool (CT)

The CIT is a GUI-based interface, which runs on the SWS. CIT performs two functions. First, it loads the SWS software onto the Solaris-based system which comprises the SWS. Then it loads the initial UNICOS/mk system, the I/O subsystem binaries, the programming environment, and asynchronous products. The UNICOS/mk and programming environment packages are loaded onto the CRAY T3E disks, which results in a bootable system that is not customized for the site.

The PACT, which is also a GUI-based interface, uses the UNICOS/mk definition files (explained later) to provide the installer (and system administrator making configuration changes later) with templates, value ranges, and help information, that are used to create the parameter file used to configure UNICOS/mk at boot time.

The PACT provides a visual interface to the configuration server (CS). The CS (explained later) runs on both the SWS and the mainframe. The PACT is the single point of access to the operating system configuration information available through/to these servers.

The CT runs on the CRAY T3E system and loads and configures daemons, such as networks, tapes, and NQS.

Configuration changes have required modifications to existing UNICOS/mk code. The kernel and servers have new header files, called definition files.

The definition files describe configuration data structures, types, and definitions. The definition files are used both by all the servers, including the configuration server (CS). As mentioned previously, the CS runs on both the SWS and PACT, as part of UNICOS/mk in the CRAY T3E.

The CS in the SWS uses the definition files to organize the information into the parameter file. The parameter file is an instantiation of a configuration. At boot time, the parameter file is passed in with the initial binary. The CS on the mainframe is one of the first servers that is started. It loads the parameter file into a local memory database.

As the other servers start up they request their configuration information from the CS. Later, if changes are made to the configuration the CS and servers can be notified and act on the changes. (This part of the configuration feature is still under development.)

For the first shipments of UNICOS/mk, we want to provide a set of well-tested configurations that can be used as templates. We would like to preinstall all systems before shipment to reduce the amount of time it takes to get a system up and running on site.

7 Resource Management

The resource management of an MPP system is an important area of system administration, and an area that requires changes in the current UNICOS resource management interface. The scheduling and management of processes on a UNICOS/mk system focus on how processes are allocated to PEs, and how PEs are shared among processes.

On UNICOS/mk, PE allocation is handled by the global resource manager (GRM). The GRM uses attributes to decide how to allocate PEs. An attribute is a characteristic of a PE or an application, or a value computed from these characteristics.

There are three types of attributes used by the GRM:

- Resource attributes
- Application attributes
- Precedence attributes

Resource attributes are associated with the hardware and/or the software characteristics of the PE. Application attributes are associated with the program to be run. Finally, precedence attributes are computed from the first two attribute types and weights associated with the attribute.

Resource attributes are used to limit the processes that can be run on a PE, and can be classified as hardware and software attributes. These attributes are set by the administrator.

The hardware resource attributes include the following:

- PE memory size
- PE clock speed
- PE availability
- PE location in the torus

The software resource attributes include the following:

- Minimum and maximum PE counts for a multi-PE application
- Number of applications on a PE at one time
- Service types, that are initially set at job initiation time, like *batch*, *login*, *rsh*, *rexec*, and several reserved for site usage
- User and group identifier (UIDs and GIDs)

The service types are ways of identifying jobs and processes by origin or usage.

Application attributes are the requirements or preferences of the application. These are specified by the end user, implicitly for the user by UDB entry, or explicitly by a service provider, like NQS. Application attributes include the following:

- Initial memory size of the application
- Number of PEs needed
- Owner UID and GID
- Requested PE clock speed
- Service type.

Precedence attributes are computed by the GRM for each PE. These combine the application attributes with resource attributes. The system administrator can assign weights to each attribute to accommodate site specific needs.

Precedence attributes include the following:

- Memory fit, contiguity (keeping multi-PE applications in PEs that are close to each other, or contiguous)
- Speed match and uniformity
- Closeness to the minimum and maximum PE ranges for the partition.

When doing process scheduling, the most important resource attribute is the minimum and maximum PE counts. If a PE has been assigned a minimum PE count of four, only applications requiring four or more PEs can be assigned to it. An application needing two PEs cannot be assigned to this PE. If the same PE has been assigned a maximum PE count of sixteen then only processes requiring no more than sixteen PEs can be assigned to it.

Command PEs are special because they only run single PE processes. A command PE has the minimum and maximum PE count both set to one.

PEs can be clustered by attributes in order to create administrative regions; a region is a contiguous number of PEs. For example, on a 128 PE system, eight PEs could be allocated for interactive and single PE applications. The minimum and maximum PE counts would be set to one for these PEs, and because there might be several service types using these PEs the service type would be set to *any.*

There could also be a partition of 24 PEs with a minimum PE count of two and a maximum PE count of eight. The service type on these PEs could also be set to *any*. This partition could be used for smaller multi-PE applications and checkout or debugging of test multi-PE applications.

The remaining 96 PEs could be given a minimum PE count of eight and a maximum PE count of forty to allow larger applications to be run. The service type could be set to *batch only* to encourage a single first-in-first-out scheduling by NQS. At night these regions could be modified to support only large multi-PE applications by reducing the size of the command partition, and creating a single multi-PE partition with very large minimum and maximum PE counts. The system can make this change without requiring a reboot.

The GRM views the PEs as a linear array. The mapping of the 3D torus to a linear array is done during system boot. The initial PE configuration is obtained from the configuration server. Once the GRM has the initial configuration the application PEs are mapped into regions based on the attributes assigned to the PEs by the site.

The resource management features are being phased into UNICOS/mk over several releases in 1996. The initial 1.2 release will include the following:

- GRM infrastructure
- Support for the UNICOS UDB for assigning user-level attributes
- Administrator interfaces that allow dynamic configuration and adjustment of the GRM attributes.

Process scheduling on command PEs will be basically the same as on most UNIX systems. Multiple processes will be allocated to a given PE, and the operating system will handle context switching between processes in memory.

The system will also manage the swapping of processes between disk and memory. Processes will always be swapped back into the PE where it was previously running.

Process scheduling will be simple, similar to round-robin scheduling. The scheduling of application PEs will be similar to the current CRAY T3D system. A multi-PE application is assigned dedicated access to a set of PEs for the life of the application. Swapping or sharing of PEs among several multi-PE applications will not be initially supported.

In the UNICOS/mk 1.3 release, support will be provided for different memory sizes and clock speeds. The associated GRM attributes will be enabled.

In the UNICOS/mk 1.4 release, support for gang swapping will be provided for multi-PE processes. This will be similar to the CRAY T3D rolling feature.

In later releases, scheduling algorithms to manage the context switches between resident processes will be enhanced to support political scheduling. (The current fair-share scheduler on UNICOS is an example of a political scheduler.)

UNICOS/mk in 1996

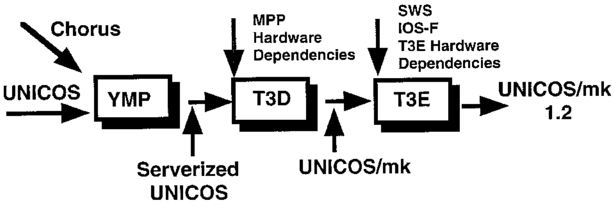
Jim Harrell, Project Leader
UNICOS/mk

UNICOS/mk in 1996

- "Pipeline" Methodology
  - "Pipeline" status
  - T3E status
- Illustrations and Evidence
  - LS-DYNA
  - UNICOS/UNIX compatibility
- Schedule
- Quality Improvement
  - Architecture and Design
  - Testing

CRAY RESEARCH Supercomputer ...delivering the performance

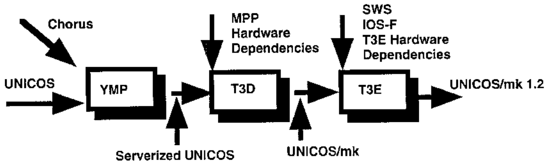
"Pipeline" Methodology



CRAY RESEARCH Supercomputer ...delivering the performance

"Pipeline" Status

- Feature content
- Usage
- Testing
- Customer access



CRAY RESEARCH Supercomputer ...delivering the performance

Feature Content

UNICOS 1.2 (April 15, 1996)

Feature	PAN Commitment	Status
Configurations: scalability for 16-256 PEs	Committed	Done
Process Management (most common UNICOS system calls, including asynchronous I/O)	Committed	Done
Single-PE memory management (process creation, malloc, and sbrk)	Committed	Done
Basic Scheduling (space sharing of multi-PE codes; time sharing of single PE codes)	Committed	Done
Swapping of single-PE codes (no swapping of multi-PE codes)	Committed	
UNICOS filesystem (NC1)	Committed	Done
TCP/IP	Committed	Done
Limited Kernel Statistics (traditional sar/sam tsar statistics, with some diffs from current UNICOS)	Committed	Done
Signals (POSIX/UNICOS compatible)	Committed	Done

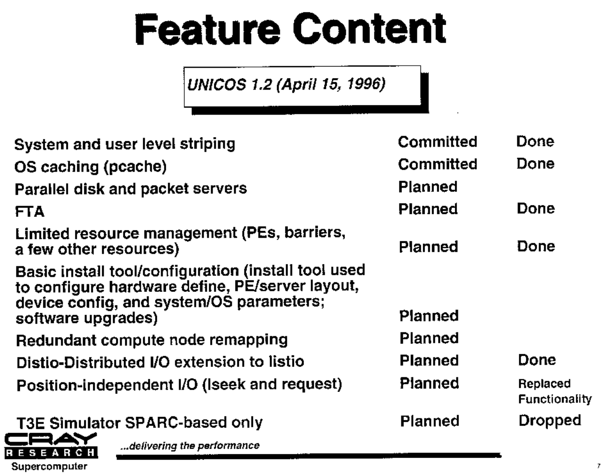
CRAY RESEARCH Supercomputer ...delivering the performance

Feature Content

UNICOS 1.2 (April 15, 1996)

Terminal/console (support for console and tty through SWS)	Committed	Done
Boot/dump/monitor	Committed	Done
Basic Accounting (UNIX style accounting)	Committed	Done
MPP launch (start multi-PE codes)	Committed	Done
Kernel logging of system messages	Committed	Done
Common Commands (UNICOS common command set; except for a few commands tied to phased-in UNICOS features, such as chkpnt/restart, and other exceptions)	Committed	Done
NQS Phase I (missing: some limits, fair share scheduling, MLS support, chkpnt/restart)	Committed	Done
NFS server	Committed	Done
NFS client	Committed	Done
Tape drivers (all current UNICOS tape devices)	Committed	

CRAY RESEARCH Supercomputer ...delivering the performance

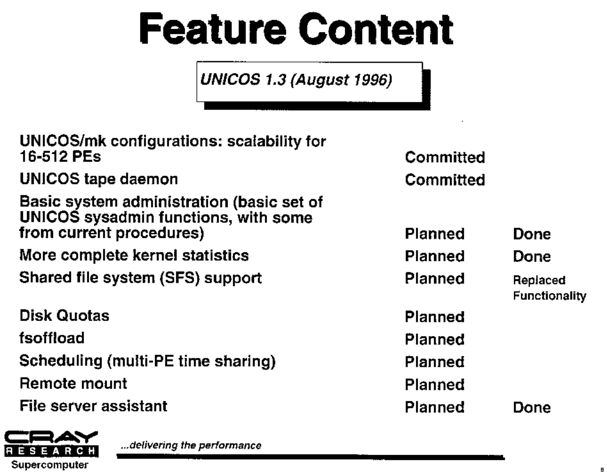


Feature Content

UNICOS 1.2 (April 15, 1996)

System and user level striping	Committed	Done
OS caching (pcache)	Committed	Done
Parallel disk and packet servers	Planned	
FTA	Planned	Done
Limited resource management (PEs, barriers, a few other resources)	Planned	Done
Basic install tool/configuration (install tool used to configure hardware define, PE/server layout, device config, and system/OS parameters; software upgrades)	Planned	
Redundant compute node remapping	Planned	
Distio-Distributed I/O extension to listio	Planned	Done
Position-independent I/O (lseek and request)	Planned	Replaced Functionality
T3E Simulator SPARC-based only	Planned	Dropped

CRAY RESEARCH Supercomputer ...delivering the performance

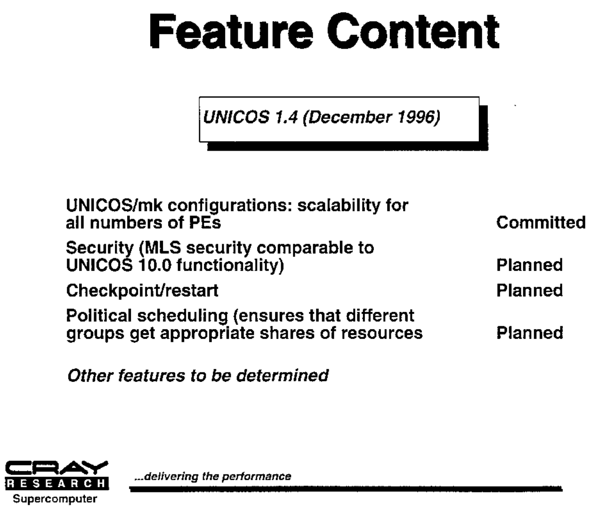


Feature Content

UNICOS 1.3 (August 1996)

UNICOS/mk configurations: scalability for 16-512 PEs	Committed	
UNICOS tape daemon	Committed	
Basic system administration (basic set of UNICOS sysadmin functions, with some from current procedures)	Planned	Done
More complete kernel statistics	Planned	Done
Shared file system (SFS) support	Planned	Replaced Functionality
Disk Quotas	Planned	
fsoffload	Planned	
Scheduling (multi-PE time sharing)	Planned	
Remote mount	Planned	
File server assistant	Planned	Done

CRAY RESEARCH Supercomputer ...delivering the performance



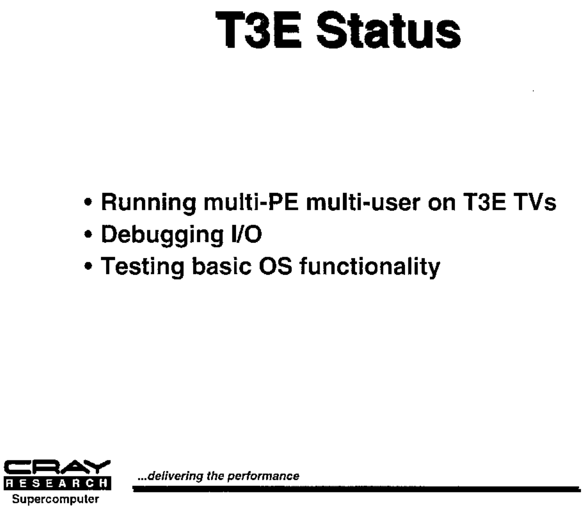
Feature Content

UNICOS 1.4 (December 1996)

UNICOS/mk configurations: scalability for all numbers of PEs	Committed
Security (MLS security comparable to UNICOS 10.0 functionality)	Planned
Checkpoint/restart	Planned
Political scheduling (ensures that different groups get appropriate shares of resources	Planned

Other features to be determined

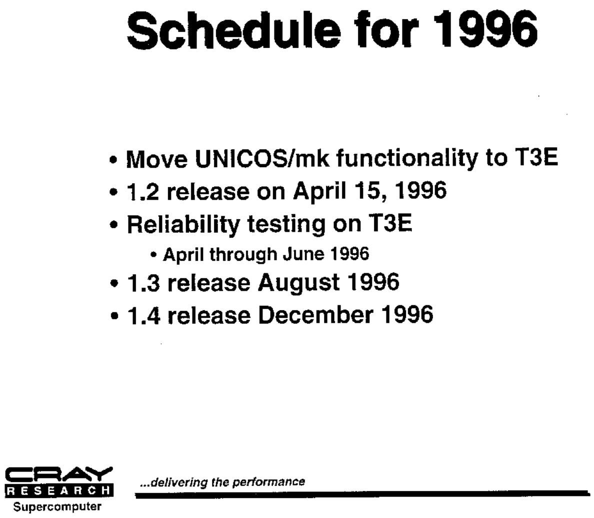
CRAY RESEARCH Supercomputer ...delivering the performance



T3E Status

- Running multi-PE multi-user on T3E TVs
- Debugging I/O
- Testing basic OS functionality

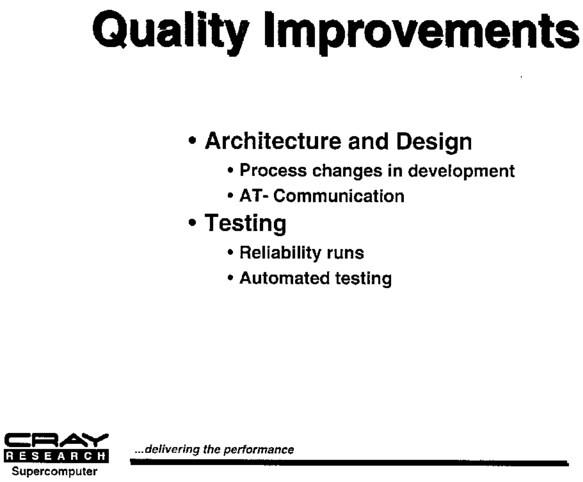
CRAY RESEARCH Supercomputer ...delivering the performance



Schedule for 1996

- Move UNICOS/mk functionality to T3E
- 1.2 release on April 15, 1996
- Reliability testing on T3E
  - April through June 1996
- 1.3 release August 1996
- 1.4 release December 1996

CRAY RESEARCH Supercomputer ...delivering the performance



Quality Improvements

- Architecture and Design
  - Process changes in development
  - AT- Communication
- Testing
  - Reliability runs
  - Automated testing

CRAY RESEARCH Supercomputer ...delivering the performance