The
Connection Machine
System

System Software Overview: Release 7.2 Beta 1

Version 7.2 Beta 1
December 1992

Thinking Machines Corporation
Cambridge, Massachusetts

First printing, December 1992

****

PRELIMINARY DOCUMENTATION

The information in this document is subject to change without notice and should not be construed as a commitment by Thinking Machines Corporation. Thinking Machines assumes no liability for errors in this document.

This document does not describe any product that is currently available from Thinking Machines Corporation, and Thinking Machines does not commit to implement the contents of this document in any product.

****

Thinking Machines Corporation
245 First Street
Cambridge, Massachusetts 02142-1264
(617) 234-1000

System Software Overview: Release 7.2 Beta 1

This overview provides a brief look at the interim version of base system software known as Release 7.2 Beta 1. This release features support for the Scalable Disk Array (SDA) and for 64-bit integers.

Fuller documentation is provided by user manuals, release notes (hardcopy and/or on-line) and on-line bug-update files. Users should consult this documentation for further details on individual products.

Release 7.2 Beta 1 contains the following components:

- CMOST Version 7.2 Beta 1
- CMRTS Version 7.2 Beta 1
- Prism Version 1.2 Beta 1
- CM Fortran Version 2.1 Beta 0
- CMMD Version 2.0 final
- *Lisp Version 7.5.1

In addition, Release 7.2 Beta 1 works with these other current or upcoming releases:

- C* 7.0 Beta with Patch 1
- CMSSL 3.1 Beta 1
- CM/AVS Version 1.0 Beta 2
- CMX11 Version 1.5 final

- CMAX Version 0.8
- pndbx Version 1.2 Beta
- CMMD 3.0 Beta
- C* 7.1 Beta

NOTE: Individual software shipments are tailored to the needs of individual sites. Therefore, not all sites will receive all the products (or versions of products) mentioned in this Overview.

1 Requirements

(1) All existing programs must be relinked to run under CMOST Version 7.2 Beta 1 and use the new Run-Time System.

(2) A site must install SunOS 4.1.2 before installing the software for Release 7.2 Beta 1. Be aware, however, that CMOST, when installed, overwrites SunOS; the CM-5 operating system is CMOST, not SunOS. Any Sun drivers, therefore, must be recompiled and relinked against CMOST kernel header files in order to be used with this release of CMOST.

2 Release 7.2 Beta 1 and the Vector Units

At Release 7.2 Beta 1, code written in CM Fortran, DPEAC, and CDPEAC utilizes the vector units. Calls from CM Fortran to the CM Fortran Utility Library, CMSSL, Version 3.1 Beta 1, CM/AVS Version 1.0 Beta 2, and CMX11 Version 1.5 final libraries also run on the vector units. The CM Run-Time System (CMRTS), the CM file systems (CMFS and SFS), and Prism all provide full support for programs that make use of the vector units.

CMMD 2.0 and C* 7.0 do not support programming on the vector units. CMMD Version 3.0 and C* Version 7.1 will provide this support.

3 CMOST Version 7.2 Beta 1

CMOST Version 7.2 Beta 1 features

- support for the Scalable Disk Array (SDA) and Scalable File System (SFS)
- support for 64-bit integers

The release also provides new diagnostic support.

CMOST Version 7.2 Beta 1 also contains a number of bug fixes. Information on these may be found in the bug update files, available on-line either directly or via Prism's **Doc** menu.

3.1 SDA Support

The Scalable Disk Array (SDA) is a collection of sets of high-speed commodity disks that reside within the CM-5 system's main storage cabinets and connect directly to the CM-5 networks. A single SDA can support multiple file systems. Data is striped over all the disks that make up a file system, transparently to the user. If an SDA is configured with a parity drive, it checks parity (and corrects single-bit errors, when necessary) on each I/O operation.

The file system for the SDA is the Scalable File System (SFS). The SFS is an enhancement of the UNIX file system, with extensions to support both parallel I/O and much larger files than most UNIX implementations can accommodate.

Each SFS is mounted on the SDA's I/O control processor (the processor that manages the SFS's files). An SFS can be mounted locally on the IOCP or can be mounted on the partition managers via NFS.

3.1.1 Restrictions

The SDA and the SFS have the following restrictions in CMOST Version 7.2 Beta 1:

- An SFS file system has a maximum size of 32 disks.
- The number of disks in an SFS file system must be a power of two.

- No more than 2 SFS file systems can be supported within one CM-5.

These restrictions should be eased in future versions of CMOST.

3.1.2 Using the SFS

Programs access the Scalable File System in any of a number of ways:

- CM Fortran utility library function calls
- CMFS commands and function calls
- CMMD I/O function calls (not available until CMMD Version 3.0)
- Standard UNIX command and function calls

  (**NOTE**: Since this is a preliminary release, the performance of UNIX commands and calls may not be as good as the performance of the other interfaces.)

From a user's perspective, the SFS file system looks like, and behaves like, any UNIX file system. For example, its files store data in canonical order, and it recognizes standard UNIX environment variables, such as PATH.

For further information on CM-5 I/O, please see the *CM-5 I/O System Programming Guide* for Version 7.2 Beta 1.

3.2 Accounting

CM-5 system accounting is an enhanced version of SunOS accounting. It can generate detailed accounting information for both CM-5 processes and UNIX-always processes and can be used both for resource monitoring and for process accounting. The accounting package can identify processes by a number of means:

- UNIX group ID and user ID
- an account ID that can be set by the environment variable `CM_ACCOUNT_ID`
- file system device number

CM-5 accounting runs on a per-partition basis. All aspects are controlled by CMOST utilities, such as the **cmacct** command, which controls all aspects of the accounting system, and the **extract** command, which creates simple accounting reports.

For full information on the CM-5 accounting, see the *CM-5 System Administrator's Guide* for Version 7.2 Beta.

3.3 Diagnostics

The beta release of the diagnostics provides the following functionality over that of the previous releases:

3.3.1 Functional Tests for the VUs

The center piece of this release of the diagnostics is the addition of functional tests for the VU. These tests are incorporated as a part of the existing PN group.

Mixed Partitions. For normal use, the CM-5 does not support partitions containing both VU and Miata chips. However, there may be occasions on which such a partition exists. (For example, one might add the SDA's SCN board to a VU partition for the sake of running certain diagnostics.)

In such cases, where there are VU and Miata within the same partition, the command **initialize-pe-memory** will print these warning messages:

```
  . Mon Nov 2 10:52:29 1992 -> initialize-pe-memory

CMPMD_initialize_memory: Mixed Memory Controller System.
  Defaulting to Serial Memory Initialization !

Initializing PE-8 as DP
Initializing PE-9 as DP
Initializing PE-10 as DP
Initializing PE-11 as DP
Initializing PE-32 as MC
```

In the above example, PN32 is the **ioba** and has Miata. All others are VU-based nodes.

After printing the messages, **ipm** then proceeds to initialize the memory accordingly. The serial initialization employed takes longer then the normal case, which uses the Control Net.

Remember: This can only happen when running diagnostics for I/O nodes, including IOBA and IOCA. Mixing VU-based and MC-based non-I/O PNs is not supported in real applications.

3.3.2 Functional Diagnostics for the SDA

The functional test group for the Scalable Disk Array has been completed. The new group is called "SDA."

In order to run the SDA there must be at least one SCN board in the partition. If **cmdiag** is to be invoked with the **-p** option then the user must first create a partition which includes at least one SCN by invoking "**cmpartition create**".

Multiple SCN boards are accepted. All functional test groups including PN and the verifiers work properly.

3.3.3 Error Signal Test

A new test for checking the error signal propagation has been added. This test must be invoked explicitly by the user.

```
test-error-signals <start-height, minimum is 1>

Alias: tes
```

This test checks to see whether the Control Net signals all errors when the error pin on each Data Network chip is driven low.

(Under normal usage, a CM-5 functions well even if error signals are not properly propagated. This can make error checking and debugging harder.)

3.3.4 Group Name Change

All JTAG test groups now have a prefix to their names. For example, the former group named CLKDN is now JTAG-CLKDN. With this change the user will be reminded of the nature of test group being run.

All connectivity tests have been moved into a new group called BSR.

3.3.5 Environmental Sensor Interface

Two new commands have been added to improve the interface to the environmental sensor:

`env-monitor` provides formatted information regarding the ambient temperature inside a cabinet and the power-fail bit status.

Usage: **`env-monitor <t | p | b>`**

`t`	for temperature reading
`p`	for power-fail status bit
`b`	for both readings

`frequency-margin` sets the **`pll`** setting on the CLKDN board.

Usage:
`frequency-margin <e | h | p | d> <pm-clk-mhz>`

`e`	- external input from backplane	
`h`	- PLL VCO / 2	32Mhz
`p`	- PLL VCO	64Mhz
`d`	- Default	32Mhz from local oscillator

`pm-clk-mhz`

range 15-50Mhz

30Mhz for low margin
32Mhz for normal setting
35Mhz for height margin

3.3.6 Forcing Autosize from cmdiag

A new switch, **-z**, has been added to **cmdiag** to allow users to force an autosize when starting **cmdiag**.

```
        cmdiag -z
```

All autosize files in **/etc/cm/diagnostics/** will be deleted. If **CMDIAG_FILES** is defined then the files from the specified directory will be deleted.

The user will not have to answer the yes/no question before the autosizing starts.

3.3.7 cmboot Prints Simpler Error Message

cmboot has been changed so that it will no longer attempt to continue the booting if the Control Net has a problem. Instead, it prints one message and stops. (Earlier versions printed many lines of error messages.)

3.3.8 Shared NI drlp Daemon and cmdiag Are Mutually Exclusive

There is a new restriction when running the DR verifier tests. No DR verifier tests should be run whenever the NI is shared by the diagnostics and the **drlp** daemon. (The **drlp** daemon becomes active after a successful invocation of **sda-cold-boot**.)

The only way to resume the running of the DR verifier is to reboot the CP to stop the **drlp** daemon.

For quick verification of a partition without going through the rebooting, the on-line test suites under **/usr/diag/tsd/** should be used after **tsd** is brought up.

If the on-line test suites should fail, it is recommended that the partition should be suspended for further OS operation. The entire machine will need to be shut down and the complete set of diagnostics run.

4 CMRTS

Version 7.2 Beta 1 of the CM Run-Time System adds support for 64-bit integers to its previous functionality. It also offers some performance speedups over previous versions.

5 Prism

Prism Version 1.2 Beta 1 is available with CMOST Version 7.2 Beta 1. This new version of Prism contains the following new features:

- C* support
- 64-bit integer support
- extensively improved performance analysis
- new structure visualizer
- split source window
- CMAX support
- enhanced support of on-line documentation
- faster loading of programs
- miscellaneous changes and enhancements

For further information, see the *Prism Release Notes* for Version 1.2 Beta. Prism release notes, as well as release notes for other products, are available on-line from within Prism; use the new **Release Notes** selection in the **Doc** menu.

6 CM Fortran

CM Fortran Version 2.1 Beta 0 provides two new features:

- SDA support
- 64-bit integers

6.1 SDA Support

The utility I/O procedures in the CM Fortran Utility Library for Version 2.1 Beta 0 now support the Scalable File System (SFS) on the Scalable Disk Array (SDA), in addition to the CM File System (CMFS) on the DataVault, VME, and CM-HIPPI.

As with other CM I/O devices, use of the SDA is transparent to the programmer calling CM Fortran I/O utilities. For systems with more than one I/O device and/or file system installed, the choice of device is governed by a site default; by a user environment variable, **CMFS_PATHTYPE**; or by the syntax of the file pathname. For more information, see the *CM-5 I/O System Programming Guide* for Version 7.2 Beta 1, the manpage for **CMFS_PATHTYPE(7)**, and the *CM Fortran Release Notes* for Version 2.1 Beta 0.

6.2 64-bit Integers

CM Fortran Version 2.1 Beta 0 provides a new data type, the 64-bit integer, on both the partition manager and the nodes. This data type is supported exclusively on CM-5 systems with vector units.

6.2.1 Declaring and Defining 64-bit Variables and Constants

To declare a 64-bit integer variable, use either:

the CM Fortran language extension indicating number of bytes:

```
INTEGER*8  VAR_NAME
```

or the Fortran 90 **KIND** mechanism:

```
INTEGER(KIND=64) VAR_NAME
```

CM Fortran also defines symbolic constants that evaluate to **KIND** numbers for integers and for all floating-point types. (The **KIND** mechanisms cannot be used with logical and character types.) The constants for the integer types are:

```
INTEGER*4  _SINGLE_INT_  (value=32)
INTEGER*8  _DOUBLE_INT_  (value=64)
```

A similar mechanism enables you to specify Fortran 90 typed constants. Append to a literal constant an underscore followed by the value or a symbolic name:

```
A = 1234_64 + 5678__DOUBLE_INT_
```

6.2.2 64-bit Integers and CM Fortran Intrinsic Functions

Arrays of 64-bit integers can be used in any expression or as arguments to CM Fortran intrinsic functions. The bit-level inquiry functions **POPCNT**, **POPPAR**, and **LEADZ** behave as follows:

- **POPCNT** and **POPPAR** always return **INTEGER*4**, for any argument of type **INTEGER*4**, **INTEGER*8**, or **LOGICAL**.
- **LEADZ** returns **INTEGER*4** for **INTEGER*4** and **LOGICAL** arguments, but returns **INTEGER*8** for argument of type **INTEGER*8**.

6.2.3 64-bit Integers and CM Fortran Utility Functions

Arrays of 64-bit integers can be used as arguments to the CM Fortran Utility Library procedures. However, 64-bit integers should not be used for integer scalar arguments to these procedures.

The Utility Library now provides a new random number procedure for use with 64-bit integers: `CMF_RANDOM_LONG_S_INTEGER`. This procedure takes only an `INTEGER*8` array argument; it also takes an `INTEGER*8 LIMIT` argument, specifying the upper bound.

See the *CM Fortran Release Notes* for Version 2.1 Beta 0 for more information on 64-bit integers, including a few additional restrictions.

7 CMMD Version 2.0 Final

CMMD Version 2.0 final differs from CMMD Version 2.0 Beta only in that assorted errors in the software and the documentation have been corrected. New editions of the *CMMD Reference Manual* and the *CMMD User's Guide* accompany the release.

See the CMMD on-line release notes and bug-update file for summary information on CMMD Version 2.0, and for information on its fixed and outstanding errors.

8 *Lisp Version 7.5.1

This version of *Lisp provides a working interface to CMX11,Version 1.5 final.