The
Connection Machine
System

CM/AVS Release Notes

Version 1.0

February 1993

Thinking Machines Corporation
Cambridge, Massachusetts

First printing, February 1993

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

Contents

Customer Support

Thinking Machines Customer Support encourages customers to report errors in Connection Machine operation and to suggest improvements in our products.

When reporting an error, please provide as much information as possible to help us identify and correct the problem. A code example that failed to execute, a session transcript, the record of a backtrace, or other such information can greatly reduce the time it takes Thinking Machines to respond to the report.

If your site has an applications engineer or a local site coordinator, please contact that person directly for support. Otherwise, please contact Thinking Machines' home office customer support staff:

Internet Electronic Mail:	`customer-support@think.com`
uucp Electronic Mail:	`ames!think!customer-support`
U.S. Mail:	Thinking Machines Corporation Customer Support 245 First Street Cambridge, Massachusetts 02142-1264
Telephone:	(617) 234-4000

CM/AVS Release Notes
Version 1.0

1 About CM/AVS Version 1.0

1.1 Overview

CM/AVS adapts and extends the Application Visualization System (AVS) to the realm of the CM-5. AVS provides a graphic programming environment in which a user builds a distributed visualization application. An application may involve diverse operations such as filtering, graphing, volume rendering, polygon rendering, image processing, and animation. CM/AVS enables an application to operate on data that is distributed on CM-5 processing nodes and to interoperate with data from other sources. CM/AVS also facilitates the incorporation of CM-5 code into a CM/AVS application.

CM/AVS is not run separately from AVS. A user runs AVS normally, using CM/AVS modules and functions to handle data on the CM-5.

1.2 Hardware Requirements

CM/AVS requires a Connection Machine CM-5.

CM/AVS has no display station or memory requirements beyond those of the AVS product. AVS requires an 8-bit color monitor as a minimum.

1.3 Software Requirements

CM/AVS Version 1.0 requires CMOST and a C* or CM Fortran compiler (or both) to be installed and running on the CM-5. The compatible CMOST and CM Fortran releases are:

- CMOST Version 7.2S2 and CM Fortran Version 2.0
- CMOST Version 7.2 Beta1 and CM Fortran Version 2.1

The following C* compiler is compatible with CM/AVS Version 1.0:

- C* Version 7.1 or later

CM/AVS Version 1.0 requires the following software to be installed and running on the workstation (or on the partition manager, if you intend to run AVS there):

- AVS (license required); CM/AVS is compatible with Version 4.0 or 5.0.
- An X server (AVS requirement)

1.4 Installation

CM/AVS is installed on the CM-5 compile server as follows:

CM/AVS libraries	`/usr/lib`
CM/AVS include files	`/usr/include`
Combined module binary, list-dir file, and library description file	`/usr/lib/cmavs_library`
Examples	`/usr/examples/cmavs`
Module help files	`/usr/doc/cmavs/modules`
Release notes	`/usr/doc/cmavs-1.0.releasenotes`
Source (by license only)	`/usr/src/cmavs`

The directory `/usr/examples/cmavs/unsupported` contains items that are available without guarantee or support. The `README` file in this directory con-

tains information about its contents. Appendix C in the *CM/AVS User's Guide* contains additional information about unsupported programs and modules.

1.5 Documentation for Version 1.0

The following Connection Machine documents are released with Version 1.0:

- *CM/AVS User's Guide* Version 1.0
- *CM/AVS Release Notes,* Version 1.0. The on-line copy is installed in `/usr/doc/cmavs-1.0.releasenotes`. If you do not find the file in this location, contact your system administrator.
- On-line module descriptions, viewable through the AVS module documentation viewer.

Users should have the AVS document set.

2 The CM/AVS Package

The CM/AVS software package includes libraries of modules, a set of routines that provide general operations for parallel fields, some on-line code examples, the help files, and release notes.

The building blocks of an AVS application program are small, packaged units of code, called *modules.* Most modules process typed data input(s) into typed data output(s). Each module performs a given function. The function may be as simple as adding two arrays, or as complicated as rendering the isosurfaces of a volume. When a CM/AVS module is used, the function is performed on a CM-5.

Modules are connected to form larger applications, called *networks.* In a network, information is passed between the modules as various data types. Only the *field* data type, which represents an array of data, is relevant to CM/AVS. CM/AVS supports a *parallel field* that accommodates the distribution of data across the CM-5 processing nodes. CM/AVS includes routines to allocate the parallel arrays,

and to access the data and coordinates as CM Fortran arrays or C* parallel variables.

When CM/AVS modules that operate on parallel data are connected with AVS modules that operate on serial data, CM/AVS routines convert the data between parallel and serial fields as required. The conversion is transparent to the user and to the module writer. CM/AVS routines also provide general operations for parallel fields.

2.1 The CM/AVS Modules

CM/AVS provides the following modules:

```
antialias cm
clamp cm
color range cm
colorizer cm
combine scalars cm
compare field cm
compute gradient cm
contrast cm
downsize cm
extract scalar cm
fft cm
field math cm
field to byte cm
field to double cm
field to float cm
field to int cm
luminance cm
orthogonal slicer cm
threshold cm
```

The following modules are included without guarantee or support:

```
field to polygons
field to spheres
```

2.2 The CM/AVS Functions

CM/AVS provides the following functions:

```
CMAVScorout_init
CMAVSdata_alloc
CMAVSfield_alloc
CMAVSfield_alloc_data_shape
CMAVSfield_alloc_points_shape
CMAVSfield_copy_points
CMAVSfield_data_get
CMAVSfield_points_get
CMAVSfield_reset_minmax
CMAVSis_field_on_CM
```

2.3 Unsupported Programs

The following programs are included without guarantee or support:

```
avstoppm
ppmtoavs
```

3 Changes from the Previous Release

The following are changes between the Beta release and Version 1.0:

- It is no longer necessary to link with the `cmvio` library.
- There is some C* support for vector unit modules (see Section 4).
- The speed of serial-to-parallel and parallel-to-serial transfers is improved.
- The following function names have changed:

FROM	TO
`CMAVSdata_alloc_parallel`	`CMAVSdata_alloc`
`CMAVSfield_alloc_parallel`	`CMAVSfield_alloc`
`CMAVSfield_allocate_data_shape`	`CMAVSfield_alloc_data_shape`
`CMAVSfield_allocate_points_shape`	`CMAVSfield_alloc_points_shape`

  Using the original names will still work, although a run-time warning message will flag the first use of the original name.

4 Limitations

These are known limitations at CM/AVS Version 1.0:

- C* support for vector unit modules is limited as follows:

  Under CMOST Version 7.2S2, there is no C* support for CM/AVS vector unit modules.

  Under CMOST Version 7.2 Beta1, Patch 3 is required, and reading byte and short fields into C* vector unit modules is not guaranteed.

- The CM/AVS subroutine modules will not work with Prism profiling.

Please observe the following constraint at this release:

- Do not use the `-noaxisreorder` switch when you compile with CM Fortran Version 2.1.

5 Outstanding and Fixed Bugs

Thinking Machines Corporation's Customer Support Group supplies an on-line bug update file that lists outstanding and fixed bugs. The default pathname for this file is `/usr/doc/cmavs-1.0.bugupdate`. If you do not find the file in this location, contact your system administrator. If you have questions about the report, contact Customer Support at (617) 234-4000.