



**Centrum voor Wiskunde en Informatica**  
Centre for Mathematics and Computer Science

---

D.T. Winter

Information about CWI Ada facilities #1

Department of Numerical Mathematics

Note NM-N8501

September

---

## Information about CWI Ada facilities #1

This is the first of an irregular series of papers about the CWI Ada facilities. It will provide information and hints about the system.

D.T. Winter

*Centre for Mathematics and Computer Science, Amsterdam, The Netherlands.*

### Equipment.

The system consists of a Data General MV4000 mini computer with 4 Mbyte memory, a 354 Mbyte disk and a 1600 bpi tape drive. Note: the largest tape reel acceptable for the tape drive is a 1200 feet reel. The system has one out-going and seven in-going ports. In the (near) future it will be connected to Ethernet to provide for printing facilities etc. The operating system running is AOS/VS (Advanced Operating System / Virtual System) with MV/UX (a Unix kclone) on top. The current version of AOS/VS is 5.00. Because of system restrictions the in-going ports are set at a fixed speed; there is no auto baud recognition.

### MV/UX

MV/UX is a V7, system III, system V and BSD hybrid. We have currently version 2.01. It has something from everything, and some features are not present at all.

One of the features non-present is 'nohup'. We may provide for something like that later (probably in the form of the ability to batch a job from MV/UX).

Because of the differences between the ACL (Access Control Lists) of AOS/VS and the permission bits of MV/UX, 'ls -l' will not always reflect the true permissions. Use of ACL from MV/UX is possible ('cli acl ...') but not recommended (you are able to give files away you are not able to retrieve again!).

## ADE

The Ada system is entered via ADE (Ada Development Environment). The current revision level of this system is 2.30. We have made local provisions to make ADE directly accessible from MV/UX (although we may still have omitted some parts). So the ADE manual is roughly valid in the MV/UX environment too.

From CLI you enter ADE with the ENTER macro without or with /OLD or /NEW switches; /OLD accesses the old ADE while /NEW accesses the future ADE. Without switch the current ADE is accessed. (Currently only one ADE is available; see however the Problem Section.)

There is no equivalent to the ENTER macro in MV/UX, so it is not used. Use the commands directly. To access older versions of the ADE append the -o flag; to access future versions append the -n flag. One may also use the -r flag followed by revision number (e.g. -r 230). An alternative to the -o, -n and -r flags is to set the shell variable ADE\_REV. Example:

```
ADE_REV=220;export ADE_REV
```

From that point onwards revision 2.20 of ADE is used. Still another alternative is to create in your directory a file with the name '.revision' containing the revision number.

When a command looks for the version required this is done in the following order:

1. Look at a file with the name '.revision'.
2. Look for the variable ADE\_REV. This overrules the revision from 1.
3. Look at -o, -n or -r flags. This overrules 1 and 2.

The MV/UX commands reside in /usr/ada/bin. On login this will be part of your PATH. Remember to name it when changing your PATH.

When using ADE from MV/UX one is warned not to use unix pathnames for files but AOS pathnames (replace / by :, etc.). There is an utility that performs this: 'aos\_form'. (It is possible that we will correct this later).

General syntax to access ADE from MV/UX:

```
ade_command [-o] [-n] [-r revision] [- switches] parameters
```

Here -o, -n and -r are mutually exclusive and explained above. -o, -n and -r override the setting of ADA\_REV and contents of '.revision'. The [- switches] part is used as follows. To get the MV/UX equivalent of:

```
ADALINK/VERBOSE MAIN
```

enter:

```
adalink - verbose main
```

To give multiple switches there are two possibilities; the first is to separate them by /:

```
ade_command - switch1/switch2
```

the second is to give multiple - parameters:

```
ade_command - switch1 - switch2
```

The two may be combined:

```
ade_command - switch1 - switch2/switch3
```

passes three switches.

MV/UX and CLI.

It is possible to switch between MV/UX and CLI (the AOS/VS command language interpreter). To switch from MV/UX to CLI use the command 'cli'. To go the other way give the command 'X :BIN:SH'. To use the Unix editor vi from CLI give the command ':BIN:VI'. (Note that CLI does not have TERM and TERMCAP information, so the settings will be all wrong; it is better to create your own interface.)

Your first steps in the system.

When you first enter the system, give the command

```
PROJCREATE
```

this will start your ada program library tree. Next enter:

```
LIBCREATE
```

which creates the program library that belongs to the current directory.

Every time you need a new directory use the command

```
DIRCREATE
```

and not CREATE or MKDIR. Enter the directory and give the command

```
LIBCREATE
```

(again for a program library).

Use CREATE and MKDIR only for directories where you are not going to create Ada programs.

Local additions.

For your convenience we added the following standard instantiations to the system:

```
package BOOLEAN_IO is new
    TEXT_IO.ENUMERATION_IO(BOOLEAN);
package INTEGER_IO is new
    TEXT_IO.INTEGER_IO(INTEGER);
package FLOAT_IO is new
    TEXT_IO.FLOAT_IO(FLOAT);
package LONG_FLOAT_IO is new
    TEXT_IO.FLOAT_IO(LONG_FLOAT);
(this last one only for revision 2.30).
```

To access them use the command:

```
LIBSEARCH/APPEND [!PATHNAME ADE_ROOT?]:LIBRARIES:LOCAL
```

from CLI and from MV/UX:

```
libsearch - append [!pathname ade_root?]:libraries:local
```

and add appropriate with clauses to your program.

The use of these packages will reduce the time and memory requirements for compilation.

## Hints.

With revision 2.30 the source level debugger is working. Do not forget to link with the /DEBUG switch (or - debug flag in MV/UX) with the ADALINK command. Also you should compile with this switch the routines you want to debug.

For efficiency of computer resources (and also for maintenance) it is better to put every compilation in a separate file. Also for compilations that are procedures you should split up specification and declaration and put them in separate files. It is also better to invoke the compiler with one source file only. (We experienced a storage overflow when compiling a number of compilations in one batch, while this did not occur when we compiled them separate.)

Always put your main program in a separate file, with the same name (except for the .ADA extension). Also follow it with: pragma MAIN. It seems to be possible to do it otherwise, but we did not yet find the correct way.

## Logging onto the system.

Configure your terminal such that it will send even parity.

## Phone numbers:

300 Baud +31 20 5611444

1200 Baud +31 20 5611222

If the connection is established (you hear a high pitched beep), make your modem connections and enter CR (Return). You will see:

\*\*\* SARA NETWORK \*\*\*

DESTINATION?

Enter now 'cwi' and you will see:

PAUSE

followed after some time by:

CONNECTED TO ...

and again sometime later:

CWI Portselector

Class name:

Now you enter 'ar' and the system responds with:

CWI Unix Network

GO

followed by (if everything is well) the first message from AOS/VS.

\*\*\*\* AOS/VS REV 5.00.00.00 /

Press NEW-LINE to begin logging on \*\*\*\*

Ignore the text and enter carriage return. You will see something like:

AOS/VS 5.00.00.00 / EXEC 5.05.00.00

14-Sep-85 0:09:17 @CON8

Username:

Now start your login sequence.

Connections through X.25 are available. They are intended in the first place for international logins and data transmission. Contact us for information.

CWI

We explain our status here.

We are not a computer center or service institute. Hence we are not able to do a lot about information and service (problem solving and the lot). However we are willing to help on an incidental basis. The contact is currently:

dik winter

UNIX E-mail: dik@mcvax.UUCP  
(older systems): mcvax!dik  
ARPA E-mail: "decvax!mcvax!dik"@Berkeley

ptt:  
Centrum voor Wiskunde en Informatica  
Afdeling NW  
Kruislaan 413  
1098 SJ Amsterdam

telephone:  
+31 20 592 4104  
+31 20 592 4146  
(From 11.00-12.30 and 14.00-18.00 dutch time only.)

### Problem section.

In this section we give the problems encountered so far. This list will be expanded in future issues. Hopefully some problems will be solved too. If some form of corrective action is available, this is indicated by the word 'patched'. If the bug is locally solved this is indicated by 'solved' (this does not mean that new releases of the system will have the problem solved). Up to date lists may be found in the directory :UDD:BUGS (cli) or /udd/bugs (MV/UX). Submitting bug reports should also be done in that form; use /udd/bugs/prototype, and mail to dik (mail dik <file>). The lists are given in four sections.

#### AOS/VS

1. If the operator logs off or if the console gets disconnected or breaks down the system halts (sometimes ungracefully).
2. The speeds for in-going ports are baked in the system. There is no way to do auto baud recognition.
3. Many permissions are wrong. Many have also been corrected already, but some wrong settings are still left.

#### Terminal interface

1. The system thinks that all terminals are Data General terminals. This means that it is (probably) impossible to set the 'kill-last-character' character to anything other than rubout (DEL) rather than the more widely used CTRL-H (Backspace). Also the 'kill-line' character is CTRL-U.
2. We are not yet clear about the best terminal settings. Perhaps they are not available.

#### MV/UX

1. Redirection of standard output to a file that already exists bombs your shell. You are left with a 1 to 2 Mbyte memory dump (no two have the same name) and if it is your login shell you are logged off the system. Please, refrain from this.
2. Due to some brain-damage somewhere 'vi' is unable or unwilling to output CTRL-H (backspace). So even if your terminal is able to backspace, and your termcap entry says so, this will not be done, your screen will in those cases not exhibit what vi thinks it does. Also newline from vi is output as linefeed only.  
Patched: remove :bs: from your termcap entry and enter :nl=^m^j:, and :bc='your cursor left key':. It won't help on terminals where cursor left is CTRL-H.
3. Contrary to what the manual says, the shell does not warn you on login if you have mail.  
Patched in /etc/profile.
4. More dumps core. Do not use it. We introduced the program lmore, it stops after every 24 lines and should be restarted with CTRL-Q. (This program is not usable when logged in through remote from the Vaxen.)

5. Vi should set up the number of lines in its initial display dependent upon the speed of the line. Instead it uses the information whether it is a modem connected line or not (and all lines at CWI must be given as modem connected).

#### ADE

1. Contrary to popular believe the command LIBCREATE can not be abbreviated.  
Patched.
2. If you declare a routine for which one or more parameters have defaults, and if you split specification and declaration, the compiler refuses to compile, saying that the specifications do not match.

Example: the following:

```
procedure X(Y: FLOAT:= 1.0);
procedure X(Y: FLOAT:= 1.0) is
  ...
```

is not accepted.

This has been solved in previous levels.

Patched: revision 2.20 is (unofficially) installed. If you need it use the /OLD switch with CLI's ENTER or the -o flag on MV/UX commands or define ADA\_REV=220;export ADA\_REV, or create a file '.revision' with the text 220. Note: use this only if you desperately need it, there is no maintenance. Further note: you should use the old version for all commands in your current directory (in MV/UX this is taken care of if you use the file '.revision').

3. An exception that propagates through the main program is not indicated. That is, the program stops but no reason is given. It appears only when your environment is MV/UX; the shell does not catch the error return (and in contradiction to the manual \$? is not set). There is a work around, always use the ADE package CURRENT\_EXCEPTION in your main program as follows (we first give the program listing and after that the result of execution):

```
$ cat testing.ada
procedure RAISE_IT is
  ANONYMOUS: exception;
begin
  raise ANONYMOUS;
end;
with RAISE_IT;
with CURRENT_EXCEPTION, TEXT_IO;
use TEXT_IO;
procedure TESTING is
begin
  RAISE_IT;
exception
  when others =>
    NEW LINE;
    PUT("Exception ");
    PUT(CURRENT_EXCEPTION.NAME);
    PUT(" found.");
```



```

NEW_LINE;
end;
pragma MAIN;
$ testing

```

Exception ANONYMOUS found.

\$

As may be seen this even works for exceptions out of scope.

There is another solution, use the (newly installed) MV/UX command XEQ when executing an Ada program: XEQ PROG, however, in that case you will not obtain the name of the exception.

4. The following attributes of floating-point types deliver wrong values: SAFE\_SMALL, SAFE\_LARGE, SAFE\_EMAX, MACHINE\_RADIX, MACHINE\_MANTISSA, MACHINE\_EMAX, MACHINE\_EMIN; correct values are:
 

SAFE_SMALL	=	~ 6.91E-76
SAFE_LARGE	=	~ 7.24E+75
SAFE_EMAX	=	252
MACHINE_RADIX	=	16
MACHINE_MANTISSA	=	6 (FLOAT) or 14 (LONG_FLOAT)
MACHINE_EMAX	=	63
MACHINE_EMIN	=	- 64
5. An attempt to write a number in absolute value less than about 1.5E-70 and not equal to 0.0 results in the raising of NUMERIC\_ERROR in PUT.
6. When you make use of the Ade standard input file, the contents are always copied (while reading) to the standard output file. There is no harm if they are both your terminal. However, this occurs also if you have redirected standard input (PROC/INPUT=... in AOS/VS and <... in MV/UX).

CWI

1. We refuse to make filesystem dumps; you are on your own.
2. We regard this system as an unsafe system, we are not going to do anything about security. So don't leave critical files around.