3

Installation

Être ce que nous sommes et devenir ce que nous
sommes capables de devenir, tel est le seul but de la vie.

Robert Louis Balfour Stevenson.

This chapter is devoted to the installation of ACE. Before starting the installation, we should be aware of the following remark: even if ACE is available for any UNIX, MS-DOS or MACINTOSH system, for any release 1, 2, 3 or 4 of *Maple V*, we get different specific versions of the environment. That is to say, we get the UNIX version for *Maple V* release 2 which is not exactly the same as the UNIX version for *Maple V* release 3, and which is quite different from any MS-DOS version. We cannot say that sources differ between the UNIX, MS-DOS and MACINTOSH versions, but some of them have special instructions for instance for the releases 3 and 4 of *Maple V*. In any case, there are some specific programs (given only with UNIX versions) to compile the environment on a UNIX system, whereas there are other programs (given only with MS-DOS versions of ACE) for the installation on a MS-DOS system.

Remember also that ACE is available on a CD-ROM which provides several pre-installed versions of ACE for different systems. This can be useful to quickly install ACE on your system.

Before installing the environment, we should know the version which corresponds to our system (UNIX, MS-DOS or MACINTOSH) and its *Maple V* release (1, 2, 3, 4), and then refer to the corresponding section. The ACE environment is distributed (in its non-preinstalled version) for each UNIX or MS-DOS version, as one single archive file which contains sources and includes pre-computed data. In the case of the MACINTOSH versions, the archive file is different for some technical reasons.

3.1 Installing ACE on a UNIX system

This section is devoted to the installation of ACE in a UNIX environment.

The ACE environment is distributed as a compressed archive of files. First of all, we have to restore these files. According to the *Maple V* release ACE has to be installed for, we have the corresponding compressed archive file, `ACE1UNIX.tgz` (*Maple V* release 1), `ACE2UNIX.tgz` (*Maple V* release 2), `ACE3UNIX.tgz` (*Maple V* release 3) or `ACE4UNIX.tgz` (*Maple V* release 4). These files contain the sources and on-line help files of ACE together with the installation programs.

In the following, we will suppose that we get the ACE3UNIX.tgz, or in other words, we have to install the ACE environment on a UNIX operating system for *Maple V* release 3. Anyway, the same work has to be done for the other archive files.

3.1.1 Restoring sources

The ACE3UNIX.tgz file is a "gziped" file, that is to say, it has been compressed by the UNIX program gzip. At first, we have to uncompress this file by the following command:

```
gunzip ACE3UNIX.tgz
```

Note that the gunzip command can be replaced by gzip -d. This command creates a new file named ACE3UNIX.tar while the ACE3UNIX.tgz has disappeared.

This file has been created by using the tar command. To see the detail on the files contained in the archive, we run the UNIX command:

```
tar tvf ACE3UNIX.tar
```

and to restore the ACE sources, we run the following UNIX command:

```
tar xvf ACE3UNIX.tar
```

that creates a directory ACE_PACK in which all files will be extracted from the archive file.

3.1.2 Description of the ACE_PACK directory

This directory contains the sources of the ACE environment.

The BIN subdirectory contains the .modelinit file that is a model of the .mapleinit *Maple* file that we should modify before installing ACE. It also contains the installation program, *i.e.* the INSTALL.ACE file (which is a C-shell program), the INSTALL.SR1, INSTALL.SR2 and INSTALL.HL1, INSTALL.HL2 files that describe how to respectively install source files and help files. We also give the INSTALL.DAT file which describes how to install pre-computed data. Finally, we provide in the BIN subdirectory the ACE program (that is a shell script) which can be used to run *Maple* and to automatically load ACE.

The CPRIGHTS subdirectory contains the rights file that gives the copyright notice.

The PACKAGES subdirectory contains the declarations of all packages included in ACE, whereas the SOURCES subdirectory contains the sources of all functions, together with some files devoted to the installation. Moreover, the data subdirectories of each directory of the SOURCES directory contain the corresponding pre-computed tables of data.

3.1.3 Description of a typical package

Let us choose the FM package as a typical one to present the organization of any package on the disk. It is essentially stored in the FM directory of respectively the SOURCES directory for the *Maple* sources, installation files and data, and the `PACKAGES` subdirectory for the *Maple* declaration of the package.

The `SOURCES/FM` directory contains the `src` directory in which the *Maple* code of all functions of `FM` is, the `help` directory that clusters the *Maple* help files for each function of the package. Moreover, the `files` directory is used by the installation programs like `COMPILE.FM`. Finally, the `data` subdirectory contains the pre-computed tables of data for the `FM` package and the `test` directory contains a file `FM.tst` to test the installation (`read ‘FM.tst‘;`).

The `PACKAGES/FM` directory contains three files: `FM` that declares the package, `FMext` which declares all functions used in the package and finally, `FMhlp` that declares to *Maple* where to find the compiled help files for the `FM` package.

3.1.4 Configuration file

Having restored all the environment on the disk, we are now able to make the installation.

First of all, we have to tell *Maple* where to store the compiled files (*i.e.* the `.m` files). This is done in the `.mapleinit` file in the `HOME` directory, by the use of a variable per package.

Since it is a non-trivial work to set all the variables, `ACELib, SGLib, SPLib, TABLib` and so on, this procedure has been integrated into the installation process so that we don't need to do it ourself. In other words, the installation program now asks whether we want it to modify or create our `.mapleinit` file.

However, if we still want to do it by hand, then here are the different steps to follow. We provide a model of the `.mapleinit` file, so-called `.modelinit` in the `BIN` directory. Here is this file for *Maple V* release 3:

```
##########
# File @(#) .modelinit (Maple) Sun Jan  1 17:01:45 Frh 1995
##########
#
############################################################
#                                                          #
# Insert the following lines in the $HOME/.mapleinit file. #
#                                                          #
############################################################
#
## ACE package...
#
####################################################################
#                                                                  #
# Modify the following line so that the ACELib variable  contains  #
# the directory where one wants maple to store the compiled files. #
#                                                                  #
####################################################################
```

```
#
ACELib:=`/people/roger/maple/lib`:
#
ACELibSeq:=SGLib, SPLib, SYMFLib, TABLib, TYPLib, HEKALib,  ERRLib,
           CLGLib, TEXLib, SGALib, NCALib, IDCALib, NCSFLib, FMLib,
           BNLib, BNALib, CGLib, COMPLib, PARTLib, FREELib, SFALib:
#
BNLib  :=``.ACELib.`/ACE`:              BNALib :=``.ACELib.`/ACE`:
CLGLib :=``.ACELib.`/ACE`:              CGLib  :=``.ACELib.`/ACE`:
COMPLib:=``.ACELib.`/ACE`:              ERRLib :=``.ACELib.`/ACE`:
FMLib  :=``.ACELib.`/ACE`:              FREELib:=``.ACELib.`/ACE`:
HEKALib:=``.ACELib.`/ACE`:              IDCALib:=``.ACELib.`/ACE`:
NCALib :=``.ACELib.`/ACE`:              NCSFLib:=``.ACELib.`/ACE`:
PARTLib:=``.ACELib.`/ACE`:              SFALib :=``.ACELib.`/ACE`:
SGALib :=``.ACELib.`/ACE`:              SGLib  :=``.ACELib.`/ACE`:
SPLib  :=``.ACELib.`/ACE`:              SYMFLib:=``.ACELib.`/ACE`:
TABLib :=``.ACELib.`/ACE`:              TEXLib :=``.ACELib.`/ACE`:
TYPLib :=``.ACELib.`/ACE`:
#
## Where to find the packages...for Maple V Release 1
#
# _liblist:=[ACELib, ACELibSeq]:
#
## Where to find the packages...for Maple V Release > 1
#
  libname:=ACELib, ACELibSeq, libname:
```

Thus, to tell *Maple* where to compile (and later, where to find) the packages, we copy the `.modelinit` file into the existing `.mapleinit` file in the `HOME` directory. If we don't have any `.mapleinit` file, then we create it by directly copying the `.modelinit` file into a new `.mapleinit` file. Then, we modify the `ACELib` variable in the `.mapleinit` file, setting it to an **existing** directory. Note that, according to the given `.modelinit` file, all packages will be compiled into subdirectories of the `ACELib` directory but even if it is not necessary, it is easier to manage.

Global installation

Having modified the `.mapleinit` file, or having decided that this file will be automatically modified by the installation program, we are ready to install all packages by using the `INSTALL.ACE` program in the `BIN` directory. Being in this directory, enter the UNIX command, `csh INSTALL.ACE` (or simply `INSTALL.ACE`) which starts the installation:

```
INSTALL.ACE
```

The program asks several questions, in particular it asks whether we want to modify (update) our `.mapleinit` file. It also automatically creates a `ACE` file which can be used to run `maple` and load the `ACE` environment.

Individual installation

We also provide the possibility to compile each package individually. For this purpose, we give for each package its own installation program in its corresponding directory. For instance, we can compile the `CLG` package by running a UNIX command that looks like:

```
csh COMPILE.CLG /people/roger/maple/lib/ACE
```

If we enter the UNIX command `csh COMPILE.CLG`, we get more information and for instance to see that we can compile one file among all by running a command like:

```
csh COMPILE.CLG /people/roger/maple/lib/ACE src/Toe
```

3.1.5 Having some troubles

The automatic installation program asks for the UNIX command to run to load *Maple*, so that now, we should not have any problem if *Maple* is not in the `PATH` variable. However, if there is a problem, that is to say, if the *Maple* command is unknown for the system, please follow the next advices. Since we run the `csh` shell to interpret the installation program, we should have the *Maple* program in the `PATH` variable while running this shell so that this trouble could be due to an incomplete `.cshrc` file (which is the configuration file of the `csh` shell program) in the `HOME` directory. Please refer to a documentation about `csh` to tell it where to find the *Maple* program: for instance, enter the UNIX command `man csh` that should give some hints about the problem.

Whenever there is a `writting error` message from *Maple* that looks like:

```
Error, could not open ... for writing
```

be sure that the corresponding directory exists. If it does not, be sure of having correctly set the corresponding variable in the `.mapleinit` file.

3.1.6 Using the environment

If you want someone to be able to use `ACE`, the first thing to do is to modify its `.mapleinit` to tell *Maple* where to find packages. For that, you need to copy the `.mapleinit` used to install `ACE`. It should contain the definitions of all `Lib` variables needed to use `ACE`. Note that the advantage of `ACE` is that even if the `ACE` compiled files have been moved in another directory (different from the one used for the installation), we can still use `ACE` by simply modifying the `.mapleinit` file without compiling again `ACE`.

Now, several possibilities are available. First of all, we can run *Maple* by the (usual) UNIX command `maple`, and then to load the complete `ACE` environment, that is to load all packages, we enter the *Maple* command `with(ACE);`. We can load a single package, say `SYMF`, by the *Maple* command `with(SYMF);`.

Another way to use the ACE environment is to execute the ACE UNIX command located in the BIN directory. It runs *Maple*, automatically prints the ACE banner and loads the ACE environment. When running ACE with the -x option, it runs *XMaple* instead of *Maple*. We can add the BIN directory to our PATH variable to have the ACE command accessible from anywhere. It can be done for instance in the corresponding shell configuration file. We also have to put the correct rights on the ACE file, with a command like: `chmod 755 ACE` if bad rights forbid the program to be executed. Normally, the automatic installation program must have correctly set the rights on this file.

3.2 Installing ACE on a MS-DOS system

This section deals with the installation of ACE in a MS-DOS environment. First, we have to extract files from the compressed archive files. According to the *Maple V* release we want ACE to be installed for, we get the corresponding compressed archive file, ACE1DOS.tgz (*Maple V* release 1), ACE2DOS.tgz (*Maple V* release 2), ACE3DOS.tgz (*Maple V* release 3) or ACE4DOS.tgz (*Maple V* release 4).

In the following, we will suppose that we get the ACE3DOS.tgz. ACE environment on a MS-DOS operating system for *Maple V* release 3. In any case, the same work has to be done for other releases, except for *Maple V* release 4 for which we don't need to modify any MAPLE.INI file.

3.2.1 Restoring sources

The ACE3DOS.tgz is a "gziped" file, that is to say, it has been compressed by the UNIX program gzip. We assume that this file is stored on a floppy disk named a: in the following.

To simplify the restoring phase on a MS-DOS system, we provide a program that can be used to uncompress the archive file and extract from it, in the same time, all files: this is the tar.exe file given on the floppy disk which is a "special" version of a UNIX compatible tar program including the gunzip program.

At first, being on the hard disk in the right directory, and assuming that we get this "special" tar program, we can uncompress the archive and extract all files by the following command:

```
a:tar.exe .xvf a:ACE3DOS.tgz
```

that creates a directory ACE_PACK. If we do not have this tar.exe program, files can be restored by first uncompressing the archive file and then extracting them from it (refer to the UNIX installation section).

3.2.2 Description of the ACE_PACK directory

This directory contains all sources of the ACE environment.

The BIN subdirectory contains the MAPLE.MOD file which is a model of the MAPLE.INI *Maple* file that we should modify to install ACE. It also contains the installation program, *i.e.* the INSTALL.BAT file (that is a batch file), the INSTALL.SR1, INSTALL.SR2 files and INSTALL.HL1, INSTALL.HL2 files which describe how to respectively install source files and help files. We also give the INSTALL.DAT file which describes how to install pre-computed data.

The CPRIGHTS subdirectory contains the rights file that gives the copyright notice.

The PACKAGES subdirectory contains the declarations of all packages included in ACE, whereas the SOURCES subdirectory contains the sources of all functions, together with some files devoted to the installation. Moreover, the `data` subdirectories of each directory of the `SOURCES` directory contain the corresponding pre-computed tables of data.

3.2.3 Description of a typical package

Let us choose the `FM` package as a typical one to present the organization of any package on the disk. It is essentially stored in the `FM` directory of respectively the `SOURCES` directory for the *Maple* sources, installation files and data, and the `PACKAGES` subdirectory for the *Maple* declaration of the package.

The `SOURCES/FM` directory contains the `src` directory in which the *Maple* code of all functions of `FM` is, the `help` directory that clusters the *Maple* help files for each function of the package. Moreover, the `files` directory is used by the installation programs like `COMPFM.BAT`. The `FM.ini` file gives the lines that must be added to the `MAPLE.INI` file in the *Maple* library directory(1). Finally, the `data` subdirectory contains the pre-computed tables of data for the `FM` package and the `test` directory contains a file `FM.tst` to test the installation (`read 'FM.tst';`).

The `PACKAGES/FM` directory contains three files: `FM` that declares the package, `FMEXT` which declares all functions used in the package and finally, `FMHLP` that declares to *Maple* where to find the compiled help files for the `FM` package.

3.2.4 Configuration file

Having restored all the environment on the disk, we are now able to make the installation.

If we use *Maple V* release 4, we can skip this section and continue at the next one. In other words, we don't need to modify any file before installing `ACE`.

First of all, we have to tell *Maple* where to store the compiled files (*i.e.* the `.m` files). This is done in the `MAPLE.INI` file in the *Maple* library directory (by default `\MAPLEV\LIB`), by using one variable per package. Since it is a non-trivial work to set all the variables, `ACELib, SGLib, SPLib, TABLib` and so on, we provide a model of rows that must be added in the `MAPLE.INI` file, so-called `MAPLE.MOD` in the `BIN` directory. Here is this file for *Maple V* release 3:

```
##########
# File @(#) MAPLE.MOD (Maple) Wed May 10 09:48:17 Fre 1995
##########
#
```

(1) which is usually the `\MAPLEV\LIB` directory.

```
#################################################################
#                                                               #
# Insert the following lines in the \MAPLEV\LIB\maple.ini file. #
#                                                               #
#################################################################
#
## ACE package...
#
####################################################################
#                                                                  #
# Modify the following line so that the ACELib variable  contains  #
# the directory where one wants maple to store the compiled files. #
#                                                                  #
####################################################################
#
ACELib:=`/MAPLEV/LIB`:
#
ACELibSeq:=SGLib, SPLib, SYMFLib, TABLib, TYPLib, HEKALib,  ERRLib,
           CLGLib, TEXLib, SGALib, NCALib, IDCALib, NCSFLib, FMLib,
           BNLib, BNALib, CGLib, COMPLib, PARTLib, FREELib, SFALib:
#
BNLib  :=``.ACELib.`/ACE`:              BNALib :=``.ACELib.`/ACE`:
CLGLib :=``.ACELib.`/ACE`:              CGLib  :=``.ACELib.`/ACE`:
COMPLib:=``.ACELib.`/ACE`:              ERRLib :=``.ACELib.`/ACE`:
FMLib  :=``.ACELib.`/ACE`:              FREELib:=``.ACELib.`/ACE`:
HEKALib:=``.ACELib.`/ACE`:              IDCALib:=``.ACELib.`/ACE`:
NCALib :=``.ACELib.`/ACE`:              NCSFLib:=``.ACELib.`/ACE`:
PARTLib:=``.ACELib.`/ACE`:              SFALib :=``.ACELib.`/ACE`:
SGALib :=``.ACELib.`/ACE`:              SGLib  :=``.ACELib.`/ACE`:
SPLib  :=``.ACELib.`/ACE`:              SYMFLib:=``.ACELib.`/ACE`:
TABLib :=``.ACELib.`/ACE`:              TEXLib :=``.ACELib.`/ACE`:
TYPLib :=``.ACELib.`/ACE`:
#
## Where to find the packages...for Maple V Release 1
#
# _liblist:=[ACELib, ACELibSeq]:
#
## Where to find the packages...for Maple V Release > 1
#
  libname:=ACELib, ACELibSeq, libname:
```

Thus, to tell *Maple* where to compile (and later, where to find) the packages, we insert all lines contained in the `MAPLE.MOD` file into the existing `MAPLE.INI` file in the *Maple* library directory (avoid to insert these lines at the end of the `MAPLE.INI` file)

Then, we modify the `ACELib` variable in the `MAPLE.INI` file, setting it to an **existing** directory that differs from the `\MAPLEV\LIB` one, but we recommand not to change this variable since one can have some troubles with the installation program to create the needed directories.

Note that, according to the given `MAPLE.MOD` file, all packages will be compiled into subdirectories of the `ACELib` directory but even if it is not necessary, it is easier to manage.

Global installation

For *Maple V* release 4, we simply go in the `BIN` directory and then we enter the MS-DOS command:

```
INSTALL.BAT
```

Then, we go to the next section.

For previous versions of *Maple V*, we have to use the following instructions. Having modified the `MAPLE.INI` file, we are ready to install all packages by using the `INSTALL.ACE` program in the `BIN` directory. Being in this directory, enter the MS-DOS command, `INSTALL.BAT` which prints some information about the installation among which it is said that this program takes as argument the directory where the packages must be installed, and this directory is exactly the same as the directory specified by the `ACELib` variable in the `MAPLE.INI` file. Then, to install the environment, enter a MS-DOS command like:

```
INSTALL.BAT \MAPLEV\LIB
```

3.2.5 Using the environment

We can use `ACE` by running *Maple* and then loading the complete `ACE` environment (all packages) by the *Maple* command `with(ACE);`. We can load a single package, say `SYMF`, by the *Maple* command `with(SYMF);`.

3.3 Installing ACE on a MACINTOSH system

This section deals with the installation of `ACE` on a MACINTOSH. `ACE` is both available as a "`gzip`ed" tar archive file and a BinHexed archive. First, we have to extract files from the archive files. According to the *Maple V* release we want `ACE` to be installed for, we get either the corresponding compressed tar file `ACE1MAC.tgz` (*Maple V* release 1), `ACE2MAC.tgz` (*Maple V* release 2), `ACE3MAC.tgz` (*Maple V* release 3) or `ACE4MAC.tgz` (*Maple V* release 4) or BinHexed archive file, `ACE1MAC.sit.hqx` (*Maple V* release 1), `ACE2MAC.sit.hqx` (*Maple V* release 2), `ACE3MAC.sit.hqx` (*Maple V* release 3) or `ACE4MAC.sit.hqx` (*Maple V* release 4). Once downloaded, we have to extract all files from the archive using the `gunzip` and `tar` programs for MACINTOSH or `Stuffit Expander` application (for `.sit.hqx` files). This last application is shareware available on every **infomac archive** on the Internet.

Remark. Note that we have not tested the installation of the MACINTOSH version on several systems. We only have installed `ACE` on a MACINTOSH running a very old release of *Maple V*. Do not hesitate to contact us if you have any trouble with the MACINTOSH installation and furthermore please send us emails if you have ideas to improve the installation of `ACE` on a MACINTOSH system. Because we had not a recent version of *Maple V*, we were not able to do a better automatic installation program.

In the following, we will suppose that we get the `ACE3MAC.sit.hqx`, or in other words, we have to install the `ACE` environment on a MACINTOSH system for *Maple V* release 3.

3.3.1 Restoring sources

The `ACE3MAC.sit.hqx` file is a BinHexed archive. To extract all files from the archive, we have to **drop** the `ACE3MAC.sit.hqx` file on the `Stuffit Expander` icon. It will create a folder `ACE_PACK` in which all files will be extracted from the archive file. It will also create a folder `ACE` (containing subfolders but without any file), and a `MAPLEINIT` file. These two folders and the `MAPLEINIT` have to be moved into the `lib` folder next to the `Maple` folder, before installing the `ACE` environment.

3.3.2 Description of the `ACE_PACK` folder

The `ACE_PACK` folder contains all sources of the `ACE` environment.

The `ACE_PACK/BIN` folder contains the installation program, *i.e.* the `INSTALL.ACE` file (which is a *Maple* file), the `INSTALL.SR1`, `INSTALL.SR2` files and `INSTALL.HL1`, `INSTALL.HL2` files which describe how to respectively install source files and help files. We also give the `INSTALL.DAT` file which describes how to install pre-computed data.

The `ACE_PACK/CPRIGHTS` subfolder contains the `rights` file that gives the copyright notice.

The `ACE_PACK/PACKAGES` subfolder contains the declarations of all packages included in `ACE`, whereas the `ACE_PACK/SOURCES` subfolder contains the sources of all functions, together with some files devoted to the installation. Moreover, the `data` subfolders of each folder of the `ACE_PACK/SOURCES` folder contain the corresponding pre-computed tables of data.

3.3.3 Description of a typical package

Let us choose the `FM` package as a typical one to present the organization of any package on the disk. It is essentially stored in the `FM` folder of respectively the `SOURCES` folder for the *Maple* sources, installation files and data, and the `PACKAGES` subfolder for the *Maple* declaration of the package.

The `SOURCES/FM` folder contains the `src` folder in which the *Maple* code of all functions of `FM` is, the `help` folder that clusters the *Maple* help files for each function of the package. Finally, the `data` subfolder contains the pre-computed tables of data for the `FM` package and the `test` subfolder contains a file `FM.tst` to test the installation (`read ‘FM.tst‘;`).

The `PACKAGES/FM` folder contains three files: `FM` that declares the package, `FMext` which declares all functions used in the package and finally, `FMhlp` that declares to *Maple* where to find the compiled help files for the `FM` package.

3.3.4 Configuration file

Having restored all the environment on the disk, we are now able to make the installation.

First of all, we have to tell *Maple* where to store the compiled files (*i.e.* the `.m` files). This is done in the `MAPLEINIT` file in the *Maple* library directory (by default `:lib`), by using one variable per package. Since it is a non-trivial work to set all the variables, `ACELib, SGLib, SPLib, TABLib` and so on, we provide a model of the `MAPLEINIT` file that we have previously moved into the `lib` folder next to the `Maple` folder. Here is this file for *Maple V* release 3:

```
##########
# File @(#) MAPLEINIT (Maple) Wed Feb  7 17:29:13 MET 1996
##########
#
######################################################
#                                                    #
# Insert the following lines in the MAPLEINIT file in #
# the folder  given  by  the libname MAPLE variable.  #
# It should be by default :lib.                      #
#                                                    #
######################################################
#
## ACE package...
#
########################################################
#                                                      #
# DO NOT modify the following line  except  if  Maple  #
# library has been moved and is not in the folder :lib. #
#                                                      #
########################################################
#
ACELib:=``.libname:
#
ACELibSeq:=SGLib, SPLib, SYMFLib, TABLib, TYPLib, HEKALib,  ERRLib,
           CLGLib, TEXLib, SGALib, NCALib, IDCALib, NCSFLib, FMLib,
           BNLib, BNALib, CGLib, COMPLib, PARTLib, FREELib, SFALib:
#
BNLib  :=``.ACELib.`:ACE`:              BNALib :=``.ACELib.`:ACE`:
CLGLib :=``.ACELib.`:ACE`:              CGLib  :=``.ACELib.`:ACE`:
COMPLib:=``.ACELib.`:ACE`:              ERRLib :=``.ACELib.`:ACE`:
FMLib  :=``.ACELib.`:ACE`:              FREELib:=``.ACELib.`:ACE`:
HEKALib:=``.ACELib.`:ACE`:              IDCALib:=``.ACELib.`:ACE`:
NCALib :=``.ACELib.`:ACE`:              NCSFLib:=``.ACELib.`:ACE`:
PARTLib:=``.ACELib.`:ACE`:              SFALib :=``.ACELib.`:ACE`:
SGALib :=``.ACELib.`:ACE`:              SGLib  :=``.ACELib.`:ACE`:
SPLib  :=``.ACELib.`:ACE`:              SYMFLib:=``.ACELib.`:ACE`:
TABLib :=``.ACELib.`:ACE`:              TEXLib :=``.ACELib.`:ACE`:
TYPLib :=``.ACELib.`:ACE`:
#
## Where to find the packages...for Maple V Release 1
#
# _liblist:=[ACELib, ACELibSeq]:
#
## Where to find the packages...for Maple V Release > 1
#
  libname:=ACELib, ACELibSeq, libname:
```

We do not in general have to modify `ACELib`. *Maple* should find installed libraries.

Global installation

We are ready to install all packages by using the `INSTALL.ACE` *Maple* program in the `ACE_PACK/BIN` folder. Run *Maple* and enter the following *Maple* command:

```
read `:lib:ACE_PACK:BIN:INSTALL.ACE`;
```

which prints instructions to install `ACE`. To follow the instructions, we recommand to select all lines (output of the previous *Maple* command), put them as `input` lines, and save them as a *Maple* `Worksheet`. Then, quit and click on the created icon to run *Maple* and **execute one block of commands each time**. For that purpose, we recommand `cut` and `copy` technic because we have to run *Maple* about 20 times(2). Blocks of commands are separated by an empty line and each block tells *Maple* to terminate the session each time.

Remark. We can see the result of each block of commands by clicking on `Cancel` when *Maple* asks us to confirm we want to terminate the session. This can be useful to detect that there is a problem.

3.3.5 Using the environment

We can use `ACE` by running *Maple* and then loading the complete `ACE` environment (all packages) by the *Maple* command `with(ACE);`. We can load a single package, say `SYMF`, by the *Maple* command `with(SYMF);`.

(2)We did not succeed in having an easier way to install `ACE`, but fill free to inform us if you have better ideas.