2

System Installation

2.1 Hardware Requirements

For the installation of the CADIC system you need one of the following hardware configurations:

- Sun Sparc Station 2, 10 or 20 with SunOS 4. 1. x
- PC486/586 with Linux 1.x
- HP Apollo Series 700

The system has been developed and tested on the first configuration. On the two other configurations it has been compiled and installed successfully, but not all components have been tested. There could occur some troubles which might be caused by specific sources of the operating system.

You should have at least 90MB of free disk space and 8MB of memory. For best use of the system your workstation or PC should have a color screen with a resolution of at least 1024×768 pixels. This is an important aspect, if you plan to install the system on a linux PC, because your VGA graphics card may not support this resolution.

If you get a copy of the executable CADIC system, you can skip the following software requirements. In this case you only have to perform the unpacking steps in section 2.3. After that you can start the system by calling one of the example scripts in the directory `Examples`.

2.2 Software Requirements

In order to compile the sources of the system you need a C– and C++– Compiler as well as a scanner and parser generator. They have been successfully compiled with one of the following configurations:

- SunOS `cc` and At&T CC with `yacc` and `lex`
- `gcc` and `g++` with `bison` and `flex`

If you use the GNU software the version of the compilers should be 2.5.8 or newer.

For the graphical surface of the system you need an installation of the X–Window System X11R5/R6.

Make sure that the compiler commands `cc` (`gcc`) and `CC` (`g++`) are in your command path as well as `lex` (`flex`) and `yacc` (`bison`). Further you need access to the X-Window libraries and header files.

2.3 Installation Procedure

If you have the required hardware and software environment you can begin with the installation procedure. You can install the system anywhere in the directory tree, i.e. you need not to have root permissions to perform the installation. If you want to take a first glance at the CADIC system we recommend that you install it within a subdirectory of your home directory. You can do this with the following commands:

```
cd $HOME

mkdir Cadic

cd Cadic
```

In the following we call the newly created subdirectory `Cadic` the top installation directory or top directory for short. Now you copy the CADIC archive file into the top directory and unpack the sources. Here we assume

that you have got the sources via magnetic tape. The commands to copy it to the hard disk are:

```
mt -f /dev/nrst0 rewind
```

```
dd if=/dev/nrst0 of=./cadic.tgz
```

You also may get the archive file `cadic.tgz` directly via anonymous ftp from `hamster.cs.uni-sb.de`.

Now you can unpack the archive file with the following command sequence:

```
cat cadic.tgz | gunzip | tar xvf -
```

After that you can remove the original archive file in order to save disk space:

```
rm -f cadic.tgz
```

After unpacking the archive file you will see the following list of directories and files in the top directory:

```
Data          Imakefile  Libs     System       cadic.tmpl
Examples      Includes   Src      Tools
INSTALLATION  Leda       Surface  cadic.rules
```

`Data`: This directory contains subdirectories for global and temporal data of the system. The descriptions of the basic cell libraries are located here as well as some example designs.

`Examples`: This directory contains some example shell scripts which setup the environment variables of CADIC and start the system. You can call these scripts after the installation is completed.

`INSTALLATION`: This file contains a short description of the installation commands. If you are really impatient and familiar with installing software on a unix machine, this file shows to you the steps to perform the installation.

`Includes`: This directory subtree contains the header files of the system.

`Leda`: This directory subtree contains version 3.0 of the LEDA [MN89] library. CADIC depends on this special version. Because you might not

have an installation of the LEDA library, it is included in the CADIC distribution. It will be compiled during the installation, but it will not replace any previous installation of LEDA.

`Libs`: Initially this directory is empty. It will contain the object code libraries of CADIC and LEDA which are created during the compilation of the system sources. If you plan to write programs using the basic data structures of CADIC , you will find the corresponding libraries in this directory and have to set the linker options appropriately.

`Src`: This directory subtree contains the sources of CADIC , divided into several libraries which contain the functions of the basic data structure and the integrated design tools.

`Surface`: This directory contains the description of the graphical surface of CADIC . For each design tool there exists a file which configures its menu entries.

`System`: This directory contains the main part of the CADIC system. After the installation is completed it will contain the executable program which is called `cadic`. You can call it directly from this directory, but it is more convenient to use a shell script from the `Examples` subdirectory which sets up a correct environment for the system, before it is started.

`Tools`: This directory contains some tools which are needed for the installation. The main tool is a compiler which translates the surface descriptions from the `Surface` subdirectory into a C module. This module serves for the menu display, selection and activation of the corresponding functions. It is included into the main part of the surface.

`cadic.rules`: This file contains some macros which are used by `imake` to create the Makefiles.

`cadic.tmpl`: This file contains a templative description of the installation environment. You have to edit this file in order to set up the compiler commands etc. as you will see in the following.

Before you can start the compilation of the sources you have to edit the

macros in `cadic.tmpl`. The macro `CC` is set to the C compiler command. In the default distribution it is set to `gcc`, change it, if you want to use the traditional SunOS C compiler. The next macro `CPLUSPLUS` points to the C++ compiler, it is set to `g++`. In the same way change the macros `YACC` and `LEX`, which represent the commands for the parser and scanner generator.

Finally take a look at the macro `CDEBUGFLAGS`. This defaults to `-g` in order to create a debugging version of the system. If you do not want this, just erase any text behind the equal sign in this line and set `CDEBUGFLAGS` to the empty string. This results in much smaller object files and executables and is recommended, if you have less free disk space.

Now start the installation from the top directory with the command:

```
xmkmf
```

Note that you need an installation of the X–Window system to use this command. The directory `/usr/bin/X11` must be in your command path. After that, you create the makefiles for CADIC with the command:

```
make Makefiles
```

After this command is completed, you can start the compilation of the source files and the graphical surface simply with the command:

```
make >& cadic.log &
```

Depending on your hardware platform this command will take several hours to complete. If there are any error messages, you can read them on the file `cadic.log`.

If it has finished without error, you can find the executable CADIC system in the subdirectory `System`. It is called `cadic`. Before you can start the program, you have to setup an environment, i.e. you have to define a list of environment variables. In the subdirectory `Examples` you can find some sample shell scripts which set these variables for some example designs and then start the system.