Napier88 Installation Guide

Release 2.2.1

August 1996

Graham Kirby
Richard Connor
Quintin Cutts*
Ron Morrison
Dave Munro

University of St Andrews, North Haugh, St Andrews, Fife KY16 9SS, Scotland.
*University of Glasgow, Lilybank Gardens, Glasgow G12 8QQ, Scotland.

This document should be referenced as:
"Napier88 Installation Guide (Release 2.2.1)".
University of St Andrews (1996).

Contents

1 Introduction

This guide describes how to install Napier88 Release 2.2.1. See the Napier88 Reference Manual [MBC+96] for the definition of the Napier88 language, and the Napier88 Standard Library Reference Manual [KBC+96] for a description of the Napier88 Standard Library.

Napier88 Release 2.2.1 runs on the following configurations:

- Sun SPARC running SunOs[1] Version 4.1.3.
- Sun SPARC running Solaris Version 1.1.
- DEC Alpha running OSF/1[2] .

Some parts of the installation will require root access. For assistance with installation problems mail `napier@dcs.st-andrews.ac.uk`.

Note 1: machines must be registered before Napier88 will run, as described in Section 4. You may wish to obtain authorisation codes before proceeding with installation.

Note 2: with OSF the kernel may need to be configured to support memory mapped files. For details mail `napier@dcs.st-andrews.ac.uk`.

2 Downloading

The release provides a number of alternative pre-populated persistent stores of various sizes. The empty store is a minimal store which does not contain any persistent data; the intermediate store contains the Napier88 compiler only, while the full store contains the entire standard library. The intermediate and full stores are available either with or without procedure source code included.

The following table shows the names used to refer to the various stores, and their sizes. The first figure for each store gives the compressed size at the ftp site; the second figure gives the expanded size.

store contents	source code included	no source code
empty	*empty*, 125K → 1.6M	
compiler only	*compiler*, 2.3M → 9.7M	*compilerNoSource*, 1.2M → 4.9M
full store	*full*, 12.3M → 48.8M	*fullNoSource*, 7.6M → 30.8M

Any or all of these stores may be downloaded as local space permits. The versions without source code are fully functional; the only difference is that the source code of the library procedures is not present for browsing.

The release may be downloaded by anonymous `ftp` from the following directory:

`ftp://ftp-ppg.dcs.st-andrews.ac.uk/pub/Napier88/release2.2.1/`

---

[1] SunOs™ and Solaris™ are trademarks of Sun Microsystems, Inc.

[2] OSF/1™ is a trademark of the Open Software Foundation.

1. The following files are required:

interpreter release file:
`napier88.rel2.2.1.tar.gz` (1.1M)

installation script:
`INSTALL` (5K)

empty formatted store file:
`stores/stablestore.empty.gz` (125K)

2. At least one of the following populated store files is required:

`stores/stablestore.compilerNoSource.gz`	(1.2M)
`stores/stablestore.fullNoSource.gz`	(7.6M)
`stores/stablestore.compiler.gz`	(2.3M)
`stores/stablestore.full.gz`	(12.3M)

3. The following documentation and registration files may also be downloaded:

`docs/agreement.ps.gz`	(6K)
`docs/registration.ps.gz`	(6K)
`docs/lib.man.ps.gz`	(246K)
`docs/ref.man.ps.gz`	(107K)

The files `agreement.ps.gz` and `registration.ps.gz` contain registration forms. The file `lib.man.ps.gz` contains the Napier88 standard library manual and `ref.man.ps.gz` contains the Napier88 reference manual.

3 Installing

3.1 Base Installation

The installation script `INSTALL` will extract the interpreter and store files and move them to the appropriate locations. It requires the `gunzip` program which is available from:

`ftp://prep.ai.mit.edu/pub/gnu/`

1. Set the environment variable `NPRDIR` to the full path of the intended final position for the Napier88 release directory.

2. Set the environment variable `NPRSTORE` to the full path of the intended final position for the initial persistent store directory. This can be within the release directory or somewhere completely different. (It is possible to have multiple store directories; this is explained later).

3. Set the environment variable `NPRHEAP` to the desired local heap size in Mbytes. It is recommended that this does not exceed a quarter of the physical memory. If the variable is not set, a default value of 8 Mbytes will be used.

4. Edit the definitions of `NPRSCRIPTS` and `STORETYPE` at the start of the installation script. The `NPRSCRIPTS` variable defines the directory which will contain the release-independent scripts and programs for running the Napier88 system. This directory should be in the search path of Napier88 users, and may lie outside the directory `NPRDIR`. The `STORETYPE` variable defines the type of store to be installed, and should be set to one of `compiler`, `full`, `compilerNoSource` or `fullNoSource`.

5. Run the installation script `INSTALL`.

3.2 Network Software Setup

The steps in this section will probably require root access. They need only be carried out if the Napier88 distributed facilities are required.

1. The NIS server must be informed which port number and protocol to use for the Napier88 socket mechanism. Find out which machine is the NIS server for the local machine:

```
ypwhich -m services
```

2. On the NIS server machine, add the following line to the file `/etc/services`:

```
napier88        8109/tcp                        # napier88 remote access
```

3. The local machine must now be informed which program to run when a socket connection request is received. On the local machine, add the following (all on one line) to the file `/etc/inetd.conf`. ***user*** should be either *root* or some user whose default file creation permissions allow read and write by users running the Napier88 system. ***arch*** should be either *sun4* or *alpha* as appropriate.

```
napier88  stream  tcp  nowait  user
      $NPRSCRIPTS/nprexec/arch_listener  listener
```

Thus an example entry might be:

```
napier88 stream tcp nowait napier /langs/napier/scripts/nprexec/sun4_listener listener
```

4. On the local machine, find the process id of the *inet* daemon and reconfigure it by sending it a HUP signal:

```
ps auxw | grep inetd | grep -v grep
kill -HUP id of resulting process
```

4 Registering a Napier88 System

Napier88 Release 2.2.1 is Copyright © University of St Andrews 1996 and is subject to a licence fee. If you have a licence for an earlier release of Napier88, however, Release 2.2.1 is free. You are required to sign a release agreement. A copy of the agreement is in the file available at:

```
ftp://ftp-ppg.dcs.st-andrews.ac.uk/
      pub/Napier88/release2.2.1/docs/agreement.ps.gz
```

The software includes host machine authorisation and will not run on unregistered machines. The licence fee covers the use of any number of machines at one site. A machine registration form is in the file available at:

```
ftp://ftp-ppg.dcs.st-andrews.ac.uk/
      pub/Napier88/release2.2.1/docs/registration.ps.gz
```

To register, sign two copies of the release agreement and fill in the host registration form. Send them, with the licence fee as specified in the release agreement if applicable, to the following address:

Napier88 Registration
Persistent Programming Group
Division of Computer Science
University of St Andrews

North Haugh
St Andrews
Fife KY16 9SS
Scotland

Payments should be made out to **The University of St Andrews**. One of the copies of the release agreement will be returned to you, counter-signed.

You will receive authorisation codes for the registered machines by electronic mail. To register your machines, use the command `nprregisterhost`. It takes as a command line argument either the authorisation code for a single machine, or the name of a file containing multiple authorisation codes.

If you already have an earlier release of Napier88 you can use the existing list of authorisation codes with Release 2.2.1. However, please still send the signed copies of the release agreement. To install the existing authorisation codes:

```
cat $OLDNPRDIR/interpreter/release/exec/PSKEYS >>
$NPRDIR/interpreter/release/exec/PSKEYS

or  cat $OLDNPRDIR/exec/PSKEYS >>
$NPRDIR/interpreter/release/exec/PSKEYS
```

depending on the structure of the earlier release directory.

Subsequently, new machines may be registered by sending a further copy of the registration page of the release agreement, or by sending the required information by email to the address given in Section 1.

If you wish you may send a copy of your registration information by email to napier@dcs.st-andrews.ac.uk as well as by postal mail. In this case authorisation codes will be emailed back by return.

5 Testing the Installation

To perform a simple test of your installation, try compiling and running the test program *hello.N* in

```
$NPRDIR/interpreter/release/napierprogs/hello.N
```

for example,

```
cd $NPRDIR/interpreter/release/napierprogs
npc hello.N
npr hello.out
```

The result should be a message saying hello. Note that this does not test the distributed facilities. See the Standard Library Manual for a full description of Napier88 commands.

6 Creating New Stores

To create a new persistent store for a user, first set the environment variable `$NPRSTORE` to the new store directory, and then run the command `nprcopystore` with the required store type as argument e.g.

```
nprcopystore full
```

7 Common Problems

When the programming environment is run from an xterm, it starts inside the xterm window instead of creating a new window.

Add the line

```
unset WINDOWID
```

at the beginning of the file `$NPRDIR/interpreter/release/bin/npr`.

———

When attempting to compile or run a program the following message is obtained:

PAM error:
stable store: cannot open the store

Check that the environment variable `$NPRSTORE` is set correctly to the directory containing the stable store, and that the access permissions allow read/write/execute access to the directory and read/write access to the files within it.

———

When attempting to compile a program or run the programming environment one of the following messages is obtained:

Cannot find InteractiveEnvironment with type:
env

Error cannot find - compileFile

Error cannot find - FailValues

This probably means that an inappropriate store file has been used to initialise the current store. Either a compiler store or full store is needed to do compilation, and a full store is needed to run the programming environment. Copy a new store file as described in Section 6.

8 Release Directory Structure

The release directory on the `ftp` server contains the following:

`README`	description of directory contents
`INSTALL`	installation script
`napier88.rel2.2.1.tar.gz`	tar'd interpreter files
`stores/stablestore.empty.gz`	empty formatted store file
`stores/stablestore.compilerNoSource.gz`	compiler store file, no source code
`stores/stablestore.fullNoSource.gz`	full store file, no source code
`stores/stablestore.compiler.gz`	compiler store file
`stores/stablestore.full.gz`	full store file
`docs/agreement.ps.gz`	release agreement document
`docs/registration.ps.gz`	registration document
`docs/install.ps.gz`	installation guide (this document)
`docs/lib.man.ps.gz`	standard library manual
`docs/ref.man.ps.gz`	reference manual

Once extracted, the interpreter directory `napier88.rel2.2.1` contains the following:

commands to use the Napier88 system:

`bin/napier88`	script to run *startProgrammingEnv.out* in napierprogs
`bin/npb`	script to run *npboot.out* in napierprogs
`bin/npc`	script to run *npcompile.out* in napierprogs
`bin/npr`	script to run *$ARCH_interpreter* in exec
`bin/nprcompact`	script to run *$ARCH_sstorecompact* in exec
`bin/nprcopystore`	script to copy store to *$NPRSTORE*
`bin/nprformat`	script to run *$ARCH_mksstore* in exec
`bin/nprgc`	script to run *$ARCH_gcstore* in exec
`bin/nprregisterhost`	script to add encrypted hosts to *PSKEYS* in exec
`bin/nprsethost`	script to set the host for the current store
`bin/nprstats`	script to run *$ARCH_stats* in exec
`bin/nps`	script to run *npsavetypes.out* in napierprogs

release-independent commands and programs to use the Napier88 system:

`cmd/napier88`	script to run *napier88* in current *bin* directory
`cmd/npb`	script to run *npb* in current *bin* directory
`cmd/npc`	script to run *npc* in current *bin* directory
`cmd/npr`	script to run *npr* in current *bin* directory
`cmd/nprcompact`	script to run *nprcompact* in current *bin* directory
`cmd/nprcopystore`	script to run *nprcopystore* in current *bin* directory
`cmd/nprformat`	script to run *nprformat* in current *bin* directory
`cmd/nprgc`	script to run *nprgc* in current *bin* directory
`cmd/nprregisterhost`	script to run *nprregisterhost* in current *bin* directory
`cmd/nprsethost`	script to run *nprsethost* in current *bin* directory
`cmd/nprstats`	script to run *nprstats* in current *bin* directory
`cmd/nps`	script to run *nps* in current *bin* directory
`cmd/nprexec/alpha_listener`	executable program to handle socket connections (alpha)
`cmd/nprexec/sun4_listener`	executable program to handle socket connections (sun4)

executable programs used by the commands in `bin`:

`exec/alpha_gcstore`	executable program to garbage collect a store (alpha)
`exec/alpha_interpreter`	executable program to run the Napier88 interpreter (alpha)
`exec/alpha_mksstore`	executable program to create an empty store (alpha)
`exec/alpha_sstorecompact`	executable program to compact a store (alpha)
`exec/alpha_sstorelockhost`	executable program to lock a store (alpha)
`exec/alpha_stats`	executable program to gather statistics (alpha)
`exec/sun4_gcstore`	executable program to garbage collect a store (sun4)
`exec/sun4_interpreter`	executable program to run the Napier88 interpreter (sun4)
`exec/sun4_mksstore`	executable program to create an empty store (sun4)
`exec/sun4_sstorecompact`	executable program to compact a store (sun4)
`exec/sun4_sstorelockhost`	executable program to lock a store (sun4)
`exec/sun4_stats`	executable program to gather statistics (sun4)
`exec/PSKEYS`	file containing encrypted hosts and domains

Napier88 programs:

`napierprogs/clearHashTable.out`	Napier88 program to clear the types cache
`napierprogs/npboot.out`	Napier88 program to run the boot compiler
`napierprogs/npcompile.out`	Napier88 program to run the compiler
`napierprogs/npsavetypes.out`	Napier88 program to compile type definitions
`napierprogs/receiver.out`	Napier88 program used in copying stores
`napierprogs/sender.out`	Napier88 program used in copying stores
`napierprogs/startProgrammingEnv.out`	Napier88 program to start programming environment
`napierprogs/hello.N`	source of Napier88 test program

9 References

[KBC+96] Kirby, G.N.C., Brown, A.L., Connor, R.C.H., Cutts, Q.I., Dearle, A., Dunstan, V.S., Morrison, R. & Munro, D.S. “Napier88 Standard Library Reference Manual (Release 2.2.1)”. University of St Andrews (1996).

[MBC+96] Morrison, R., Brown, A.L., Connor, R.C.H., Cutts, Q.I., Dearle, A., Kirby, G.N.C. & Munro, D.S. “Napier88 Reference Manual (Release 2.2.1)”. University of St Andrews (1996).

An index to these and other publications on persistence is available at:

`http://www-ppg.dcs.st-andrews.ac.uk/Publications/`