



## The Helios Getting Started Tutorial

---

### ACKNOWLEDGEMENTS

This manual was written by Bill Noble and Rachel Ganz of Vardas.

### COPYRIGHT

This document is Copyright ©1990, Distributed Software Ltd. All rights reserved. This document may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from Distributed Software Limited, The Maltings, Charlton Rd, Shepton Mallet, Somerset, BA4 5QE, UK.

### TRADEMARKS

Helios is a trademark of Perihelion Software Ltd. DSL/// is a trademark of Distributed Software Ltd. UNIX is a trademark of AT and T.

### EDITION

This edition first published October 1990.  
Published by Distributed Software Ltd.

Printed in the UK.  
Part number H09016  
Edition 1

The Helios software is available from Distributed Software Limited.  
The Maltings, Charlton Rd, Shepton Mallet, Somerset, BA4 5QE, UK, and from selected distributors worldwide.

1	INTRODUCTION . . . . .	2
1.1	Typographic Conventions . . . . .	3
2	USING HELIOS . . . . .	5
2.1	Starting Helios . . . . .	5
2.2	Typing, Editing and Executing Commands . . . . .	9
2.3	Files and the Filing System . . . . .	21
3	ADVANCED USE OF HELIOS . . . . .	50
3.1	More About Files . . . . .	52
3.2	Pipes, Background Jobs and Remote Tasks . . . . .	58
3.3	The Helios Shell . . . . .	65
3.4	Logging in and Login Directories . . . . .	66
3.5	Customising Your Shell . . . . .	72
3.6	Useful Utilities . . . . .	85
3.7	Conclusion . . . . .	90
4	FURTHER READING . . . . .	91
APPENDICES		
A	SUMMARY OF COMMANDS . . . . .	92
B	INDEX . . . . .	94

---

# INTRODUCTION

This tutorial is aimed at readers who are about to use Helios for the first time and have little or no experience of Unix-like systems. It provides a simple introduction to the most commonly used features of the Helios user interface.

The tutorial consists of a sequence of descriptions of commands with associated examples. The examples should be done in the order they are given. They are boxed to make them stand out from the descriptions. The bold text shows what you should type in and the lighter text shows what you should see on the screen. All actions in the tutorial are numbered consecutively.



**It is essential that you do all the examples in the correct order as they are interdependant.**

Each time a new command is introduced its name appears in the margin together with a one line summary in the main column. This should help you to reference them quickly.

In order to keep this tutorial short it does not give complete descriptions of the commands that are mentioned. Some of the command options are not mentioned at all. Inevitably there will be questions raised which are not answered here. Further information on all the commands is provided in "The Helios Operating System" manual.

## 1.1 *TYPOGRAPHIC CONVENTIONS*

Throughout this tutorial the following typographic conventions have been used:

*italic* Words appearing in *italic* mark a new term in the text. This is where the term will be defined.

screen text Words appearing in the screen text font refer to command names, command lines, and any text which is displayed on a screen.

**user text** Words appearing in the **user text** font refer to text that should be typed by the reader when doing one of the examples.

**KEY** Words that appear within a **BOX** refer to keys on the keyboard.

**X|Y** Means key **X** should be pressed and then key **Y**.

**X+Y** Means key **X** should be pressed at the same time as key **Y**.

DSL//

3

---

### The Helios Getting Started Tutorial

This tutorial assumes that you have correctly installed your copy of Helios.

Some of the practicals assume you are running Helios with a PC hosted system.

DSL//

4

---

## USING HELIOS

This chapter provides a brief introduction to the most commonly used features of the Helios user interface.

### 2.1 STARTING HELIOS

Before trying the examples in this tutorial you must start Helios.

1 Make the `\helios` directory the current directory. E.g.,

```
C: cd \helios RETURN
C:
```

DSL//

5

---

### The Helios Getting Started Tutorial

2 Start Helios by executing the `server` program:

```
C: server RETURN
```

The `server` program loads Helios onto the transputer board.

You should now have a running Helios system (if not, you should check carefully that you have installed Helios correctly).

DSL//

6

The information displayed on your screen will look something like this:

```

                                                    console
                Helios Operating System
                  Version xx
    (C) Copyright 1987-90, Perihelion Software Ltd.
  
```

login:

Helios will be waiting for you to *login*. Logging in is the process by which you tell Helios who you are. This is described in more detail in the next chapter.

The default Helios system knows about a user called *guest*. You should use this name when logging in for the tutorial.



If Helios does not ask you to login it means your system has been set up differently. In this case you should consult the person responsible for installing the Helios system before continuing with this tutorial.

DSL//

7

## The Helios Getting Started Tutorial

3 Login to Helios as 'guest':

```
login:  guest RETURN
```

```
                Welcome to the Helios Operating System
```

```
%
```

If Helios does not recognise the login name you give it will first ask for a password and then display the message "Login incorrect". You will then have the opportunity to login again.

The '%' character displayed on the line after the welcome message is the Helios *command-line prompt*. This tells you that Helios is ready to receive a command. When a command has been executed Helios responds with another command-line prompt.

DSL//

8

## 2.2 TYPING, EDITING AND EXECUTING COMMANDS

All commands issued to Helios at the command-line prompt, are *interpreted* (i.e., understood and acted upon) by a part of the Helios operating system called the *shell*. The shell is a program which interfaces between you and the operating system. The Helios shell has been designed to look as similar to the standard Unix shell (`csh`) as possible.



Users who are familiar with the Unix shell should have no difficulty in using Helios. However you should note that some Unix shell commands are not available under Helios, and that some commands with the same name may not behave in exactly the same way.

Commands are given to the shell by typing the command name, together with any parameters, at the command-line prompt ('%'). When you press the `RETURN` key the shell program interprets the command and causes the appropriate program to run.

You can interrupt or cancel a command that has just been issued by pressing `CTRL+C` (i.e., the control key and the 'C' key pressed together).

DSL//

9

### The Helios Getting Started Tutorial

The simplest commands consist of a single lower-case word. For example:

To display today's date:

```
4 % date RETURN
Date : Tue Sep 4 10:00:22 1990
%
```

If you make a mistake when typing a command name you will get an error message of the form:

Command not found

For example:

Entering an incorrect command:

```
5 % fate RETURN
fate: Command not found.
%
```

DSL//

10

## EDITING THE COMMAND LINE

If you notice a mistake before pressing the **RETURN** key you can correct it by deleting back to the point where the error was made and retyping the rest of the command. The labeling of the delete key will vary between makes of keyboard. The delete key is usually one of:

**BACKSPACE**, **DELETE**, **DEL**, **RUBOUT**, **<—**

Alternatively you can use the control sequence **CTRL+H** to delete the previous character on the line.

Many Helios commands take parameters which affect what the commands do. The format of a command with parameters is:

**command-name** *option(s)* *filename(s)*

This means;

a command name followed by none or more options, followed by none or more file names. These terms are described below.

### command-name

All Helios commands should be entered in lower case. For example, 'Date' is not the same as 'date'. If you type 'Date' Helios will respond with the message "Command not found".

DSL//

11

---

## The Helios Getting Started Tutorial

---

### *option(s)*

The options affect the way the command works. Options are normally prefixed by the minus character '-'. Option names are mostly single letter and are case sensitive. This means that option '-t' is different from option '-T'. If a command has more than one option the options can either be typed separately, each preceded by the '-' character, or combined with just one minus sign preceding the lot. For example,

-t -s or -ts

### *filename(s)*

Many commands take information from a file and manipulate it in some way before producing some output. The *filename* parameter(s) specify which file(s) are to be processed.

You must put a space between the command name and the options, and between the options and the file names.

DSL//

12

**ls** LISTING THE FILES IN A DIRECTORY

An example of a command with parameters is the listing command, 'ls', which lists the contents of a directory.

The ls command without parameters produces a simple listing of the files within the current directory:

To display a simple listing of the names of the files in the current directory:

```
6 % ls RETURN
cshrc          examples/      login
%
```

This generates an alphabetical listing of the contents of the guest directory.

DSL//

13

---

**The Helios Getting Started Tutorial**


---

The ls command with the -l (long) parameter produces a detailed listing of the files in the directory:

To display a more detailed listing of the directory contents:

```
7 % ls -l RETURN
f rwe---da 0 128 Fri Aug 31 12:53:20 1990 cshrc
d rwvxyzda 0 0 Mon Sep 3 09:17:38 1990 example/
f rwe---da 0 1 Thu Aug 9 10:04:12 1990 login
%
```

The '-l' option tells ls to provide a line of information on each file. Amongst the information listed is whether something is a file or a directory, the size of the file and the time and date the file was last altered. The first column of the listing indicates the *type of file*. 'd' is a directory file and 'f' an ordinary file such as a data file or executable program. Following the type of file is the *access mode* which specifies who is allowed to read (r), write (w) or execute (e) the file.

DSL//

14



The `ls` command with a filename as a parameter produces a listing for that file:

To display detailed information on the file 'cshrc':

```

8 % ls -l cshrc RETURN
f rwe---da  0      128 Fri Aug 31 12:53:20 1990 cshrc
%
```

In this example the listing is limited to the file 'cshrc'.

## help

### ONLINE HELP INFORMATION

There will be times when you cannot remember what parameters a particular command takes. When this happens you can use the Helios on-line help facility which provides detailed information on every Helios command.

DSL//

15

## The Helios Getting Started Tutorial

Here is an example of the help command:

To display help information on the `printenv` command:

```

9 % help printenv RETURN
printenv: Displays environment variables

Format: printenv

Description: The printenv command is used to display the names
and values of all environment variables that are currently set.

See also: setenv, unsetenv

Quit      ?Help      Go back
 Q
%
```

To exit from the help program you must hit the `Q` key.

DSL//

16

The help facility is much more powerful than this example might imply. You can find out more about help simply by typing:

```
help RETURN
```



## COMMAND-LINE HISTORY – REISSUING OLD COMMANDS

Helios keeps a record of the commands that are issued at the command-line prompt. You can browse through this *history* using the cursor up and down keys (**↑**, **↓**).

Pressing the **↑** key causes the previously executed command line to be displayed at the command-line prompt. You can then re-issue this command by pressing the **RETURN** key. Each press of the **↑** key displays an earlier command line which can be re-issued, or not, as you wish. Helios has a limit to the number of command lines it can remember (this is set at 20 in the default system and can be modified by editing the `cskr` file (see later)).

The **↑** key scans backwards through the command-line history whilst the **↓** key scans forwards.

You can re-issue the previously issued command by typing **!! RETURN**.

DSL//

17

## The Helios Getting Started Tutorial

**To re-display and re-issue the previous command line:**

```
10 % ↑  
% help printenv RETURN  
  <More text>  
%
```

After pressing the **↑** key the cursor is positioned at the end of the line to allow you to re-issue the command.

**Examining the command-line history**

```
11 Play with the ↑ and ↓ keys to browse backwards and forwards through the command-line history.
```

history

## DISPLAYING A LIST OF RECENTLY EXECUTED COMMANDS

Helios keeps a record of previously executed commands. In the default system the last twenty commands that you typed are recorded. You can display this list using the `history` command, which takes two parameters:

```
history [-r] [<n>]
```

DSL//

18

Where:

`[-r]` is an option which causes the list to be displayed in reverse order.

`[<n>]` is an optional number which specifies that only the last `<n>` command lines should be displayed.

The shell variable 'history' specifies the maximum number of command lines displayed by the history command. Shell variables are described in the next chapter.

To display a list of previously issued command lines:

```

12 % history RETURN
    1 date
    2 fate
    3 ls
    4 ls -l
    5 ls -l cshrc
    6 help printenv
    7 help printenv
    8 history
%

```

DSL//

19

## The Helios Getting Started Tutorial

The command lines displayed by `history` are numbered. You can use these numbers to selectively re-issue the command lines of your choice. The command,

`!<n>`

will re-issue the command line number `<n>`

To re-issue the 'date' command (which was the first command issued in this tutorial):

```

13 % !1 RETURN
    date
    Date : Tue Sep  4 10:02:12 1990
%

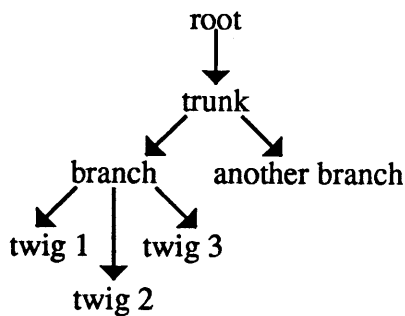
```

DSL//

20

## 2.3 FILES AND THE FILING SYSTEM

Helios files are kept in places called *directories*. Directories are hierarchical storage areas that resemble upside-down trees in structure.



Each directory has a name, in the same way as files have names. A directory is a file containing information about the whereabouts of the files in that directory.

Every file is located in a unique directory. Directories can contain sub-directories. You can think of a directory as being a folder which can contain files and other folders.

In the diagram above 'trunk' is a directory containing the directories 'branch' and 'another branch'.

DSL//

21

---

### The Helios Getting Started Tutorial

A user of Helios is always considered to be in a directory known as the *current directory*. Which directory is the current directory will depend on what your default login directory is and whether or not you have changed directory since starting Helios (for information on login directories see the next chapter).

Helios uses the current directory as the default source and destination for files used by the various Helios commands. This means that Helios will always look for the files you are using in the current directory, unless you specify otherwise (see below).

DSL//

22

When you first ran the Helios system at the start of this tutorial the current directory was `guest`. Let's have another look at this directory:

To display a list of the files in the current directory (`guest`):

```
14 % ls RETURN
cshrc          examples/      login
%
```

The `guest` directory contains a sub-directory called `examples`. The trailing `'/'` character indicates that this is a directory name.

To display the contents of the sub-directory `examples`:

```
15 % ls examples RETURN
convol/        factor/        hello/         lb/
pi/            tut/
%
```

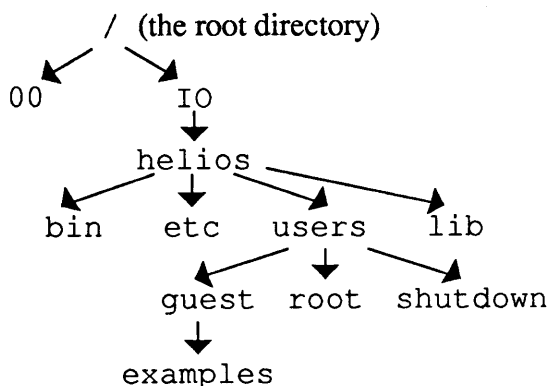
Note that the parameter to the listing command `ls` was `'examples'` and not `'examples/'`. When referring to a directory you can omit the trailing `'/'` character.

DSL//

23

## The Helios Getting Started Tutorial

The following illustration shows a simplified view of the directory structure for the standard Helios installation:



At the very top of the directory structure is the *root directory* called `'/'`. The root directory contains two sub-directories called `00` and `IO`. The `IO` directory contains a sub-directory called `helios` and so on.

By the end of this tutorial you will have created your own directory area which will be the default current directory when you next run Helios. This is called the *login directory* and is normally unique for different users.

DSL//

24

Directories are identified using *path names*. A path name specifies the unique path through the directory structure to the directory you want. Path names consist of a sequence of directory names separated by the '/' character.

There are three ways of giving the path name for a directory:

- 1 **Relative to the root directory.** This is known as the *absolute path name* because it will always identify the correct directory no matter what the current directory is (hence the term 'absolute').
- 2 **Relative to the current directory.** This is known as the *relative path name*. The relative path name will only work if it is correctly specified relative to the current directory. Relative path names are often more convenient because fewer characters have to be typed in for the path name. If the current directory is changed the relative path name for a particular directory should also change.
- 3 **Relative to an alias server.** This is described in the next chapter.

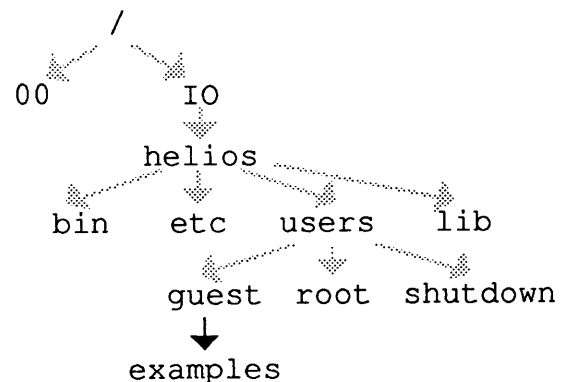
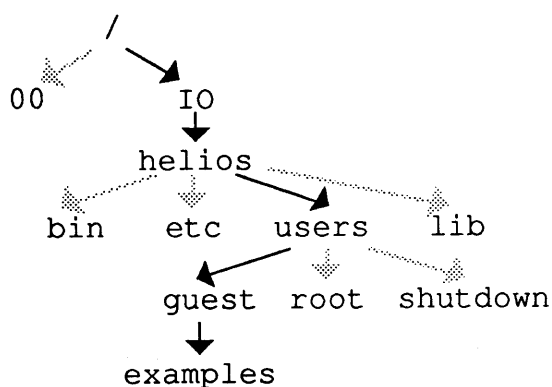
The absolute path name for the guest directory is '/IO/helios/users/guest'. Similarly, the absolute path name for the examples directory is '/IO/helios/users/guest/examples'.

If the current directory is guest the relative path name for the examples directory is 'examples'. If the current directory is IO the relative path name for examples would be 'helios/users/guest/examples'.

DSL//

25

## The Helios Getting Started Tutorial



/IO/helios/users/guest/examples

examples (relative to the guest directory)

The first character of the path name tells Helios what sort of path name is being used:

**<name>** A path name starting with a <name> is relative to the current directory.

**../** A path name starting with '../' is relative to the directory containing the current directory. For example, if the current directory is /IO/helios/users/guest/examples then ../cshrc refers to the cshrc file in the guest directory.

DSL//

26

~/ A path name starting with '~/' is relative to the login directory (see the next chapter for information about login directories).

/ A path name starting with '/' is an absolute path name.

### **pwd DISPLAYING THE ABSOLUTE PATH NAME FOR THE CURRENT DIRECTORY**

Helios provides a command called `pwd` which displays the absolute path name of the current directory. This can be useful for checking if a relative path name is correct relative to the current directory.

To find out the absolute path name of the current directory:

```
16 % pwd RETURN
    /IO/helios/users/guest
    %
```

### **cd CHANGING DIRECTORY**

All the examples so far have been relative to the default login directory 'guest'. To move to another directory (i.e., to change the current directory) use the `cd` command.

DSL//

27

## **The Helios Getting Started Tutorial**

To move to the root directory:

```
17 % cd / RETURN
    %
```

To check that the current directory is the root directory:

```
18 % pwd RETURN
    /
    %
```

To move to the IO sub-directory using a relative path name:

```
19 % cd IO RETURN
    %
```

Don't forget to use capitals for 'IO'. Directory names, like file names and command names, are case sensitive.

DSL//

28

To move to the `examples` sub-directory using an absolute path name:

```
20 % cd /IO/helios/users/guest/examples RETURN
%
```

If you use `cd` on its own (i.e., without giving it a directory name) the current directory will become the login directory (or whatever the shell variable `HOME` is set to – see next chapter).

DSL//

29

---

## The Helios Getting Started Tutorial

---

### **mkdir**

#### **CREATING NEW DIRECTORIES**

You can create new directories using the `mkdir` command. Let's create a directory called `red`:

Move to the `guest` directory which is one level up from the current (`examples`) directory:

```
21 % cd .. RETURN
%
```

Create a new directory called 'red':

```
22 % mkdir red RETURN
%
```

This creates a new sub-directory within the `guest` directory called 'red'.

DSL//

30



```

Display a list of files to check that the new directory has been created:
23 % ls RETURN
cshrc          examples/     login        red/
%
```

## FILE NAMES

There are several ways you can give a file name parameter to an application.

- The file name alone
- The complete path name of the file
- A file name using wildcards
- File name completion

### 1 The file name alone

If you just give the name of the file, Helios assumes that the file is in the current directory. For example 'cshrc' will correctly identify the cshrc file if the current directory is /IO/helios/users/guest. Many file names contain two parts separated by the character '.'.

DSL//

31

## The Helios Getting Started Tutorial

The convention is that the second part of the name (called the extension) identifies the type of the file. Common file name extensions are:

```

.con  Helios configuration file
.c    C source code files
.o    Object files (the output from the compilers)
.d    Assembler macro preprocessor files
.s    Assembly files
```

### 2 The complete path name of a file

A more precise way of identifying a file is to give the complete path name of the file. The path name can be either relative to the current directory or relative to the root directory. For example (assuming that the current directory is /IO/helios/users):

```
guest/cshrc
```

Uses a path name relative to the users directory to identify the file cshrc within the guest sub-directory.

```
/IO/helios/users/guest/cshrc
```

Uses an absolute path name to identify the file called cshrc.

DSL//

32

### 3 A file name using wildcards

Helios provides a shorthand method called *wildcards* for identifying files or groups of files. There are three types of wildcards which, if used within a file name, cause Helios to fill in the file name for you.

The examples below assume that the current directory is

```
/IO/helios/users/guest/examples/tut
```

which contains the files:

```
testa1          testa2          testb1          testb2
testc1          testc2
```

The three types of wildcard are:

- \* If you put a '\*' in a file name, Helios will attempt to replace the '\*' with any sequence of characters which results in a valid file name. A valid file name is the name of an existing file in the current directory (or the directory given by the path name). Helios will generate a filled-in file name for every valid solution. For example (assuming the current directory is tut):

```
test*   will expand to  testa1
                and     testa2
```

DSL//

33

---

### The Helios Getting Started Tutorial

```
and  testb1
and  testb2
and  testc1
and  testc2
```

```
test*1 will expand to  testa1
                and     testb1
                and     testc1
```

```
xyz*   will generate the error message "No match"
```

- ? If you put a '?' character in a file name, Helios will attempt to replace the '?' with any **single** character which results in a valid file name. For example (assuming the current directory is tut):

```
test?1 will expand to  testa1
                and     testb1
                and     testc1
```

```
?      will generate the error message "No match"
```

DSL//

34