

BASH Navigation Toolkit – Architecture, Design, and Use

Richard R. Landon – rlandon@usa.net

SourceForge Project Page -- <http://sourceforge.net/projects/bashnavigation>

Version History

| Version | Date | Author | Description |
|---------|-------------|-------------|---|
| 1.0 | 05-Apr-2008 | Rick Landon | Initial architecture and design |
| 1.1 | 19-Apr-2008 | Rick Landon | Update with command summary |
| 1.2 | 10-May-2008 | Rick Landon | Add conversion instructions and document cut functionality. |

Table of Contents

| | |
|--|-----|
| Version History | ii |
| Table of Contents | iii |
| Table of Figures | iii |
| Table of Tables | iii |
| Table of Listings | iii |
| Overview | 1 |
| Purpose | 1 |
| Architecture and Design | 1 |
| PATH Variables | 1 |
| Fundamentals | 2 |
| Components and Functions | 2 |
| Reference Manual | 3 |
| Installation and Usage | 3 |
| UNIX File Conversions | 4 |
| GNU cut compatibility mode | 4 |
| Command Summary – Bash Navigation Toolkit, Version 1.0.0 | 6 |

Table of Figures

| | |
|--|---|
| Figure 1: BASH Navigation Toolkit Reference Architecture | 3 |
|--|---|

Table of Tables

| | |
|--|---|
| Table 1: BNT Usage Guide and Command Summary | 8 |
|--|---|

Table of Listings

| | |
|---|---|
| Listing 1 -- BNT Deployment Structure | 4 |
|---|---|

Overview

This document describes the architecture of the navigation toolkit for BASH. It includes an overview of the architecture and design of the toolkit and also provides a basic set of reference materials that describe the installation and use of the toolkit.

Purpose

The BASH Navigation Toolkit (BNT) is a set of standard BASH functions and aliases that provide a means to define, maintain, and navigate a set of directory (path elements) from within the BASH shell environment.

While various other solutions have been provided within the open source community (), the BNT has been designed to provide the ability to navigate and maintain sets of directories using a technique that has the following features:

- 1) Ease-of-use and conceptual simplicity
- 2) Portability across versions of the BASH shell, including the CYGWIN platform
- 3) Compatibility with the standard UNIX idiom of commands, filters, and pipes
- 4) Reusability and extensibility
- 5) Lightweight use of system related resources

Architecture and Design

PATH Variables

The BNT leverages the well-know and understood concept of the PATH (path) environmental variable as found in various UNIX systems. In UNIX systems, PATH is an environment variable that consists of a set of directories separated by a delimiter (the colon, :). When an execution request is made, UNIX uses the PATH (path) environmental variables to find the location of the specified command.

For UNIX systems, path-related environmental variables can be expressed in either of the following Bacchus normal forms (BNF):

```
path ::= {<directory>:}
path ::= <directory1>: <directory2>: ... <directoryi>: ... <directoryn>
```

Various other facilities on UNIX systems make use of the concept of a delimited list of directories or files. These include those associated with shared, dynamic shared object libraries LD_LIBRARY_PATH, LIBPATH, SHLIB_PATH under Linux/Solaris, AIX, and HP-UX, respectively. These variables define the set of directories that will be searched to find the locations from which to load shared libraries upon which various executable programs depend. The various versions of UNIX provide mechanisms that can

be used to overload the standard locations for shared-libraries in order to support use of debug versions during development.

Another example is the class path (CLASSPATH) environment variable used by Java system. The class path variable forms the connection between the Java Runtime Environment (JRE) and the file system. It defines where the Java interpreter (JVM) looks for .class files to load. The basic idea is that the file system hierarchy mirrors the Java package hierarchy, and the class path specifies which directories in the file system serve as roots for the Java package hierarchy.

An important feature of PATH variables on UNIX systems is that they define a highly context specific semantics of usage that impose an ordering on the directory or file elements they maintain. For example, when using the path variable, the first occurrence of the requested command encountered within the PATH structure will be executed. The ordering of the elements with PATH variables effectively define a hierarchical data structure and the process used by the various software components can be modeled using the iterator pattern over their associated data structure.

Fundamentals

The BNT leverages the idea of the delimited list of directories to create, maintain, and utilize a set of directory markers (or favorites). Like the PATH related variables described above, BNT keeps this information within a special-purpose environmental variable (BNTPATH). BNT follows the UNIX idiom of using commands, filters, and pipes to compose new functionalities using existing components.

Components and Functions

Figure 1 presents the architecture of the Bash navigation toolkit. It shows that the contents of the environment variable, and its associated conceptual data structure, mapped to several categories of additional functionality. These categories include

- **Structural commands** – commands that are used to create and maintain the elements of the associated environment variable, and thereby provide capacity to add or remove directory elements to or from the associated data structure
- **Presentation commands** – commands that are used to format and present the contents of the environment variable to the user; they indicate to the user the current state of the directory hierarchy, and provide a context for requesting operations on or use of the information within the associated structure
- **Navigation commands** – commands that enable movement, navigation or iteration over the set of directory elements within the UNIX file system using the data maintained within the environmental variable
- **Miscellaneous commands** – command that provide supporting functionalities such as persisting the current directory hierarchy to a file, reloading or merging the contents of a previously saved hierarchy, or resetting the current set of information maintained within the structure.

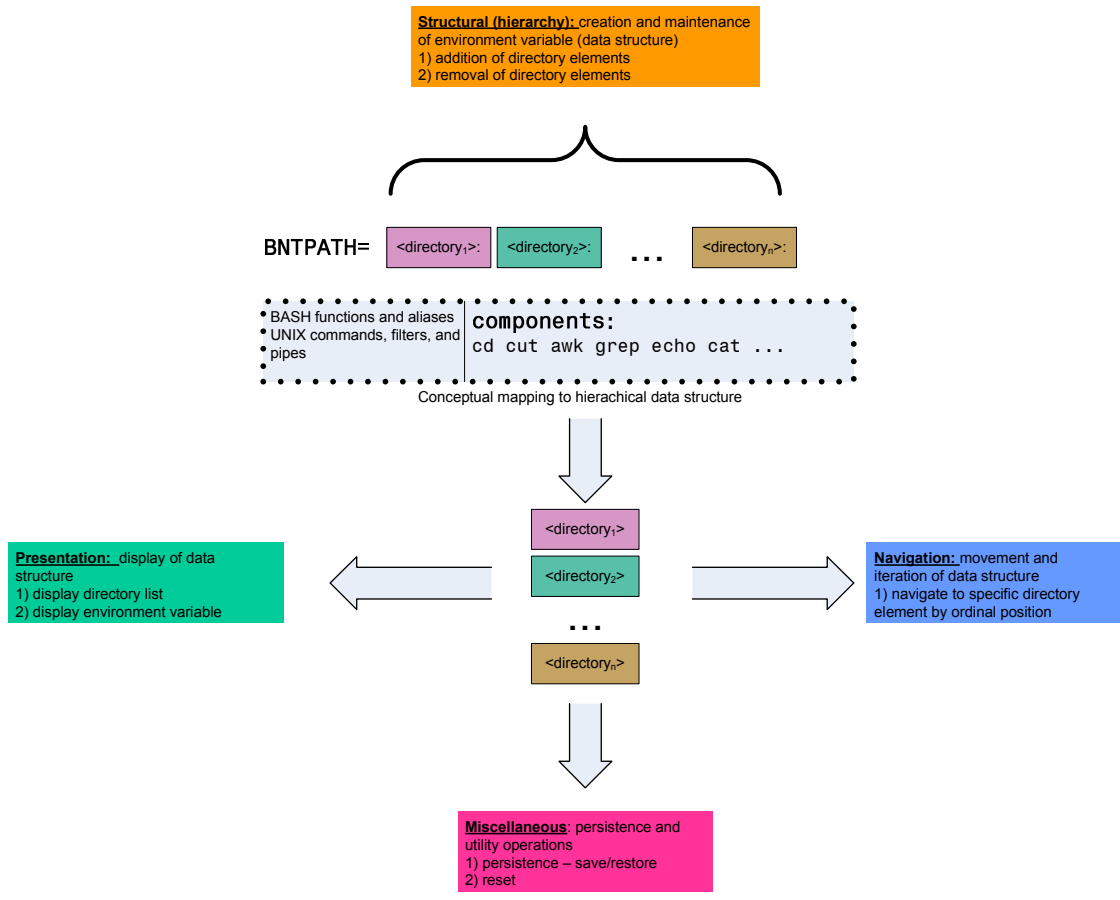


Figure 1: BASH Navigation Toolkit Reference Architecture

Within the BNT reference architecture, all functionality is composed from existing UNIX commands using filters and pipes, or make use of existing capacities of the BASH shell (primarily functions and aliases).

Reference Manual

This section presents a reference manual for the BASH navigation toolkit. It describes how to install and use BNT. In addition, it contains a command summary of the functions and aliases made available for use within the BASH shell. A summary of BNT commands and operations is presented in **Table 1**

Installation and Usage

The Bash Navigation Toolkit is downloaded as a zip file with the following form:

```
bashnavigation-<major version>.<minor-version>.<release number>.zip
```

For example, the initial version of BNT was named `bashnavigation-1.0.0.zip`.

This archive can be expanded using the WINZIP executable on Windows platforms, the unzip command on UNIX platforms, or even the jar tool provided by the JRE. Once expanded the BNT directory structure is organized as shown in **Listing 1**

```
./bnt
./bnt/docs
./bnt/docs/Bash Navigation Toolkit.pdf
./bnt/docs/readme-navigate
./bnt/scripts
./bnt/scripts/.navigate
./bnt/web
./bnt/web/compass.png
./bnt/web/index.html
```

Listing 1 -- BNT Deployment Structure

The main script is the **./bnt/scripts/.navigate** file. This file can be loaded into the user or system profile for BASH (**.bashrc**) using the following command (assuming the toolkit has been installed in the directory **BNTHOME**):

```
source $BNTHOME/bnt/script/.navigate [-q]
```

The optional **-q** (quite) argument suppresses the display of the BNT logo show below:

```
Directory Navigator Version 1.0/19-April-2008
Copyright (c) 2008 -- Richard Landon (rlandon@usa.net). GPL rights reserved
```

Once included in the user environment, the functionalities associated with BNT can be issued by typing the corresponding command within the shell at the prompt.

UNIX File Conversions

If the distribution package is installed on UNIX platforms, then the text files associated with the release will need to be converted from the DOS style line endings. The DOS style line endings are used to insure compatibility on CYGWIN platforms.

The **dos2unix** (or similar) program can be used to convert the line endings from DOS style (0x0d 0x0a) to UNIX style (0x0a).

GNU cut compatibility mode

The BASH navigation toolkit supports a mode of compatible that enables it to utilize the GNU version of the **cut** program. This version of the standard UNIX program **cut** supports the complement mode, which returns the logical complement of the **cut**

operation. For example, compare the output of the two following invocations of GNU `cut`

```
echo mr:fred:flintstone | cut -d: -f 2
fred
```

```
echo mr:fred:flintstone | cut -d: -f 2 --complement
mr:flintstone
```

In other words, the logical complement of the `cut` operation removes the indicated fields from the original data and then returns the remaining data, while the normal mode simply returns the fields selected by the operation.

The BNT uses the logical complement function within several of its commands. In order to support portability to environments that do not have the GNU version of `cut` (and its associated complement mode) the BNT has the ability to emulate the logical complement operation using the standard `awk` and `getopts (3)` facilities. However, for platforms with the GNU compatible version of `cut`, the BNT can be configured to utilize that function. An example of a BASH environment with support for a GNU compatible version of `cut` is the CYGWIN platform (Windows implementations). Many LINUX distributions also provide a GNU compatible version of `cut`.

To enable the use of the GNU `cut` compatibility mode, the environment variable **BNT_GNU_CUTENABLED** should be defined (can contain any value). By default, this value is not defined, so BNT will utilize its internal emulation for the logical complement where appropriate. Setting this variable may have a minor impact on performance of the BNT toolkit.

As mentioned above, this function uses the `getopts` built-in; user scripts also calling this facility should be careful to reset the `OPTIND` environment before processing arguments. Please consult the URL (<http://bugs.centos.org/view.php?id=2803>) for additional information on the use of `getopts` from within shell functions. The interested reader can review the implementation of the internal `bntcut` function for additional details.

Command Summary – Bash Navigation Toolkit, Version 1.0.0

| Category or Type | | | |
|---|----------------|------------------|---|
| Description of command | | | |
| Function | Standard Alias | Usage | Comment |
| Structural: Commands that create, build, modify or maintain the navigation hierarchy | | | |
| <i>Add a directory to the top of the navigation hierarchy</i> | | | |
| sethome | push | push [directory] | Adds the specified directory to the top of the navigation hierarchy. If directory is not specified adds the current directory. The specified directory can be given using either an absolute or relative directory path. Must specify an existing directory, or an error message is produced Returns the name of the specified directory. |
| <i>Remove the first directory from the top of the navigation hierarchy</i> | | | |
| unsethome | pop | pop | Remove the current directory from the top of the navigation structure. Returns the name of directory removed from the top of the navigation structure. |
| Navigation: Commands the support navigation or iteration of the navigation hierarchy | | | |
| <i>Navigate to a specified location with the navigation hierarchy</i> | | | |
| jumphome | jump | jump [ordinal] | Changes the current directory (cd) to the directory corresponding to the ordinal; unspecified will navigate to the first position. If the specified ordinal is larger than the current size of the directory hierarchy, then navigation to the last element within the structure will be performed. Returns the directory corresponding to entry at the indicated position. |

| Category or Type | | | |
|--|----------------|---------------|---|
| Description of command | | | |
| Function | Standard Alias | Usage | Comment |
| Presentation: Commands the display information about the navigation hierarchy | | | |
| <i>List the topology of the navigation hierarchy</i> | | | |
| listhomes | homes | homes | Presents an enumerated list of the content of the current navigation hierarchy. Data is presented in the form <ordinal>:)<directory path>. Example: homes 1:)/bin 2:)/usr/bin 3:)/usr/local/bin 4:)/usr/sbin |
| <i>Display the path (BNTPATH) environmental variable.</i> | | | |
| showhomes | not defined | showhomes | Display the value of the underlying path variable. Example: showhomes Current homes:/bin:/usr/bin:/usr/local/bin:/usr/sbin |
| <i>List the path element at a given position in the BNTPATH</i> | | | |
| showhome | show | show[ordinal] | Display path element at ordinal. If ordinal is not specified, then the first path element is displayed. Example: show 2 /usr/bin |
| Auxiliary | | | |
| Reset the current navigation hierarchy | | | |
| resethome | reset | reset | Resets the current navigation hierarchy. The command reset the contents of the <i>BNTPATH</i> environment variable. |
| Persist (save) current navigation hierarchy to disk (-/.navigaterc) file | | | |

| Category or Type | | | |
|---|----------------|---------|--|
| Description of command | | | |
| Function | Standard Alias | Usage | Comment |
| persisthome | persist | persist | Persists (saves) the current navigation hierarchy to disk. The command writes the contents of the <i>BNTPATH</i> environment variable to the <i>~/ .navigaterc</i> file (in the user home directory). If the file exists, then its contents will be overwritten. |
| Load previously saved navigation hierarchy from disk (<i>~/ .navigaterc</i>) file | | | |
| loadhome | load | load | Loads the contents of <i>BNTPATH</i> variable from the previously saved <i>.navigaterc</i> file (from the user home directory). If the file does not exist, then no action is taken. |
| Remove previously saved navigation hierarchy from disk (<i>~/ .navigaterc</i>) file. | | | |
| purgehome | purge | purge | The command removes (deletes) the <i>.navigaterc</i> file from the user home directory. If the file does not exist, then no action is taken. |
| | | | |

Table 1: BNT Usage Guide and Command Summary