NAME

djinn − UNIX daemon wrapper for Java programs

SYNOPSIS

djinn [**-j** *jvm*] [[**-J** *arg*] ...] [**-p** *classpath*] [**-o** *logfile*] [**--help --version --verbose**] *class* [*arg* ...]

DESCRIPTION

djinn is a simple C wrapper program that allows a Java program to run as a daemon. It invokes the Java Virtual Machine on the named *class* in a subprocess that has no standard I/O streams and no controlling terminal. Any arguments which follow the class name are passed to the **main()** function in the class. This subprocess is created as the leader of its own process group, and its signal mask is modified to ignore the **SIGHUP** signal.

The absolute path to the Java Virtual Machine may be specified on the command line or through the **JAVA** environment variable, with the command line switch taking precedence. If neither of these is specified, **djinn** attempts to locate an executable named "java" in the binary search path, and if successful, uses that as the JVM.

After spawning the JVM, if **djinn** detects that the JVM immediately exited with an error, it writes a message to that effect to standard error; any messages from the JVM itself will be available in the log file, if one has been specified.

OPTIONS

--help

Display a command synopsis and exit.

--version

Display the program's version information and exit.

--verbose

Run in verbose mode.

-j *jvm*

Specify the absolute path to the Java VM executable. If this switch is omitted, **djinn** will check the **JAVA** environment variable, and if that does not contain the path of an executable file, it will attempt to find a program named ‘‘java’’ in the binary search path, and use that as the JVM.

-J *arg*

Pass the argument *arg* to the Java Virtual Machine. If several arguments need to be passed, this switch may be repeated. The argument *arg* should be enclosed in quotes to prevent the shell from interpreting it as a switch to **djinn** itself. If no occurrences of this switch appear in the argument list, **djinn** will check the **JAVA_ARGS** environment variable, and if it is defined, will pass the arguments specified therein to the JVM.

-o *logfile*

Specify an output file; the Java VM's standard output and standard error streams will be redirected into *logfile*. If this switch is omitted, the JVM's output is sent to /dev/null.

-p *classpath*

Specify the *classpath* for the Java Virtual Machine. If this switch is omitted, the value of the **CLASSPATH** environment variable will be used instead.

EXAMPLES

The following example invokes the client JVM on the class **com.foo.Test**, passing the arguments *10* and *Hello* to its **main()** function.

```
djinn -J "-client" -p . com.foo.Test 10 Hello
```

ENVIRONMENT

JAVA	An absolute path to a Java Virtual Machine executable.
JAVA_ARGS	Any additional arguments to pass to the JVM.
CLASSPATH	The classpath for the JVM.
PATH	The binary search path for locating the JVM executable.

NOTES

Java is a trademark of Sun Microsystems, Inc.

AUTHOR

Mark Lindner <markl@gnu.org>