**Parsing Arguments with getopt**

* Normally, getopt is called in a loop. When getopt returns -1, indicating no more options are present, the loop terminates.
* A switch statement is used to dispatch on the return value from getopt. In typical use, each case just sets a variable that is used later in the program.
* A second loop is used to process the remaining non-option arguments.

/\* Process command line options \*/

while ((c = getopt(argc, argv, "c:f:h:H:i:l:p:s:t:x:z:For")) != -1)

{

switch (c)

{

case 'c':

tempMaxConnections = atoi( optarg ) ; /\* Maximum connections/threads allowed \*/

break;

case 'f':

tempThresHold = atof( optarg ) ; /\* Threshold value in seconds where new handlers should be created \*/

break;

case 'F':

ignoreThresHold = TRUE ; /\* Ignore Threshold value, do not fork new handlers \*/

break;

case 'h':

tempMinHandlers = atoi( optarg ) ; /\* Minimum number of handler processes \*/

break;

case 'H':

tempMaxHandlers = atoi( optarg ) ; /\* Maximum number to handler processes \*/

break;

case 'i':

handlerInactiveTimeout = atoi( optarg ) ; /\* Number of seconds a child may be idle before terminating \*/

break;

/\* case 'l':

logDirNode = strdup( optarg ) ;

break; \*/

case 'o':

sharedMemoryOverride = TRUE ;

break;

case 'p':

controlPrefix = strdup( optarg ) ;

break;

case 'r':

redirect = FALSE ;

break;

case 's':

selectTimeout = atol( optarg ) ;

break;

case 't':

handlerTimeout = atol( optarg ) ;

break;

case 'd':

debug\_level = DEBUG\_ON ; /\* Set debug on \*/

break;

/\* case 'x':

possrvrConfigFileName = strdup( optarg ) ;

break; \*/

case 'z':

soapConfigFileName = strdup( optarg ) ;

break;

}

}

<https://linux.die.net/man/3/optarg>

Other Parse command-line options :

* getopt
* getopt\_long
* getopt\_long\_only
* optarg
* optind
* opterr
* optopt

**int getopt(int argc, char \* const argv[],**

**const char \*optstring);**

**extern char \*optarg;**

**extern int optind, opterr, optopt;**

1. The getopt() function parses the command-line arguments. Its arguments argc and argv are the argument count and array as passed to themain() function on program invocation. An element of argv that starts with '-' (and is not exactly "-" or "--") is an option element. The characters of this element (aside from the initial '-') are option characters. If getopt() is called repeatedly, it returns successively each of the option characters from each of the option elements.
2. The variable optind is the index of the next element to be processed in argv. The system initializes this value to 1. The caller can reset it to 1 to restart scanning of the same argv, or when scanning a new argument vector.
3. If getopt() finds another option character, it returns that character, updating the external variable optindand a static variable nextchar so that the next call to getopt() can resume the scan with the following option character or argv-element.
4. If there are no more option characters, getopt() returns -1. Then optind is the index in argv of the first argv-element that is not an option.
5. optstring is a string containing the legitimate option characters. If such a character is followed by a colon, the option requires an argument, so getopt() places a pointer to the following text in the same argv-element, or the text of the following argv-element, in optarg. Two colons mean an option takes an optional arg; if there is text in the current argv-element (i.e., in the same word as the option name itself, for example, "-oarg"), then it is returned in optarg, otherwise optarg is set to zero. This is a GNU extension. If optstring contains W followed by a semicolon, then -W foo is treated as the long option --foo. (The -Woption is reserved by POSIX.2 for implementation extensions.) This behavior is a GNU extension, not available with libraries before glibc 2.

This command asks gcc which **C++** preprocessor it is using, and then asks that preprocessor where it looks for includes.

`gcc -print-prog-name=cc1plus` -v

*ignoring nonexistent directory "NONE/include"*

*ignoring nonexistent directory "/usr/local/lib/gcc/ia64-hp-hpux11.31/4.2.3/../../../../ia64-hp-hpux11.31/include"*

*#include "..." search starts here:*

*#include <...> search starts here:*

*/usr/local/lib/gcc/ia64-hp-hpux11.31/4.2.3/../../../../include/c++/4.2.3*

*/usr/local/lib/gcc/ia64-hp-hpux11.31/4.2.3/../../../../include/c++/4.2.3/ia64-hp-hpux11.31*

*/usr/local/lib/gcc/ia64-hp-hpux11.31/4.2.3/../../../../include/c++/4.2.3/backward*

*/usr/local/include*

*/usr/local/lib/gcc/ia64-hp-hpux11.31/4.2.3/include*

*/usr/include*

*End of search list.*