

# GNU Autoconf, Automake, and Libtool

## 1-57870-190-2

Gary V. Vaughan, Ben Elliston, Tom Tromey, and Ian Lance Taylor

*Copyright © 2001 by New Riders Publishing*

**Warning and Disclaimer:** Every effort has been made to make this book as complete and accurate as possible, but no warranty or fitness is implied. The information is provided on an as-is basis. The authors and New Riders Publishing shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

| <b>Misprint</b>  | <b>Correction</b>   |
|--|---|
| <p><b>Page xx</b></p> <ul style="list-style-type: none"> <li>A momospaced (Courier) font is used for commands, command-line options, functios, and all code examples.</li> </ul>   | <ul style="list-style-type: none"> <li>A monospaced (Mono) font is used for commands, command-line options, functios, and all code examples.</li> </ul>   |
| <p><b>Page xx</b></p> <ul style="list-style-type: none"> <li>Bold is used for the emphasis of terms, interactive shell log lines that are typed by the user, and function names.</li> </ul>  | <ul style="list-style-type: none"> <li>Bold is used for the emphasis of terms and function names.</li> </ul>  |
| <p><b>Page 20</b></p> <p>They represent the names of the target and the first dependency for the rule in which they appear.</p>  | <p>They represent the names of the target and the first dependency for the rule in which they appear. <code>\$\$</code> is available in any rule, but for some version of make <code>\$\$</code> is only available in suffix rules.</p> |
| <p><b>Page 20</b></p> <pre>@echo "\$@ depends on \$&lt;"</pre>   | <pre>@echo "\$@ depends on dummy"</pre>   |
| <p><b>Page 22</b></p> <p>The source files used to build 'foonly' are the C source files 'main.c', 'foo.c', and 'foo.h';...</p>   | <p>The source files used to build 'foonly' are the C source files 'nly.c', 'main.c', 'foo.c', and 'foo.h';...</p>   |
| <p><b>Page 54</b></p> <pre>\$Xsed -e 's\.[^\.]*/.\\$U&amp;/g;s\.[^\.]*/.\\$U&amp;/'</pre>  | <pre>[\$Xsed -e 's/\.[^\.]*/.\\$U&amp;/g;s\.[^\.]*/.\\$U&amp;/']</pre>  |
| <p><b>Page 67-Missing paragraph after Autoconf macros</b></p> <p>Recall the use of <code>bzero</code> in the Memory Management subsection of section 7.2.1 is not entirely portable. The trick is to provide a <code>bzero</code> work-alike, depending on which funtions Autoconf detects. Add the following towards the end of <code>configure.in</code>:</p> <pre>... AC_CHECK_FUNCS(bzero memset, break) ...</pre> <p>With the addition of this small snippet of code to <code>common.h</code>, I can now make use of <code>bzero</code> even when linking with a C library that has no implementation of its own:</p> <pre>#if !HAVE_BZERO &amp;&amp; HAVE_MEMSET #define bzero(buf, bytes)      ((void) memset (buf, 0, bytes)) #endif</pre> |   |
| <p><b>Page 83-Terminology section made clearer</b></p> <p>The sheer number of uses of the word "library" in this book could be easily ver confusing. In this chapter and throughout the rest of the book, I will refer to various kinds of libraries as follows:</p> <p><i>native</i>—Low level libraries, that is, libraries</p>  |   |

|   |  |
|---|--|
| <p>provided by the host architecture.</p> <p><i>Libtool library</i>—The kind of library built by Libtool. This encompasses both the shared and static native components of the implementation of the named library.</p> <p><i>pseudo-library</i>—The level .la file produced by Libtool. The “pseudo-library” is not a library in its own right, but is treated as if it were from outside the Libtool interface.</p> <p>Furthermore, in context of Libtool, there is another subtle (but important) distinction to be drawn:</p> <p><i>static library</i>—A Libtool library which has no shared archive component.</p> <p><i>static archive</i>—The static component of a Libtool library.</p> |  |
| <p><b>Page 112</b><br/> <code>LTLIBOBJS='echo X"\$LIBOBJS"   \</code><br/> <code>[\$Xsed -e "s,\.[^.]* ,.lo,</code><br/> <code>g;s,\.[^.]*\$, .lo," '</code></p>  | <code>LTLIBOBJS='echo X"\$LIBOBJS"   \</code><br><code>[\$Xsed -e "s,\.[^.]* ,.lo,</code><br><code>g;s,\.[^.]*\$, .lo," '</code>   |
| <p><b>Page 112</b><br/> <code>LTALLOCA='echo X"\$ALLOCA"  \$Xsed -e "s,\.</code><br/> <code>[^.]* ,.lo, g;s,\.[^.]*\$, .lo," '</code></p>   | <code>LTALLOCA='echo X"\$ALLOCA"   [\$Xsed -e "s,\.</code><br><code>[^.]* ,.lo, g;s,\.[^.]*\$, .lo," '</code>  |
| <p><b>Page 127</b><br/> <code>LTALLOCA= 'echo X"\$ALLOCA"  \$Xsed -e</code><br/> <code>"s,\.[^.]*\$, .lo"</code></p>  | <code>LTLIBOBJS= 'echo X"\$LIBOBJS"  \$Xsed -e</code><br><code>"s,\.[^.]*\$, .lo"</code>   |
| <p><b>Page 134</b><br/> <code>LTLIBOBJS=echo X"\$LIBOBJS"   \</code><br/> <code>sed s,^x,;,s,\.[^.]* ,.lo</code><br/> <code>,g;s,\.[^.]*\$, .lo;'</code><br/> <code>AC_SUBST(LTLIBOBJS)</code></p>  | <code>Xsed="sed -e s/^X//":</code><br><code>LTLIBOBJS='echo X"\$LIBOBJS"   \</code><br><code>[\$Xsed -e "s,\.[^.]* ,.lo</code><br><code>,g;s,\.[^.]*\$, .lo," '</code>                   |
| <p><b>Page 143</b><br/> <code>echo Making \$ in \$(docdir)</code><br/> <code>cd \$(docdir) &amp;&amp; make \$</code></p>  | <code>echo Making \$@ in \$(docdir)</code><br><code>cd \$(docdir) &amp;&amp; make \$@</code>   |
| <p><b>Page 202</b><br/> <code>\$ gcc -fPIC -c simple-module.c</code><br/> <code>\$ gcc -shareed -o simple-module.so simple-</code><br/> <code>module.o</code></p>   | <code>\$ gcc -fPIC -c simple-module.c</code><br><code>\$ gcc -shareed -o simple-module.so simple-</code><br><code>module.o</code>  |
| <p><b>Page 268</b><br/> <code>\$ ls /usr/bin/*   wc -l</code><br/> <code>sh: error: line too long</code></p>  | <code>\$ ls -d /usr/bin/*   wc -l</code><br><code>sh: error: line too long</code>  |
| <p><b>Page 271</b><br/> <code>for dir</code><br/> <code>do</code><br/> <code>shell="\$dir/\$cmd"</code><br/> <code>done</code></p>  | <code>for dir</code><br><code>do</code><br><code>shell="\$dir/\$cmd"</code><br><code>done</code>   |
| <p><b>Page 276</b><br/> Realistically, no one is going to have '-x', for example, as the first element of his 'PATH' variable, so the 'dummy' could be omitted—as I did earlier in the script in Section 21.2.3.</p>  | Realistically, no one is going to have '-x', for example, as the first element of his 'PATH' variable, so the 'dummy' could be omitted—as I did earlier in the script in Section 21.2.4. |
| <p><b>Page 290—Missing sentence at the end of first paragraph.</b><br/> Note that the name of the cache variable must contain '_cv_' in order to be saved correctly.</p>  |  |
| <p><b>Page 290</b><br/> You can find the macro archive at <a href="http://peti.cys.de/autoconf-archive/">http://peti.cys.de/autoconf-archive/</a>.</p>  | You can find the macro archive at <a href="http://www.gnu.org/software/ac-archive">http://www.gnu.org/software/ac-archive</a> .  |
| <p><b>Page 316</b><br/> If you don't use Libtool, you must manually call the Libtool macro, 'AC_EXEEXT' in you 'configure.in', to make sure that it is initialized correctly.</p>   | If you don't use Libtool, you must manually call the Autoconf macro, 'AC_EXEEXT' in you 'configure.in', to make sure that it is initialized correctly.                                   |

This errata sheet is intended to provide updated technical information. Spelling and grammar misprints are updated during the reprint process, but are not listed on this errata sheet.