

Essential Skills (4) Team 1

Sharifullin Emil, Tykushin Anatoly, Vaneev Bogdan, SNE

September 30, 2016

Contents

1	GNU Automake	2
1.1	Autoheader	3
1.2	GNU Libtool	3
1.3	Steps involved in creating a software	3

List of Figures

1	GNU Automake procedure	2
---	----------------------------------	---

List of Tables

1 GNU Automake

- helps to create portable Makefiles for use with the make utility (input *Makefile.am*; output *Makefile.in*)
- *Makefile.in* is then used by the configure script to generate a *Makefile* [?]

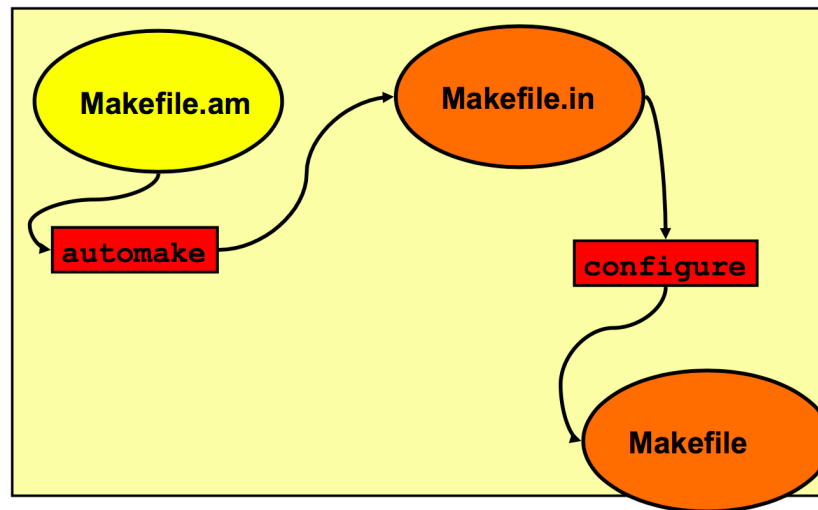


Figure 1: GNU Automake procedure

Usage example (as shown on the figure 1 Our package builds a program sne, simply add the following to the *Makefile.am* in the directory where the program is Built.

Example File *foo.c*

```
#include <stdio.h>
main()
{
    printf("Cum grano salis \n");
}
```

File: *Makefile.am*

```
bin_PROGRAMS = foo
foo_SOURCES = foo.c
```

File: *configure.ac*

```
AC_INIT(foo.c)
AM_INIT_AUTOMAKE(latin_words, 0.9)
AC_PROG_CC
AC_HEADER_STDC
AC_PROG_INSTALL
AC_OUTPUT([Makefile])
```

Where

AC-INIT(sourcefile) Initializes autoconf, should be the first macro [?]

AM-INIT-AUTOMAKE(latin-words, 0.9) Runs many macros required for proper operation of the generated Makefiles

AC-PROG-CC Determines a C compiler to use, sets the CC variable, initializes the CFLAGS

AC-HEADER-STDC Checks for *stdlib.h*, *stdarg.h*, *string.h* and *float.h*, defines STDC-HEADERS on success

AC-PROG-INSTALL Set variable INSTALL to the path of a BSD-compatible install program

AC-OUTPUT([Makefile]) Create output files in the case the Makefile.

1.1 Autoheader

- Create a template header for configure
- Usage ? run autoheader on a directory with a *configure.ac* that contains a AC-CONFIG-HEADER macro call, and it will write the *configure.in* file

1.2 GNU Libtool

- helps manage the creation of static and dynamic libraries on various Unix-like operating systems
- hides the complexity of using shared libraries on different [?]

Platforms - provide a generic interface for developers

clean Remove files from the build directory

compile Compile a source file into a libtool object.

execute Automatically set the library path, then run a program.

finish Complete the installation of libtool libraries.

install Install libraries or executables.

link Create a library or an executable.

uninstall Remove libraries from an installed directory.

1.3 Steps involved in creating a software

1. create source
2. create configure.ac
3. create makefile.am
4. aclocal
5. autoconf
6. automake -a //to add missing files

References

- [1] Autoconf. *GNU Operating system site*, "<http://www.gnu.org/software/autoconf/autoconf.html>", -1, 2016.
- [2] Automake. *GNU Operating System site*, "<http://www.gnu.org/software/automake/automake.html>", -1, 2016.
- [3] Libtool. *GNU Operating System site*, "<http://www.gnu.org/software/libtool/libtool.html>", -1, 2016.