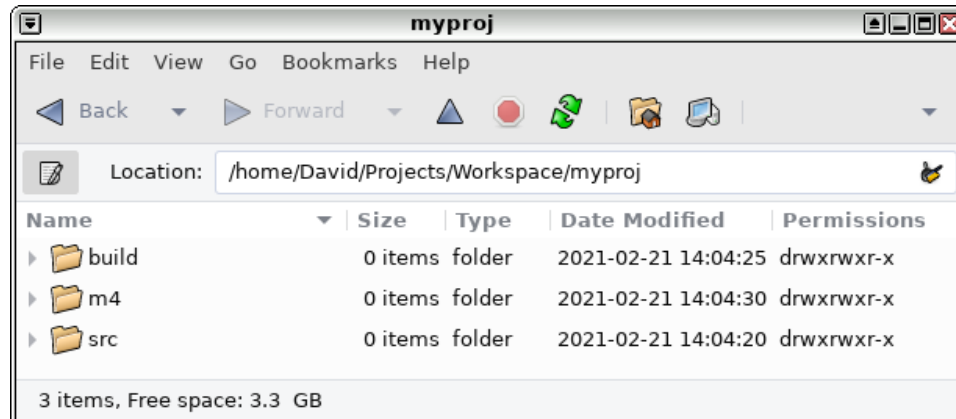


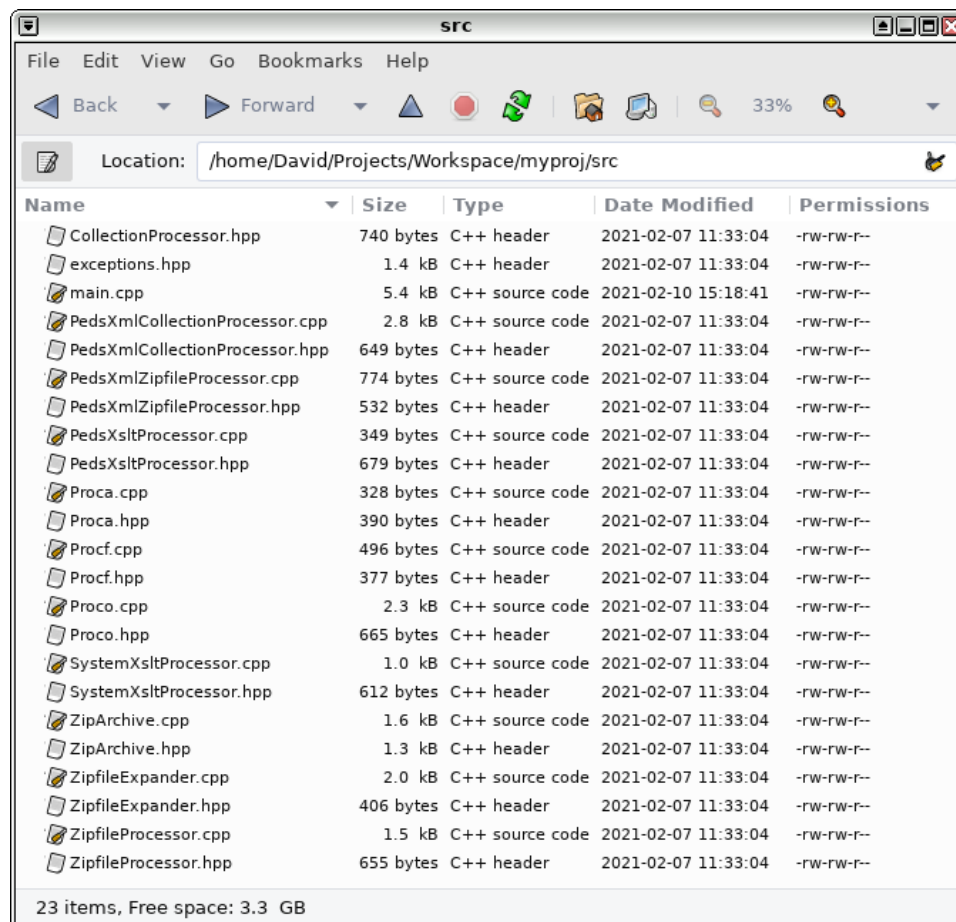
Create a GNU Autotools C/C++ Project

From Existing C/C++ Code or From Scratch

1. Create a project directory (*myproj* in this guide) containing *src*, *build*, and *m4* directories, and any additional desired directories such as *debug*, *test*, etc.:



2. Copy source code files into (or create them in) *src*:



3. Create a *Makefile.am* file listing source files in the project directory:

```
David@Fedora64:~/Projects/Workspace/myproj - /usr/bin/vim Makefile.am
File Edit View Search Terminal Help
bin_PROGRAMS = myproj

myproj_SOURCES = \
    src/CollectionProcessor.hpp \
    src/exceptions.hpp \
    src/main.cpp \
    src/PedsXmlCollectionProcessor.cpp \
    src/PedsXmlCollectionProcessor.hpp \
    src/PedsXmlZipfileProcessor.cpp \
    src/PedsXmlZipfileProcessor.hpp \
    src/PedsXsltProcessor.cpp \
    src/PedsXsltProcessor.hpp \
    src/Proca.cpp \
    src/Proca.hpp \
    src/Procf.cpp \
    src/Procf.hpp \
    src/Proco.cpp \
    src/Proco.hpp \
    src/SystemXsltProcessor.cpp \
    src/SystemXsltProcessor.hpp \
    src/ZipArchive.cpp \
    src/ZipArchive.hpp \
    src/ZipfileExpander.cpp \
    src/ZipfileExpander.hpp \
    src/ZipfileProcessor.cpp \
    src/ZipfileProcessor.hpp

~
~
```

4. Add preprocessor and library loading flags to *Makefile.am* as needed:

```
David@Fedora64:~/Projects/Workspace/myproj - /usr/bin/vim Makefile.am
File Edit View Search Terminal Help

myproj_CPPFLAGS = \
    -I/usr/include/libxml2

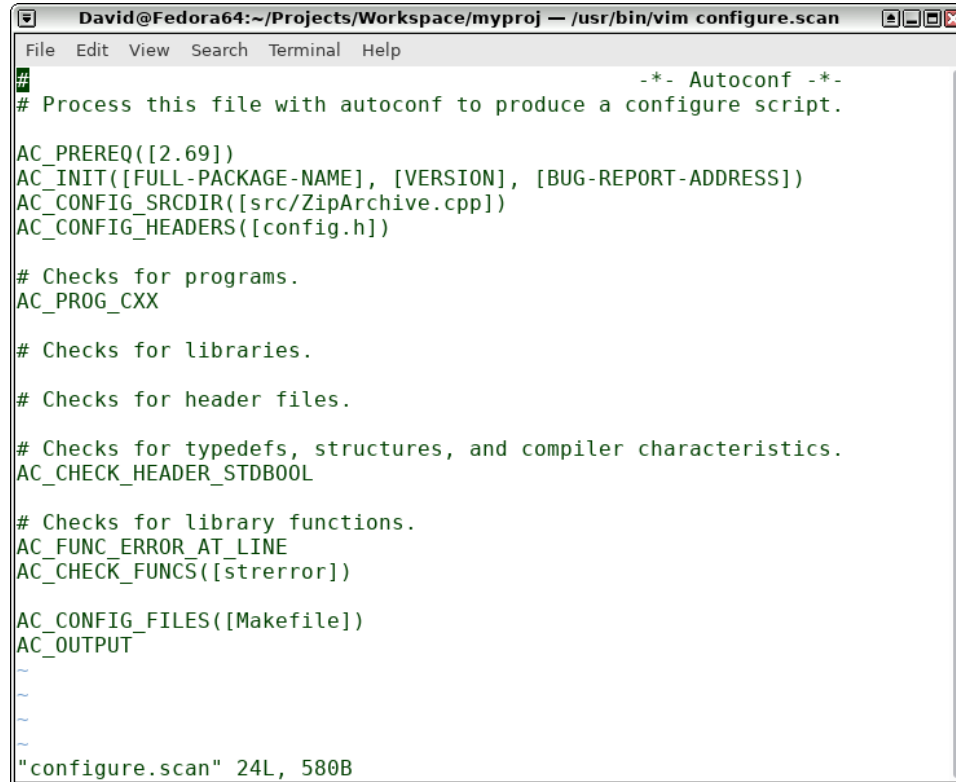
myproj_LDFLAGS = \
    -lboost_filesystem \
    -lboost_program_options\
    -lzip \
    -lxslt \
    -lxml2
```

- i. Use *pkg-check* command, where possible, to get the necessary flags.
 - ii. Where libraries (e.g. Boost libraries) don't support *pkg-check*, manual determination of the flags is required.
 - iii. Variables defined in *configure.ac* or elsewhere to the extent possible may be used to provide values for *x_CPPFLAGS* and *x_LDFLAGS*. See Bibliography items 2 and 3.
5. Add nonstandard directories and object installation directives to *Makefile.am* and (create any directories or objects within the project directory) as necessary. See Bibliography item 4.

```
David@Fedora64:~/Projects/Workspace/myproj - /usr/bin/vim Makefile.am
File Edit View Search Terminal Help

myprojdatadir = $(datadir)/myproj
dist_myprojdata_DATA = share/myproj/peds.xsl
```

6. Run *autoscan* in the project directory (i.e. *myproj*) to create a *configure.scan* configure script template (log file *autoscan.log*, also created, isn't necessary and can be deleted):



```
David@Fedora64:~/Projects/Workspace/myproj - /usr/bin/vim configure.scan
File Edit View Search Terminal Help
# Process this file with autoconf to produce a configure script.

AC_PREREQ([2.69])
AC_INIT([FULL-PACKAGE-NAME], [VERSION], [BUG-REPORT-ADDRESS])
AC_CONFIG_SRCDIR([src/ZipArchive.cpp])
AC_CONFIG_HEADERS([config.h])

# Checks for programs.
AC_PROG_CXX

# Checks for libraries.

# Checks for header files.

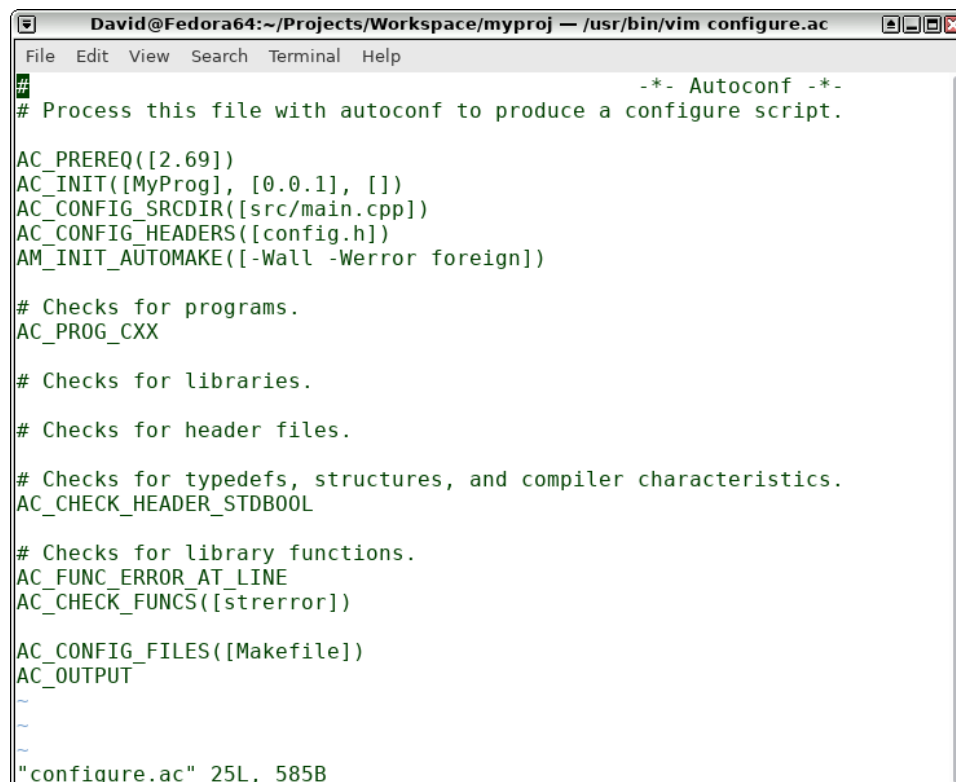
# Checks for typedefs, structures, and compiler characteristics.
AC_CHECK_HEADER_STDBOOL

# Checks for library functions.
AC_FUNC_ERROR_AT_LINE
AC_CHECK_FUNCS([strerror])

AC_CONFIG_FILES([Makefile])
AC_OUTPUT

~
~
~
"configure.scan" 24L, 580B
```

7. Rename *configure.scan* to *configure.ac* (i.e. execute `mv configure.scan configure.ac`).
8. Customize *configure.ac* as needed:



```
David@Fedora64:~/Projects/Workspace/myproj - /usr/bin/vim configure.ac
File Edit View Search Terminal Help
# Process this file with autoconf to produce a configure script.

AC_PREREQ([2.69])
AC_INIT([MyProg], [0.0.1], [])
AC_CONFIG_SRCDIR([src/main.cpp])
AC_CONFIG_HEADERS([config.h])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])

# Checks for programs.
AC_PROG_CXX

# Checks for libraries.

# Checks for header files.

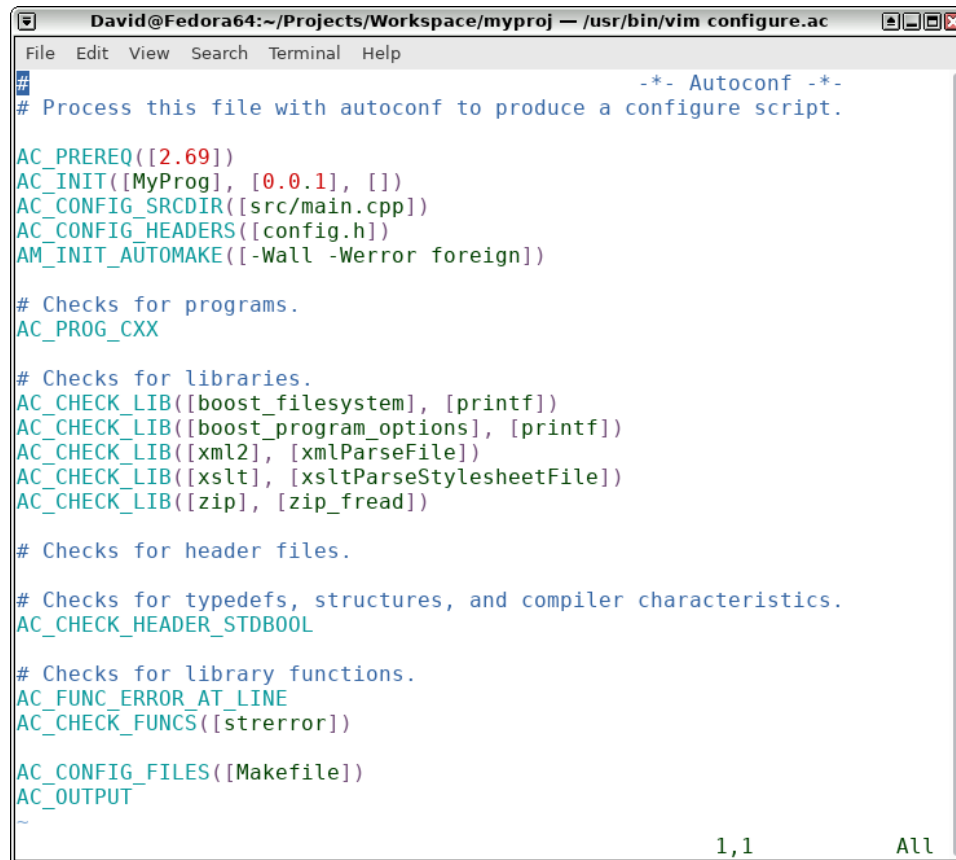
# Checks for typedefs, structures, and compiler characteristics.
AC_CHECK_HEADER_STDBOOL

# Checks for library functions.
AC_FUNC_ERROR_AT_LINE
AC_CHECK_FUNCS([strerror])

AC_CONFIG_FILES([Makefile])
AC_OUTPUT

~
~
~
"configure.ac" 25L, 585B
```

9. Add checks for libraries and header files to *configure.ac* where desired:



```
David@Fedora64:~/Projects/Workspace/myproj - /usr/bin/vim configure.ac
File Edit View Search Terminal Help
#                                     -*- Autoconf -*-
# Process this file with autoconf to produce a configure script.

AC_PREREQ([2.69])
AC_INIT([MyProg], [0.0.1], [])
AC_CONFIG_SRCDIR([src/main.cpp])
AC_CONFIG_HEADERS([config.h])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])

# Checks for programs.
AC_PROG_CXX

# Checks for libraries.
AC_CHECK_LIB([boost_filesystem], [printf])
AC_CHECK_LIB([boost_program_options], [printf])
AC_CHECK_LIB([xml2], [xmlParseFile])
AC_CHECK_LIB([xslt], [xsltParseStylesheetFile])
AC_CHECK_LIB([zip], [zip_fread])

# Checks for header files.

# Checks for typedefs, structures, and compiler characteristics.
AC_CHECK_HEADER_STDBOOL

# Checks for library functions.
AC_FUNC_ERROR_AT_LINE
AC_CHECK_FUNCS([strerror])

AC_CONFIG_FILES([Makefile])
AC_OUTPUT
~
1,1 All
```

10. Run *autoreconf -vi* to generate a *configure* file.

- i. If a warning is generated indicating that a source file "is in a subdirectory, but option 'subdir-objects' is disabled", add the following line at the top of *Makefile.am*:

`AUTOMAKE_OPTIONS = subdir-objects`

11. To build the project and thereby create a *Makefile*, run *configure* from the *build* directory (so that files generated during the build are kept separate from the project directory, just to keep things neat):

`$ cd build`
`$../configure`

12. To create an executable, run *make* in the *build* directory.

`$ make`

13. Run *autoreconf -vi* again after subsequent modification of *Makefile.am* and/or *configure.ac*.

Bibliography

1. "Creating a simple Autotools based project", <https://www.youtube.com/watch?v=dc1kEJvS248>
2. "Adding GTK to an Autotools Project", <https://www.youtube.com/watch?v=TDGwgz1e710>
3. "PKG_CHECK_MODULES considered harmful?",
<https://stackoverflow.com/questions/10220946/pkg-check-modules-considered-harmful>
4. *Automake Manual* (for GNU Automake version 1.16.3, 19 November 2020), "2.2.3 Standard Directory Variables" and "3.3 The Uniform Naming Scheme", Copyright © 1995–2020 Free Software Foundation, Inc.,
<https://www.gnu.org/software/automake/manual/automake.html#Uniform>
5. "Detecting C++ Libraries with Autotools", <https://nerdland.net/2009/07/detecting-c-libraries-with-autotools/>
6. *Autoconf Manual* (for GNU Autoconf version 2.67, 21 July 2010), Copyright © 1992, 1993, 1994, 1995, 1996, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 Free Software Foundation, Inc., https://www.gnu.org/software/autoconf/manual/autoconf-2.67/html_node/index.html