

RFC: New Autotools Behavior

Dana Robinson

For HDF5 1.8.15 (release date: May 2015), the autotools have been overhauled. As a part of this work, changes have been made to the autotools files that are under version control. How and when the generated files are created has also changed.

This document describes the changes that have been made to how the autotools are used in HDF5.

Note that this document does NOT describe the specific changes to the `configure.ac` and `Makefile.am` files. These will be described in the HDF5 1.8.15 `RELEASE.txt` file.

Introduction

Running the `bin/reconfigure` when working with the HDF5 source currently requires particular versions of each autotool to be used. If not working on THG machines, developers also have to set several environment variables to point to local tools. These restrictions severely limit the ability to perform routine tasks such as adding new files to the source code or modifying configuration files, particularly when working off-site.

Fixing `bin/reconfigure`

The `bin/reconfigure` script can easily be modified to use the local system's autotools. As a part of this, we should also move to using an `autogen.sh` script, which has become the standard way of handling any autotools processing after checking out code from version control.

Conflicts in Generated Files

Unfortunately, allowing a wider range of the autotools to be run by developers means that there will be churn in the generated files due to slight variations in the tools' output. Most organizations handle this by not checking autotools output files into version control. Instead, developers are expected to run the `autogen.sh` script to create these files after each checkout.

`autogen.sh`

A new file named `autogen.sh` will be created in the source root. This file will run the individual autotools and (optionally) the source/API processing scripts (like `make_err`). It will be very similar in spirit to the existing `bin/reconfigure`, but with some changes:

- By default, the script will use the system's autotools and will not check that particular versions of the autotools are being used.
-

- The script will take an option (-p | --production) that will use specific versions of the autotools. This is intended for use by The HDF Group and includes hard-coded paths to THG machines. It will not be useful for non-THG developers.
- The script will take an option (-s | --process_source) that will run the source/API processing scripts.
- The environment variables used by bin/reconfigure to specify autotool paths will be prepended with HDF5_ (e.g.: HDF5_ACLOCAL). These will be checked and used if set whether in production mode or not.

Version Control and Source Distributions

Two classes of autotools files need to be considered:

- Generated files like configure and Makefile.in
- Copied m4 files

Non-Release Branches

Generated and copied m4 files will be removed from version control in non-release branches. These files will be regenerated when developers run autogen.sh after checking out the code.

Release Branches

In release branches (e.g.: hdf5/branches/hdf5_1_8_15), generated and copied m4 files will be checked into version control. These files will use the canonical HDF Group versions of the autotools. This is so that we have a record of what went into the source distribution as well as being convenient for users who would like to obtain HDF5 source code from version control.

Source Distributions

Tarballs and zipped files containing HDF5 source code will contain generated files and the copied m4 scripts. Users should not have to have the autotools installed in order to build HDF5.

Workflow Changes

Users

- Users will not have to do anything special when they build from source distributions or check out code from release branches.
- Users who check out code from non-release branches will have to follow the 'developers' changes, below.

Developers

- Development machines will need to have relatively recent versions of the autotools available.
- Developers will have to run `autogen.sh` after checking out source code from non-release branches.

Release Team

- The release team will have to run '`autogen.sh -p`' (production mode) after creating a release branch and check the resulting files into version control. It would be best to create a script to perform this task.
- Source distributions from non-release branches should be created after running '`autogen.sh -p`'.

Acknowledgements

This work was internally funded by The HDF Group.

Revision History

February 2, 2015: Version 1 circulated for comment within The HDF Group.

1)