Tips for contributors

---------------------

\* Please \*\*make pull requests against `master`\*\*.

\* If you're making \*\*changes to files for which the Travis build is not

relevant\*\*, please \*\*add `[ci skip]` to the end of the commit message\*\*.

\* Report bugs using [GitHub Issues].

GLOSSARY

----

Developing the Ledger software uses a number different tools, not all of

which will be familiar to all developers.

\*\*[Boost]\*\*: a standard set of C++ libraries. Most

Boost libraries consist of inline functions and templates in header files.

\*\*[Boost.Python]\*\*: C++ library which enables seamless interoperability

between C++ and the Python programming language.

\*\*[Cheetah]\*\*: a Python templating engine, used by `./python/server.py`.

\*\*[CMake]\*\*: A cross platform system for building from source code. It uses

the `CMakeLists.txt` files.

\*\*[Doxygen]\*\*: generates programming documentation from

source code files. Primarily used on C++ sources, but works on all. Uses

the `doc/Doxyfile.in` file.

\*\*[GCC]\*\*: Gnu Compiler Collection, which includes the

\*gcc\* compiler and \*gcov\* coverage/profiler tool.

\*\*[clang]\*\*: C language family frontend for LLVM, which

includes the \*clang\* compiler.

\*\*[GMP]\*\*: Gnu Multiple Precision Arithmetic Library

provides arbitrary precision math.

\*\*[MPFR]\*\*: Gnu Multiple Precision Floating-point Library

with correct rounding.

\*\*[Markdown]\*\*: A typesetter

format that produces \*html\* files from \*.md\* files. Note that GitHub

automatically renders \*.md\* files.

\*\*[SHA1]\*\*: a marginally secure cryptographic hash function, used only for

signing the license file.

\*\*[Texinfo]\*\*: Gnu documentation

typesetter that produces \*html\* and \*pdf\* files from the `doc/\\*.texi` files.

\*\*[Travis CI]\*\*: a hosted continuous integration

service that builds and runs tests each commit posted to GitHub. Each

build creates a [log], updates a [small badge] at

the top left of the main project's

[README.md], and

emails the author of the commit if any tests fail.

\*\*[utfcpp]\*\*: a library for handling utf-8 in a variety of C++ versions.

Orientation

---

The source tree can be confusing to a new developer. Here is a selective

orientation:

\*\*./acprep\*\*: a custom thousand-line script to install dependencies, grab

updates, and build. It also creates `\\*.cmake`,

`./CmakeFiles/` and other CMake temporary files. Use `./acprep --help`

for more information.

\*\*./README.md\*\*: user readme file in markdown format, also used as the project

description on GitHub.

\*\*./contrib/\*\*: contributed scripts of random quality and completion. They

usually require editing to run.

\*\*./doc/\*\*: documentation, licenses, and

tools for generating documents such as the \*pdf\* manual.

\*\*./lib/\*\*: a couple of libraries used in development.

\*\*./python/\*\*: samples using the Python ledger module.

\*\*./src/\*\*: the C++ header and source files in a flat directory.

\*\*./test/\*\*: a testing harness with subdirectories full of tests

\*\*./tools/\*\*: an accretion of tools, mostly small scripts, to aid development

Building

---

If you are going to be working on Ledger, you'll want to enable both debug

builds (which are the default, using `acprep`), and also the use of

pre-compiled headers. To do this, specify your compiler as either `clang++`

or `g++` as follows:

mkdir build

./acprep --compiler=clang++

cd build

make

This will set up a debug build using clang++ (and pre-compiled headers, which

is enabled by the combination of those two), and then start a build.

For even quicker rebuilds, try the Ninja build tool, which is very fast at

determining what to rebuild, and automatically takes advantage of multiple

cores:

mkdir build

./acprep --compiler=clang++ --ninja

cd build

ninja

[Boost]: http://boost.org

[Boost.Python]: http://www.boost.org/libs/python/

[GitHub Issues]: https://github.com/ledger/ledger/issues

[GMP]: http://gmplib.org/

[MPFR]: http://www.mpfr.org/

[Cheetah]: http://www.cheetahtemplate.org

[CMake]: http://www.cmake.org

[Doxygen]: http://doxygen.org

[Markdown]: https://daringfireball.net/projects/markdown/

[SHA1]: http://en.wikipedia.org/wiki/SHA-1

[Texinfo]: http://www.gnu.org/software/texinfo/

[Travis CI]: https://travis-ci.org

[GCC]: http://gcc.gnu.org

[utfcpp]: http://utfcpp.sourceforge.net

[log]: https://travis-ci.org/ledger/ledger

[small badge]: https://img.shields.io/travis/ledger/ledger/master.svg?&style=flat

[git-flow]: http://nvie.com/posts/a-successful-git-branching-model/

[README.md]: https://github.com/ledger/ledger/blob/master/README.md

[clang]: http://clang.llvm.org