POCO

- Has everything
-
  - forms, etc.
-
  - database access
-
  - interprocess communication (process, pipes)
- In Debian, but old
- Latest version in unstable is still 1.3.6. Experimental has 1.4.6. Current is 1.7.2.
- Example web server at http://pocoproject.org/docs/00100-GuidedTour.html#5
- Annoying Java style
- Allows you to send to a stream
- No http2 support
- C++11 support
- Version in Debian (1.3.6) does not work with C++11?: http://stackoverflow.com/a/14680628/1446838
- But "Hello Server" works fine.
- aborts if using lots of threads :(

[Proxygen] (https://github.com/facebook/proxygen)

- Not in Debian
- Fancy
- Lots of dependencies (folly etc.)
- Building it fetches things from the net and takes 2GB of ram.
- Looks easy enough to program
- Only validated on Ubuntu 14.04
- CLA for contributions
- Fixed a memory leak recently (Mar 22, 2016) :(

[H20] (https://h2o.examp1e.net/index.html)

- http 1.1 and 2
- Not in Debian
- Claims to be fast
- Advertised mostly as a standalone webserver. Library is secondary and hard to build.
- Weird behavior where POST's are converted into GET's
-
  - https://github.com/h2o/h2o/issues/566
-
  - Not fatal. It can be worked around, but it is annoying.

[Boost Asio]
(http://www.boost.org/doc/libs/1_55_0/doc/html/boost_asio/examples/cpp11_examples.html#boost_asio.examples.cpp11_examples.http_server)

- Simple things are not simple
- Maybe not fast?
-
  - http://comments.gmane.org/gmane.comp.lib.boost.asio.user/5133
-
  - http://cmeerw.org/blog/751.html#751
-
  - http://cmeerw.org/blog/748.html#748

[Mongoose] (https://github.com/cesanta/mongoose)

- CLA for contributions? It says it is a license, not a copyright grant.
- Feels more like a commercial project.
- Actively developed

- Not in Debian

[CivetWeb] (https://github.com/civetweb/civetweb)

- MIT licensed fork of Mongoose
- Actively developed
- Written in C with C++ bindings
- Not in Debian

[libmicrohttpd] (http://www.gnu.org/software/libmicrohttpd/)

- C library
- Actively developed.
- Active mailing list
- Claims high performance
- Supports iterating through POST's.
- In Debian

[libhttpserver] (https://github.com/etr/libhttpserver)

- C++ library built on libmicrohttpd
- Actively developed
- Not in debian, but has a debian/rules file which does not work :(
- THREAD_PER_CONNECTION works better than INTERNAL_SELECT for lots of long simultaneous connections

[Simple Web Server] (https://github.com/eidheim/Simple-Web-Server)

- Built on Boost::Asio
- C++11
- More complicated?
- Very new
- Allows me to send to a stream
- May need to increase default stack size
- Will need to add a way to limit request size
- Not very fast. Takes 140 s to satisfy 1024 requests with 128 requests/second that each last 10 s.

[libhttpserver] (https://github.com/RipcordSoftware/libhttpserver)

- C++11
- built on boost::asio
- Example server is really simple and works ok
- Documentation is mostly doxygen
- No way to limit request size: https://github.com/RipcordSoftware/libhttpserver/issues/11

[nxweb] (https://bitbucket.org/yarosla/nxweb/wiki/Home)

- Written in C
- Claims high performance: https://bitbucket.org/yarosla/nxweb/wiki/Benchmarks
- Not in Debian. Claims to be easy to build in Debian
- Actively developed
- No documentation

[Mimosa] (https://github.com/abique/mimosa)

[Kore] (https://kore.io/)

- C library

[libevent web server example] (http://kukuruku.co/hub/cpp/lightweight-http-server-in-less-than-40-lines-on-libevent-and-c-11)

- Not multithreaded?

[libapache2-mod-raii] (http://blackmilk.fr/www/cms/dev/libapache2_mod_raii_en)

- Apache module to directly link in C++ code to Apache.

[Webem] (http://www.codeproject.com/Articles/29290/A-C-Embedded-Web-Server)

- More like a simple Wt. So has templates and such.

[Served] (https://github.com/datasift/served)

- Looks simple enough to use
- based on boost::asio
- Requires Boost >= 1.56
- Requires RE2
- Does not support chunks

[Casablanca (C++ Rest SDK)] (https://github.com/Microsoft/cpprestsdk)

- HTTP server is beta
- In Debian testing

[cpp-netlib] (http://cpp-netlib.org/)

- Mature
- In Debian (reasonably recent)
- Simple example is pretty simple.
- Actively developed
- Does not handle "100 Continue" requests, leading to long latencies. See http://stackoverflow.com/questions/30179476/where-is-a-delay-in-an-http-post-coming-from
- The simple example does not unbind the port (annoying).

[node.native] (https://github.com/d5/node.native)

- C++11 port of node.js
- simple example is simple
- No development for 2 years
- Not in Debian

[Boost.http] (https://github.com/BoostGSoC14/boost.http)

- GSOC from 2014
- http://boostgsoc14.github.io/boost.http/
- Actively developed
- No http2.
- Requires very new boost (>=1.57)
- Rejected from Boost: http://lists.boost.org/boost-announce/2015/08/0452.php
-
  - Maybe hard to add http2?
-
  - Mostly stuff not important to me (not header only, want a client API)

[Pistache] (http://pistache.io/)

- Pure C++11, but uses MACROS
- Not in Debian
- Not built on boost::asio
- Uses Net namespace, not pistache?
- Does not build as non-root

[libnghttp2_asio] (http://nghttp2.org/documentation/libnghttp2_asio.html)

- http2 is well supported.
- Says that the C++ bindings are experimental
- Implemented in C with a C++ api
- In Debian. Seems well maintained, but does not include the C++ library :(
- Only support http/2, not http/1.1. So curl and wget do not work

[Pion] (https://github.com/splunk/pion)

- Somewhat actively developed
- Documentation is just doxygen.
- Hello Server is a bit hard to figure out. Compiling with -std=c++11 makes it hang in a busy loop
- In Debian

- No http2 support
- Has handy iterator for url parameters