Building httpd from trunk was traditionally very challenging, since the build schema was for Visual Studio '98. While this product has been long unsupported and unavailable now, it was the only option that could be converted to (most) modern Visual Studio versions and the last version which could also exported to nmake Makefiles.

The Apache HTTP Project has been shifting toward CMAKE as a build strategy, as have many of the OSS projects it depends on.

The advantage is that PCRE can generate traditional nmake files, and can also assemble Visual Studio project and solution files for a number of different versions of that product, as well as Eclipse and other IDE's (type the 'cmake --help' command for supported build environments).

Here are the required elements to build the "bleeding edge" of the minimal components required by any httpd build. This is somewhere to start for development, but is not for the faint of heart, and is not to be trusted for most "real" server deployments.

The illustration below omits openssl, nghttp2, iconv. The libxml2 (which could replace expat entirely) is not so easily built, just yet.

```bash
Rem
Rem Illustration of building httpd and dependencies
Rem

set INSTPATH=c:\dev\apache-2.x

set CMAKE_LIBRARY_PREFIX=%INSTPATH%\lib
set CMAKE_INCLUDE_PREFIX=%INSTPATH%\include

Rem An alternative using Studio solution/projects, needs install help
Rem SET CMAKEOPTS=-G "Visual Studio 14 2015 Win64"
Rem -DCMAKE_BUILD_TYPE=Release
Rem -DCMAKE_COLOR_MAKEFILE=OFF
Rem -DCMAKE_INSTALL_PREFIX=%INSTPATH%
Rem SET BUILDCMD=devenv expat.sln /Build "Release|x64"

Rem An alternative with simple nmake Makefile...
SET CMAKEOPTS=-G "NMake Makefiles"
-DCMAKE_BUILD_TYPE=Release
-DCMAKE_COLOR_MAKEFILE=OFF
-DCMAKE_INSTALL_PREFIX=%INSTPATH%
SET BUILDCMD=nmake -f Makefile && nmake -f Makefile install

set ZLIBDIR=zlib-1.2.8
set EXPATDIR=expat-2.1.0
Rem set LIBXML2DIR=libxml2-2.9.3
set APRDIR=apr-2.0
set PCREDIR=pcre-8.38
set HTTPDDIR=httpd-2.x

if not exist %ZLIBDIR%    git clone https://github.com/madler/zlib.git %ZLIBDIR%
Rem This does not work, blame sourceforge or offer a fix?
if not exist %EXPATDIR%    https://sourceforge.net/settings/mirror_choices?projectname=expat&filename=expat-2.1.0/expat-2.1.0.tar.gz &&
tar -zxvf expat-2.1.0.tar.gz

Rem if not exist %LIBXML2DIR%    wget http://xmlsoft.org/sources/libxml2-sources-2.9.3.tar.gz &
Rem gzip -dc < libxml2-sources-2.9.3.tar.gz | tar -xvf -

Rem This requires wget/unzip - unpack it by hand if needed
if not exist %PCREDIR%    wget --passive-ftp ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/pcre-8.38.zip &
unzip pcre-8.38.zip

if not exist %APRDIR%    git clone https://github.com/apache/apr.git %APRDIR%

if not exist %HTTPDDIR%    git clone https://github.com/apache/httpd.git %HTTPDDIR%

mkdir zlib-build
cd zlib-build

mkdir httpd-build
cd httpd-build

cmake %CMAKEOPTS% -DAMD64=OFF -DASM686=OFF ..\%ZLIBDIR%
```
Rem -DAMD64=ON ought to work; but it does not
%BUILDCMD%

adir expat-build

cd ..

mkdir expat-build

cd expat-build

cmake %CMAKEOPTS% ..\%EXPATDIR%

%BUILDCMD%

cd..

Rem cd %LIBXML2DIR%\win32

Rem cscript configure.js zlib=1 prefix=%INSTPATH% sodir=%INSTPATH%\bin include=%INSTPATH%\include lib=%INSTPATH%\lib

Rem nmake -f makefile.msvc

Rem cd ..\..

mkdir pcre-build

cd pcre-build

cmake %CMAKEOPTS% -DBUILD_SHARED_LIBS=ON ..\%PCREDIR%

Rem Other valid options; -DPCRE_BUILD_PCRECPP=OFF -DPCRE_BUILD_PCREGREP=OFF

Rem since httpd itself does not use these features

%BUILDCMD%

cd..

mkdir apr-build

cd apr-build

cmake %CMAKEOPTS% ..\%APRDIR%

%BUILDCMD%

cd..

Rem Some private files of apr on Windows are used by httpd, pick these up;

mkdir %INSTPATH%\include\arch\win32

copy %APRDIR%\include\arch\win32\*.h %INSTPATH%\include\arch\win32

mkdir httpd-build

cd httpd-build

cmake %CMAKEOPTS% ..\%HTTPDDIR%

%BUILDCMD%

cd ..