

Introduction Mozilla is not just a Web browser. Mozilla is also a framework for building cross-platform applications using standards such as CSS (Cascading Style Sheets), XML languages such as XUL (XML-based User-interface Language), XBL (eXtensible Binding Language), and RDF (Resource Description Framework), as well as Gecko, Mozilla's rendering engine, and other technologies. The Mozilla development framework also makes use of programming languages such as JavaScript, C++, C, Python, and IDL (Interface Definition Language), plus framework technologies such as XPConnect and XPCOM, Mozilla's component model.

XPFE (Cross-Platform Front End) XPFE uses a number of existing web standards, such as Cascading Style Sheets, JavaScript, and XML (the XML component is a new language called XUL, the XML-based User-interface Language). In its most simple form, XPFE can be thought of as the union of each technology.

Check <http://books.mozdev.org/html/mozilla-chp-1.html#mozilla-CHP-1-SECT-1.1>

XPCOM (Cross-Platform Component Object Model)

XPCOM is Mozilla's cross-platform component object model. A component is a reusable or modular piece of code that implements a clearly defined interface. In Mozilla, this code can exist as a singleton service or an object instance. A singleton service is an object instance that is created only once and then used by other code (usually called "callers," "clients," or "consumers").

Check <http://books.mozdev.org/html/mozilla-chp-8.html>
<http://benjamin.smedbergs.us/tests/XPCOM-intro.pdf>

XPIDL (Cross-Platform Interface Definition Language) All XPCOM interfaces are defined with the Interface Definition Language (IDL). IDL provides a language-neutral way to describe the public methods and properties of a component. Mozilla actually uses a modified, cross-platform version of IDL called XPIDL to compile interface source files.

Check <http://books.mozdev.org/html/mozilla-chp-8.html#mozilla-CHP-8-SECT-1.3>

¹© CC-BY-SA by Michael Adeyeye

Assignment 1 Discuss the equivalent technologies to XPCOM, XPIDL and XPFE used in one of these browsers - Google Chrome and IE. Not less than 600 words and do include diagrams

The MXR site This is a cross reference designed to display the Mozilla source code. The sources displayed are those that are currently checked in to the main-line of the mozilla.org CVS server, Mercurial Server, and Subversion Server. The MXR site has replaced the LXR site.

Check <http://mxr.mozilla.org>

Assignment 2 Find out the improvements/changes the MXR site has over the LXR (though it no longer exist). In other words, what changes were made to the LXR to make the MXR?

Bonsai

It is a tool that lets you perform queries on the contents of a CVS archive; you can: get a list of checkins, see what checkins have been made by a given person, or on a given CVS branch, or in a particular time period. It also includes tools for looking at checkin logs (and comments); doing diffs between various versions of a file; and finding out which person is responsible for changing a particular line of code ("cvsblame").

Check <https://developer.mozilla.org/en/Bonsai>
<http://www.mozilla.org/projects/bonsai/>

Tinderbox

Tinderbox is a webtool that Mozilla developers use to check whether the current source code compiles on various platforms and passes automated test suites.

Check <https://developer.mozilla.org/en/Tinderbox>
For Firefox, check <http://tinderbox.mozilla.org/showbuilds.cgi?tree=Firefox>

Getting build requirements - Platform (OS and Browser ver.) dependencies

https://developer.mozilla.org/En/Developer_Guide/Build_Instructions/Linux_Prerequisites
https://developer.mozilla.org/en/Windows_Build_Prerequisites
https://developer.mozilla.org/En/Simple_Firefox_build
https://developer.mozilla.org/en/Configuring_Build_Options

Web Browser Build Process - Using Revision Control Systems (CVS, SVN, HG) and Compressed Files (tar and zipped files)

Introduction to makefiles, using .mozconfig
https://developer.mozilla.org/en/Build_Documentation
https://developer.mozilla.org/en/Download_Mozilla_Source_Code (via HTTP)
https://developer.mozilla.org/en/Mozilla_Source_Code_%28Mercurial%29 (via Mercurial)
https://developer.mozilla.org/en/Mozilla_Source_Code_Via_CVS (via CVS)

Debugging

- <http://www.ngportal.com/micadeyeye/index.php/2008/12/12/building-ff-200x-on-ubuntu-8x/>
- <http://www.ngportal.com/micadeyeye/index.php?s=.mozconfig>

Assignment 3 Build firefox from source and post on your blog a screenshot of the (successful) build. Report your challenges (problems you encountered) during the build process.