

Building a Mozilla IDE based on OpenKomodo

Existing projects & notable features:

XulExplorer

- Mozilla based - XulRunner application
- Extensible - Addons as Firefox
- Snippets
- XUL Checker

A set of XUL *Best Practices*. From the XulExplorer documentation:

The XUL Checker feature is a XUL structure check and best practice guide. It will scan the XUL in the selected editor tab and log any potential problems it finds with the XUL..

- XUL formater - reformatting the XML code
- XML syntax checking
- DOMi & Venkman integration
- Extension & XulApp wizards
- Live preview of your XUL code
- Launcher : You can easily try your extension or your XulRunner application from a menu.

XulBooster

- Eclipse based
- Extension Wizard
- XBL editor and structural view
- XUL snippets
- Extension launcher
- Preview

XulMaker *dead project*

- Mozilla based

- GUI Builder based on a DOMi like UI
- XUL Snippets
- preview

OpenKomodo - aka KomodoEdit:

- Mozilla & python based
- A multi language editor
- powerful text editor, based on Scintilla
 - o completion
 - o syntax highlightning
 - o folding
 - o ...

for a exhaustive list of features, see: <http://scintilla.sourceforge.net/ScintillaADoc.html>

- highly extensible:
 - o Addons - as Firefox
 - o Text Macro record - a suite of text handling
 - o Script Macro

Launch a Python or a Javascript script. You can easily create a XUL file that can be launched from a such script. It's a way to provide some specific tools for projects, as the existing "Firefox Extension template" which provide a graphic tool (in XUL) to configure the extension.

- o Template - pre-structured file and/or project
- o Command - launch a command and play with I/O
- o Snippets - insert a piece of code in the current file
- o toolbar/menu customizable

In a project, you can create specific menu or toolbarbuttons which will be added to the general UI of

Building a Mozilla IDE based on OpenKomodo

OpenKomodo (as a dynamic overlay)

These buttons/menu are linked to a command or a macro.

- syntax checking
- generic project concept

Requirements

The Mozpad community concluded with a set of global requirements:

The three main goals of a Mozilla IDE are:

- *make the creation of a Mozilla related project easier (chrome registration, install.rdf and application.ini creation, ...)*
- *make some labored concept more accessible (chrome, build process, IIO, XPCOM, overlay, ...)*
- *make debugging experience better*

A Mozilla IDE must be based on Mozilla technologies and be an OpenSource software.

A Mozilla IDE must provide a powerful code editor engine (syntax highlighting, autocompletion mechanism, multi language support, folding, ...).

A Mozilla IDE must be highly extensible.

A Mozilla IDE must provide a good documentation and help mechanism.

The closest project to these rules is OpenKomodo, especially since it provides a Scintilla based text editor and a generic and extensible project manager.

Here's a more detailed list of requirements.

yellow text: Implemented in OpenKomodo

Project management:

The purpose is to provide easy ways to create and manage Mozilla product extensions, XulRunner application and any kind of Mozilla based projects.

kind of projects

- Add-on
 - o Created from
 - a XPI file
 - an assistant
 - o A tool to build XPI files

OpenKomodo already provides a template to create Firefox extension, but it is actually pretty basic. XulExplorer provide a similar tool but with an overlay assistant and a set of predefined sUI components

- XulRunner application
 - o Created from an assistant
 - o A tool to build the application (with a specific branding)
- classic Mozilla based product
 - o as Firefox or Thunderbird
 - o Created from an existing source tree or from an assistant (with a way to select a project from the Mozilla CVS repository)
 - o Compilation assistant (with a mozconfig creator tool)
 - o Possibility to create an extension (--enable-extension) or an application (--enabled-application)

For each kind of projects, the IDE must provide a way to test the project.

chrome

- Chrome browser: a tool for managing all chrome packages and for browsing

Building a Mozilla IDE based on OpenKomodo

through all chrome package

The Verboisio project provides a manifest parser. It could be interesting to consider a such code.

- FIXME: what about overlays ?

XPCoM & Interfaces

- XPCoM and IDL creation assistant
- A list of XPCoM components and interfaces hosted

I10n

- A tool for managing the localization of the project
- A way to populate a .dtd file from a XML file with not referenced entities

source code

- Code Browser

KomodoIDE (not OpenKomodo) includes a tool providing a view of the global structure of the project source. From the Komodo documentation:

*Use the **Code Browser** to view the general program structure of all source files open in the Editor Pane. For each source file, the Code Browser displays a tree of symbol nodes, including modules, classes, functions, interfaces, namespaces, imports and variables.*

A such feature could be really interesting.

- source code could be retrieved and synchronized from a remote repository (svn, cvs, hg, git...)
- It must be possible to edit files hosted in a zipped file and to browse zipped files

see bugs. 309296 & 379633

Code editing

multi language support

- XUL, XBL, XHTML, CSS, RDF, IDL, CPP, SVG, XML, properties, dtd, manifest, ini
- syntax checking
- automatic completion mechanism based on set of API
 - o XPCoM
 - o DOM
 - o FUEL
- XUL Checker

It should be based on the XulExplorer related code

An advanced text editor

- syntax highlightning
- folding
- auto indentation
- macro recording
- bookmarking lines

Snippets

- a tool to manage a list of snippets
- Some snippets:
 - o XUL (all XUL tags, with, if it makes sense, a wizard to ask some information, for instance, the creation of a new <dialog/> tag)
 - o Javascript (as instantiation of well-know XPCoM services)
 - o ... and more.

language specific tools

It must be possible to handle some files

Building a Mozilla IDE based on OpenKomodo

from an abstract graphic view (as a tree)

- XML, with a DOM view
- XBL, with an API view and a DOM view
- IDL, with an API view
- JS: a structural view (objects and prototype)
- CSS: FIXME ???

UI related tools

- GUI Builder, a GUI assistant
- A *wysiwyg* tool
- An interactive preview of the XML code if it's possible
- CSS advanced editor
- color picker for CSS editing

Debugging

FIXME

- Venkman
- DOMi
- Firebug
- jssh

Extensibility

- add-ons as Firefox
- possibility to add other language support
- possibility to add other API support

Documentation

- language reference
- interactive help
- tutorial projects

It's actually possible to create tutorials with OpenKomodo

- a cview like

GUI

- Dockable GUI
- Toolbars and menubars must be able to host snippets, commands or macros

OpenKomodo allow such items to be added to the main toolbar. It could be interesting to add a way to create new toolbar for specific projects

misc

- It could be really interesting to provide the Scintilla based text editor as a standalone element for other projects
- What about rebuild OpenKomodo with a specific branding ?
- Must work with Gecko 1.9

FIXME: add screenshots:

<http://www.mozbox.org/pub/IDE/>