

Mozilla Technology Introduction

NTU CSIE and *csie.org*
Mozilla Localization: Traditional Chinese
Hung-Te Lin 林弘德

icos'03

1

Mozilla Technology Introduction

- Part A: Web sites with Gecko
 - For Web Authoring
 - Req.: HTML/CSS/JavaScript
- Part B: Extending Mozilla: XUL-World!
 - For Mozilla extensions developer
 - Req.: XML*/JavaScript
- Part C: Hack the source
 - For source level programmers
 - Req.: Programming common sense

2

Part A: Web sites with Gecko

For Web Authoring
Req.: HTML/CSS/JavaScript

3

Outline

- Websites INCOMPATIBLE!
 - Presentational Error
 - Functional Error
 - Misc Error
- Writing Good Pages

4

[Pages INCOMPATIBLE!!!!]

- Presentational Error
 - CSS rendering
- Functional Error
 - JavaScript error
- Misc Error
 - Content-Type Error

5

[Presentational Error: Plugins]

- Plugin presentational error
 - Flash not transparent
 - no solution until Macromedia fixed it
 - Java cannot display ODBC like Chinese
 - Incorrect version, or many reasons
 - ActiveX Controls
 - Currently not available
 - Windows Media not playing
 - Must follow Mozilla's embed rule

6

[Presentation Error: HTML Tags]

- IE-specific tags like <marquee>
 - Solved by XUL+XBL, emulation
- XML/MathML not correct?
 - Maybe DTD not found
 - MathML requires correct Content Type

7

[Presentational Error: CSS!!]

- CSS is the most common error recently.
- IE does not support full CSS2
- IE's ActiveX and filter: is not supported
 - IE Alpha filter: filter:alpha(opacity=75);
 - Mozilla way: -moz-opacity:0.75;
- DIV layout box model error
 - No solution
- IE has some CSS error
 - Some hacks can prevent this

8

[Functional Error]

- VB Script?
 - No Solution
- JavaScript Error?
 - Mostly only small piece of code needs to be modified
 - Fix it if you can access the page, or ask the site maintainer to do so.
 - Or, Save the page and change it locally

9

[Functional Error: Common JavaScript Error]

- Use "JavaScript Console" to see if any error occurs, and click will bring you to the location of source
- The most common error is "Cannot find object: XXXX" which is due to IE's syntax.
- We DO have innerHTML but you must get the object first
- Mozilla has no 'all' attribute
 - You may use all other ways to find elements, actually even document.NAME is better. See also: *document.getElementByld*, *document.getElementsByTagName*.
- There is no "children". We have *childNodes*
- Cannot submit Forms
 - IE can sometimes just use object's name to access it. Mozilla is more restricted. Only forms can use this format for compatible with old pages

10

[Functional Error: JavaScript Error -- Sample]

- <http://www.kingstone.com.tw/>, the search is not functional.
- JS Console says "input_form is not defined", line 122
- Click to view:
 - ```
document.search_form.p19.value = "";
```
  - ```
if (input_form.p.value=="") {
```
 - ```
 alert("請至少輸入一個查詢條件！");
```
  - Submit is O.K now, but this page has more bugs
  - // ----- Check and fixup incoming data block
  - ```
if(!document.body.children.incoming.children.properties)
```
 - ```
{ document.all.incoming.innerHTML = "<DIV"}
```
  - ```
<DIV id="1">
```
 - ```
 <div id="summary"> </div> </div>
```
  - ```
<DIV id="2">
```
 - ```
 <div id="summary"> ...
```
  - **You cannot repeat ID.**
  - .....

11

## [ Misc Error: Content-Type ]

- Mozilla trusts HTTP Header "Content-Type", while IE trusts on content preload.
- Situation: downloaded some file and Mozilla treated as plain/text
- Solution:
  - Update remote magic
  - Use .htaccess
    - AddType application/x-msdos-program .xpi
- I want to view in plain/text?
  - Will be solved in future
  - Currently: about:config and change
    - Network.http.accept.default
    - `text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,video/x-mng,image/png,image/jpeg,image/gif;q=0.2,*/*;q=0.1`

12

## [ The Standards ]

- What JavaScripts can we use in Gecko?
  - ECMA262-Edition 3
  - DOM
  - Netscape JavaScript
- <http://www.xulplanet.com/references/elemref/>

13

## [ Writing Good Pages ]

- Use CSS, because you can avoid some useless HTML Tables, JavaScript and images.
- Try to make your page available to Gecko, IE, and text-based like lynx
- You may choose the best presentation target, like Gecko and IE
- JavaScript should run on most browsers

14

## [ Q & A ]

- Websites INCOMPATIBLE!
  - Presentational Error
  - Functional Error
  - Misc Error
- Writing Good Pages

15

## [ Part B: Extending Mozilla: XUL-World! ]

For Mozilla extensions  
developer  
Req.: XML\*/JavaScript

16

# [ Outline ]

- XULPlanet
- XUL: XML, JavaScript, XBL, XPCOM
  - Writing XUL Programs
  - Mozilla packages and Layout
- userChrome and userContent
- XPInstall, Localization, Themes
- Sidebar and Sherlock

17

# [ XULPlanet ]

- XULPlanet: <http://www.xulplanet.com/>
- A good tutorial and source for XUL



18

# [ XUL: XML and JavaScript ]

- XUL: pronounced as “zool”, like “cool”
- XML User-interface Language
- Case sensitive
- Usually split into:
  - XML/RDF for layout and elements
  - CSS for style
  - Entity like Locale Resource
  - Scripts (currently most JavaScript)
  - Additional images

19

# [ XUL: Packaging ]

- Chrome: UI elements in application
  - Content: Windows and Scripts
  - Skin: Style sheets, images
  - Locale: Locale and resource
  - `chrome://<component>/content/<file.xul>`
    - `chrome://dom/content/dom.xul`
- Content Area: request document

20

## XUL: Manifest

- To specify package location

- Sample:

```
<?xml version="1.0"?>
<RDF:RDF
 xmlns:RDF="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:chrome="http://www.mozilla.org/rdf/chrome#">
 <RDF:Seq about="urn:mozilla:package:root">
 <RDF:li resource="urn:mozilla:package:findfile"/> </RDF:Seq>
 <RDF:Description about="urn:mozilla:package:findfile"
 chrome:displayName="Find Files" chrome:author="Whoever"
 chrome:name="findfile">
 </RDF:Description>
</RDF:RDF>
```

- Chrome directory: "installed-chrome.txt"
  - content,install,path,/main/calculator/

21

## XUL: XUL for a simple window

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin/" type="text/css"?>
<window id="findfile-window" title="Find Files" orient="horizontal"
 xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
 <button label="Normal"/>
 <button label="Disabled" disabled="true"/>
 <button label="Default" default="true"/>
</window>
```

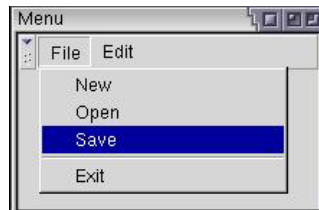


Invocation:

1. just browse this xul file and it'll be shown.
2. If you want to specify the size, use JavaScript and assign.
  - `window.open("chrome://navigator/content/navigator.xul", "bmarks", "chrome,width=600,height=300");`
3. Run mozilla -chrome chrome://.....
  - `resource://` refer to the directory where Mozilla installed

## XUL: Menu

```
<toolbox flex="1">
 <menubar id="sample-menubar">
 <menu id="file-menu" label="File">
 <menupopup id="file-popup">
 <menuitem label="New"/>
 <menuitem label="Open"/>
 <menuitem label="Save"/>
 <menuseparator/>
 <menuitem label="Exit"/>
 </menupopup></menu>
 <menu id="edit-menu" label="Edit">
 <menupopup id="edit-popup">
 <menuitem label="Undo"/>
 <menuitem label="Redo"/>
 </menupopup>
 </menu>
 </menubar>
</toolbox>
```



23

## XUL: Add Events! (DOM2)

```
<script src="blah.js">
<menuitem label="New" accesskey="n"
 oncommand="alert('What's new?');"/> ...
<button id="cancel-button" label="Cancel"
 oncommand="window.close();"/>
```

24

## XUL: Style it

- HTML styles
  - <html:table>, <html:br>, .....
- CSS Styling
  - <?xml-stylesheet href="chrome://global/skin/" type="text/css"?>
  - Use HTML 'class=' 'id=' and other style rules to style your user interface

25

## XUL: XBL, eXtensible Bindings Language

- Declaring behavior of XUL widgets.
- Limitation of XUL:
  - You can't change how elements works
- Sample: define a new 'box' with 3 buttons

```
<?xml version="1.0"?>
<bindings xmlns="http://www.mozilla.org/xbl">
 <binding id="binding1">
 <!-- content, property, method and event descriptions go here -->
 </binding>
</bindings>

scrollbar { -moz-binding: url('chrome://findfile/content/findfile.xml#binding1'); }
```

## XUL/XBL: Making OO widgets!

- Attribute Inheritance
- XBL Bindings:
  - Content
    - Appearance of widget
  - Property/Field
    - Field: Generated variables like random
    - Property: Get/Set
  - Method and event descriptions
    - Action of some event

27

## XPCOM: Cross-Platform Component Object Model

- Limits of XUL JavaScripts:
  - You cannot access low level APIs
- Solution:
  - Write native code to do low level API
  - Provide an interface for JavaScript to call
  - Like most OO Models with interface/implementation
- Using: (ActiveX way)
  - Get a component
  - Get the part that implements interface you want
  - Call the function

```
var aFile = Components.classes["@mozilla.org/file/local;1"].createInstance();
if (!aFile) return false;
var LocalFile=aFile.QueryInterface(Components.interfaces.nsILocalFile);
if (!aLocalFile) return false;
```

## Mozilla Themes (Skins) and Locales

- Change chrome files!
- Change Skin/Locale: switch specified package in manifest file.
- Locales: UTF8 defined in DTD files
  - Use: `<button label="&findLabel"/>`
  - Declare: `<!ENTITY findLabel "Find">`
- DTD for XUL, and properties for scripts
  - In XUL: `<stringbundle id="strings" src="strings.properties"/>`
  - In script: `var strbundle=document.getElementById("strings");`
  - In Prop: `notFoundAlert=Not found!`

29

## Why're Mozilla themes so hard?

- The most flexible than others
- Hardest than others
  - Most themes only change toolbar
- You have to style all CSS
- All of these applies to langpacks
- Themes error may lead to unavailable while langpacks error leads to crash

30

## XPIInstall

- Cross Platform Installation
- Package your work and make a install.js then name it ".xpi"
- Many ways to install it, like
  - Javascript: `var xpi = {}; xpi[caption]= url; InstallTrigger.install(xpi);`
  - `InstallTrigger.startSoftwareUpdate(url);`
  - Just Open XPI!

31

## Mozilla Package and Layout

- mozilla/bin/
  - chrome/ the chrome (XUL)
    - en-US.jar, en-win/mac/unix.jar, US.jar: locale
    - comm.jar, messenger.jar: content
    - modern.jar, classic.jar: skin
    - Installed-chrome.txt: text index
    - chrome.rdf: registered chrome
  - components/ XPCOM native
  - defaults/ default settings
  - plugins/ Plug-ins
  - res/ resources like HTML
  - searchplugins/ Sherlock

32



## [ Profile Layout ]

- [personal folder]/[.]mozilla/
  - registry.dat Profile Index
  - profiles/PROFILE\_NAME
    - XXXXXXXX.slt
      - chrome/ User chrome
      - XUL.mfl XUL cache
      - pref.js, user.js pref (now about:config)
      - bookmarks.html
      - .....
      - Mail/

33

## [ ~/chrome/userChrome|Content ]

- userChrome.css and userContent.css
  - Customize your favorite style
- userChrome sample:
  - Change the Mozilla Status Bar small font
  - \* {  
font-size: 14pt !important;  
}
- userContent sample:
  - MSDN wrong text format

34

## [ Sidebar: Just As Easy As XUL ]

- Sidebar is a small area
- You may use both HTML and XUL
- javascript:window.sidebar.addPanel(  
caption, URL, customizeURL);
- [http://developer.netscape.com/docs/  
manuals/browser/sidebar/](http://developer.netscape.com/docs/manuals/browser/sidebar/)

35

## [ Sherlock ]

- Easy to be modified
- Look at searchplugins/\*
- Each file for .gif logo and .src file
- Javascript:  
window.sidebar.addSearchEngine(  
srcURL, gifURL, Caption, Category);

36

## [ Sherlock: Sample ]

- <http://www.google.com/search?q=blah&hl=zh-TW&num=20&sourceid=mozilla-search&start=0>
- # Mozilla/Google plug-in by amitp@google.com
- `<search name="Google" description="Google Search for Traditional Chinese" method="GET" action="http://www.google.com/search" queryEncoding="2563" queryCharset="Big5" # updateCheckDays=30 update="http://www.google.com/mozilla/google.src" #updateCheckDays=1 >`
- `<input name="q" user> <input name="hl" value="zh-TW">`
- `<input name="lr" value=""> <input name="num" value="20">`
- `<input name="sourceid" value="mozilla-search">`
- `<inputnext name="start" factor="20">`
- `<inputprev name="start" factor="20">`
- `<interpret charset = "Big5" resultEncoding = "2563" browserResultType="result" resultListStart="<!--a-->" resultListEnd="<!--z-->" resultItemStart="<!--m-->" resultItemEnd="<!--n-->" >`
- `</search>`

37

## [ Q&A ]

- XULPlanet
- XUL: XML, JavaScript, XBL, XPCOM
  - Writing XUL Programs
  - Mozilla packages and Layout
- userChrome and userContent
- XPInstall, Localization, Themes
- Sidebar and Sherlock

38

## [ Part C: Hack the source ]

For source level programmers  
Req.: Programming common sense

39

## [ Outline ]

- Getting the source
- Build the lizard
- Where should I get started?
- Porting: NSPR and XPToolkit

40

## Getting the source

- You can download the tarball
  - Currently Firebird has only source tarballs for releases, not nightly
- CVS:
  - `:pserver:anonymous@cvms-mirror.mozilla.org:/cvsroot`
- Browse online at [lxr.mozilla.org](http://lxr.mozilla.org)



41

## Build the lizard

- <http://www.mozilla.org/build/>



42

## Where should I get started?

- Simple APPs: Just use XUL!
- When you need Native support:
  - Look at XPCOM
  - XPConnect: binding of XPCOM and JS
- When you want to write plugins:
  - <http://www.mozilla.org/projects/plugins/>
- I want to use Gecko Engine!
  - <http://www.mozilla.org/projects/embedding/>

43

## Porting

- NSPR: non-GUI OS facilities
  - Threading
  - File and network I/O
  - Timing
  - Memory allocation
  - Shared library linking
- XPTToolkit(XPFE or Toolkit)
  - The key to XUL
  - <http://lxr.mozilla.org/seamonkey/source/widget/>

44

## [ Q&A ]

- Getting the source
- Build the lizard
- Where should I get started?
- Porting: NSPR and XPToolkit

Thanks all!