

2016 - 2017

# Programarea Clientului Web

---

conf dr. ing. Simona Caraiman

asist drd.inf. Tiberius Dumitriu

mailto: [tiberiusdumitriu@yahoo.com](mailto:tiberiusdumitriu@yahoo.com)

**Universitatea Tehnica “Gheorghe Asachi” din Iasi**  
**Facultatea de Automatica si Calculatoare**

# Curs 1-3 - overview

---

Internet vs. WEB

Website vs. Web app. vs. Desktop App

Programare (client) Web – Context

Programare (client) Web – Basics

- structura website (static vs. dinamic)
- comunicatia client-server pe Web
  - URI, DNS, HTTP
- dezvoltarea unui site Web
  - responsabilitati client/server
  - web stacks – tehnologii client/server

# Curs 1-3 - overview

---

Web programming - client-side

- Browser-ul Web
  - arhitectura, exemple,
  - extensibilitate: plugins

# Curs 4+ - preview

---

Web programming - client-side

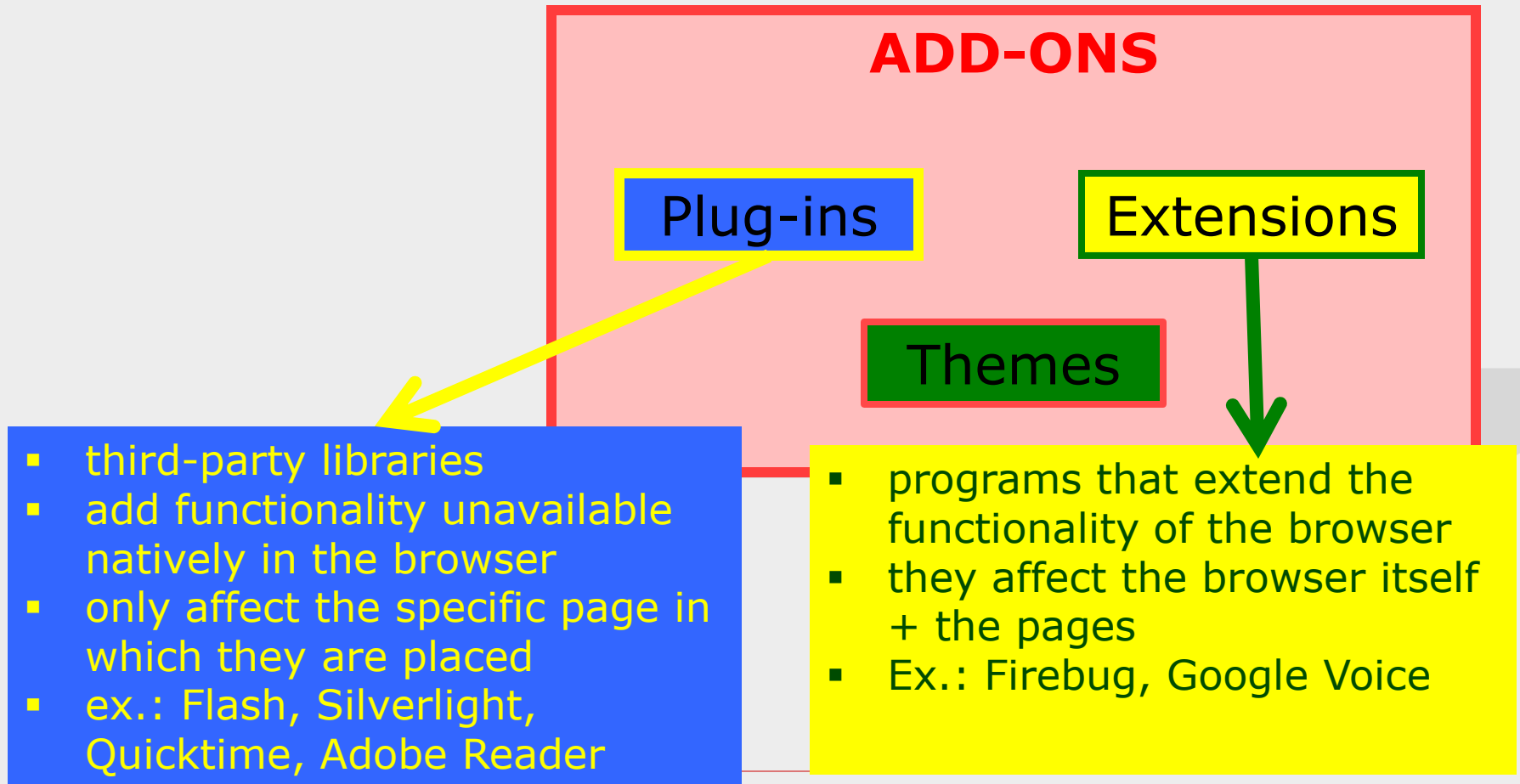
- Browser-ul Web
  - arhitectura, exemple,
  - extensibilitate: plugins, extensii
- tehnologii
  - HTML, CSS
  - Javascript
  - XML
  - AJAX
  - Web Workers, Web Storage

---

# **Add-ons - plug-ins, extensii**

# Add-ons

---



# Plug-ins vs. extensii

---

## □ **Plug-ins**

- biblioteci shared
- ajuta browser-ul sa afiseze continut specific (ex. fisiere multimedia)
- sunt incluse ('embedded') intr-o pagina web

## □ **Extensii**

- adauga noi functionalitati browser-ului
- pot fi vazute ca parte integranta din browser
- pot include un plug-in

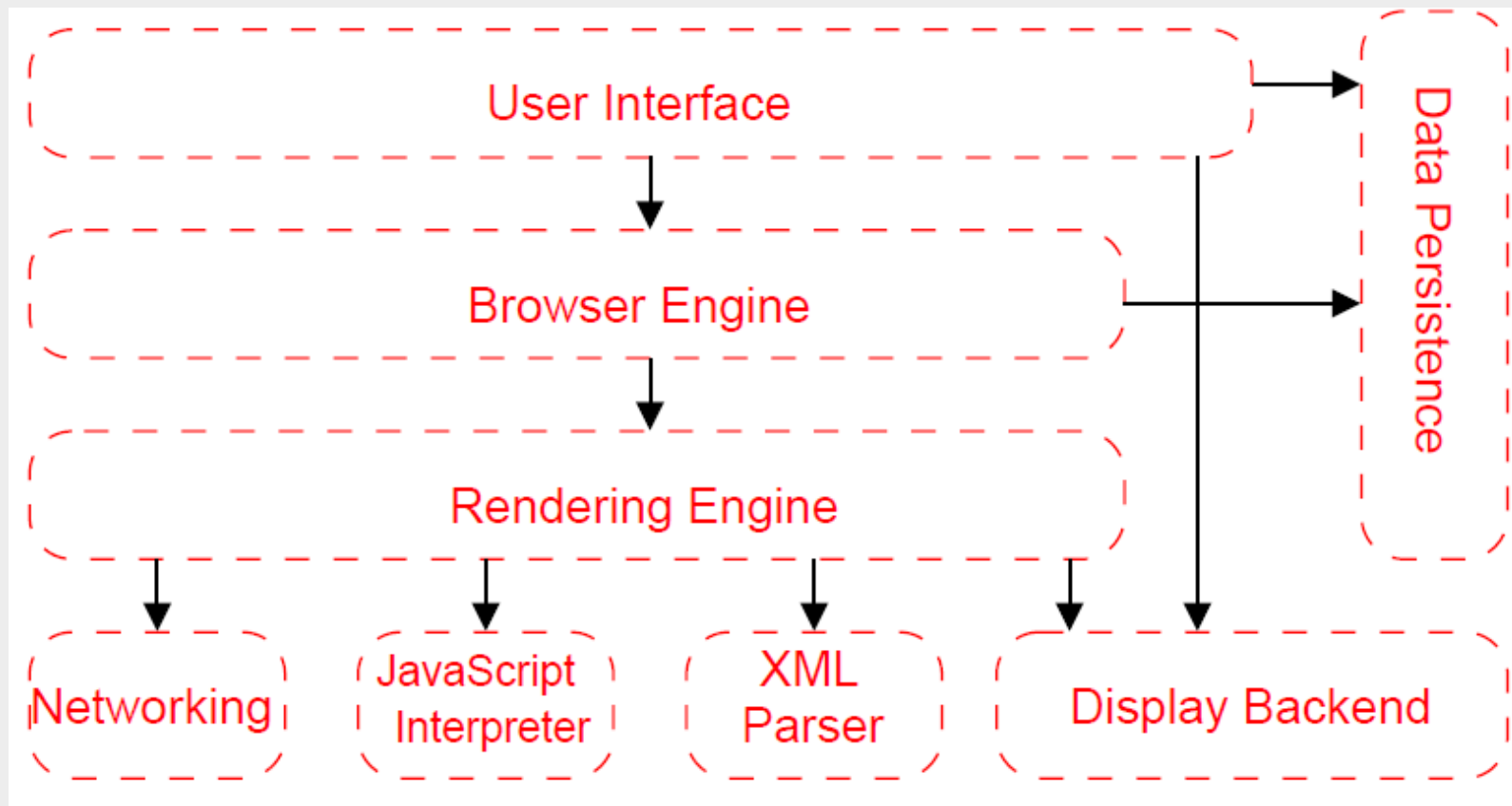
---

# Extensii Mozilla Firefox



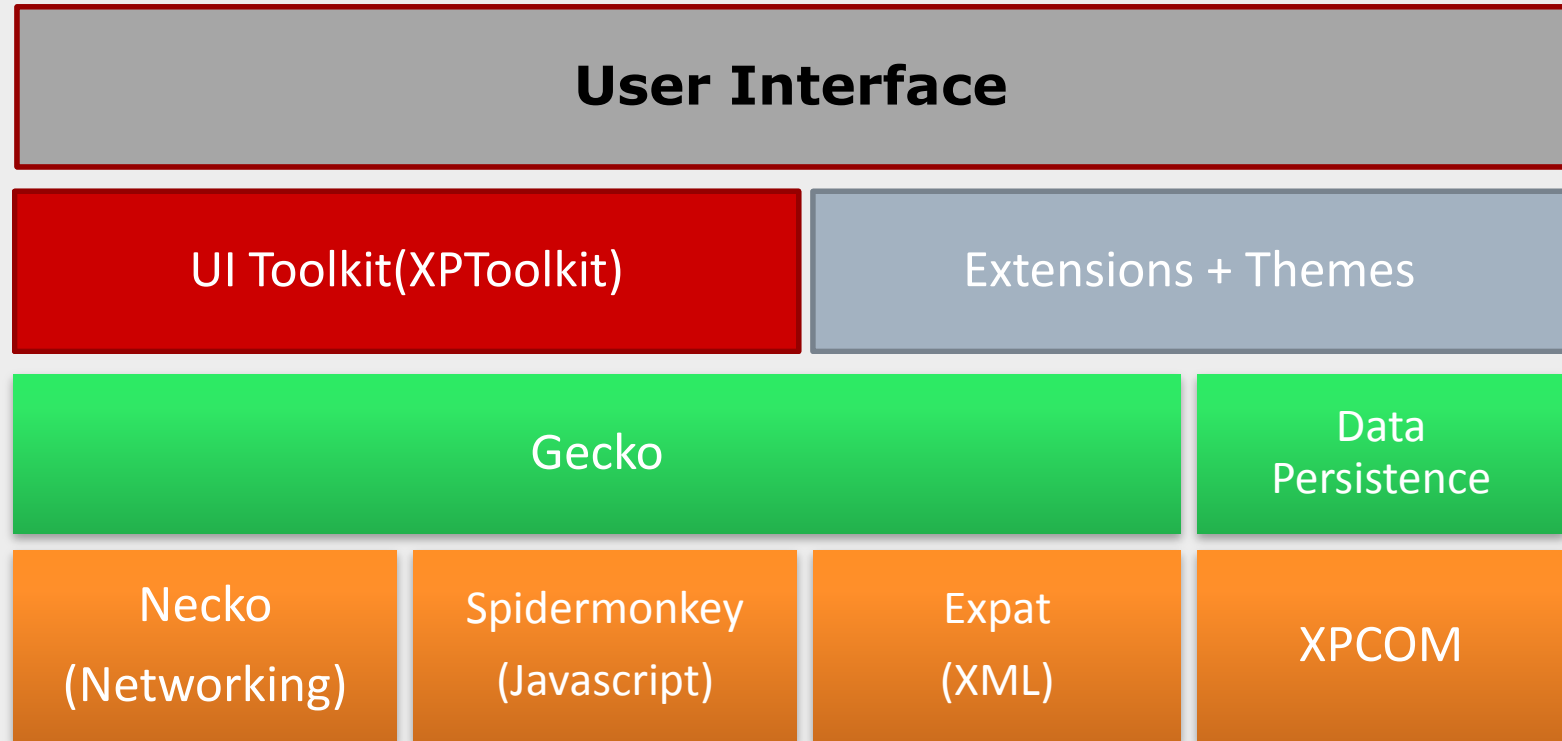
# Browser Web – arhitectura de referinta

---



# Arhitectura Firefox

---



# Tehnologii

---

javascript

dom

xul

ajax

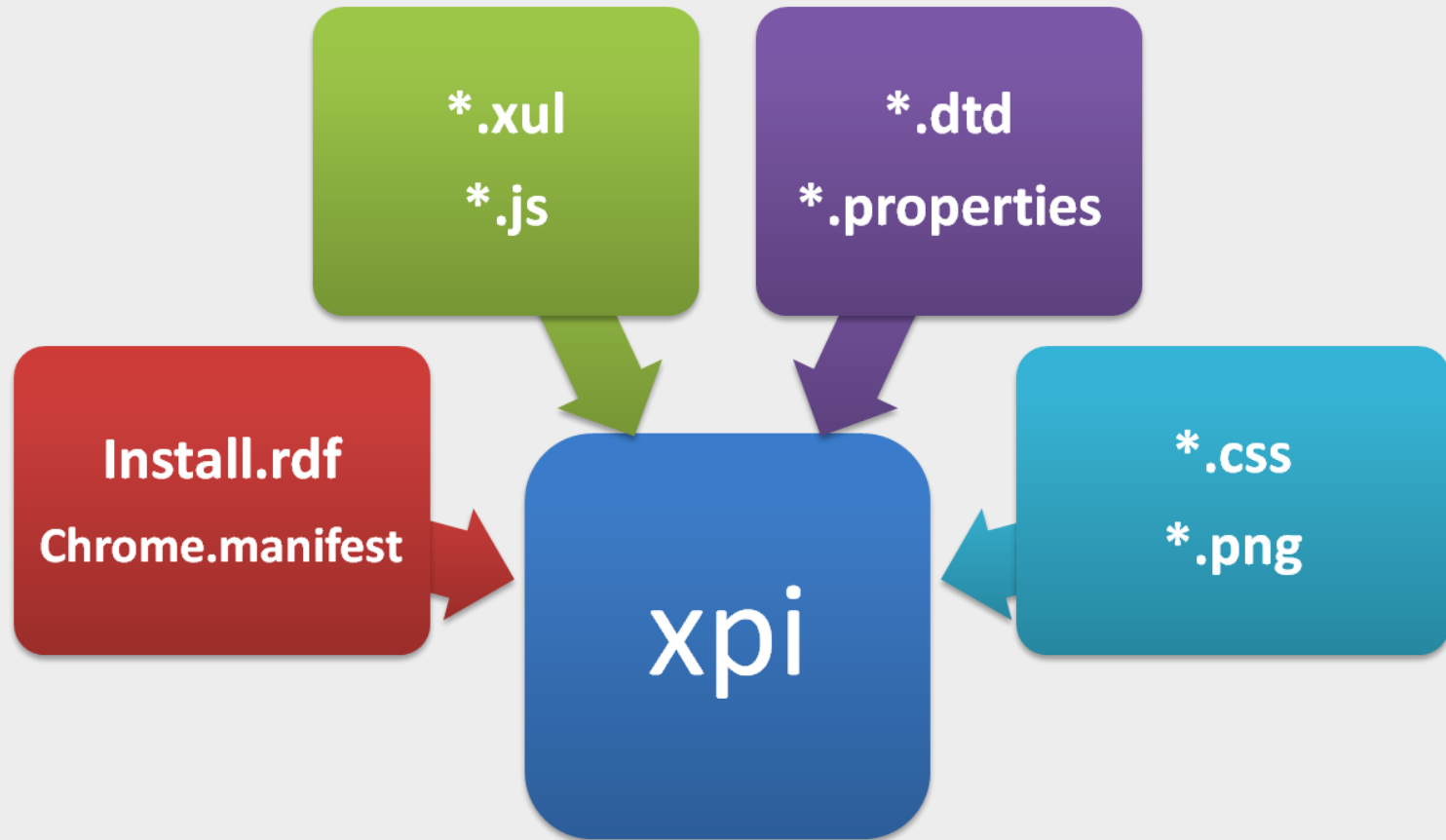
xml

css

xpcom

# O extensie Firefox

---



# Considerente teoretice

---

- ❑ Opensource: comunitatea participa la dezvoltarea unei aplicatii.
- ❑ Securitate: In sistemul opensource autoritatea nu garanteaza solutia. Oricine poate lua sursa, introduce orice vrea el in ea si distribui aplicatia in continuare.
- ❑ Orice aplicatie opensource ar trebui sa aiba o autoritate centrala oficiala (grup de programatori, fundatie, etc...), care sa distribuie aplicatia, si sa aprobe modificarile: The Mozilla Foundation.
- ❑ Mozilla separa interfata (UI) de cod. Interfata este definita independent de platforma/OS si interactioneaza cu codul compilat dependent de platforma.

# Interfata

---

- ❑ Interfata independenta de platforma (care include si o mare parte din functionalitate !) se numeste Chrome si este compusa din:
- ❑ XUL (.xul) - extensible user interface language - defineste practic interfata (pozitionarea widget-urilor, ce functii sunt apelate cand trebuie sa se intample ceva)
- ❑ Skin-uri (.css, .gif, etc...) – aspectul (“lookul”) programului
- ❑ Javascript (.js) - functionalitatea UI-ului (ce se intampla cand apesi un buton, etc)
- ❑ Elemente locale - textul interfetei pentru diferite limbi

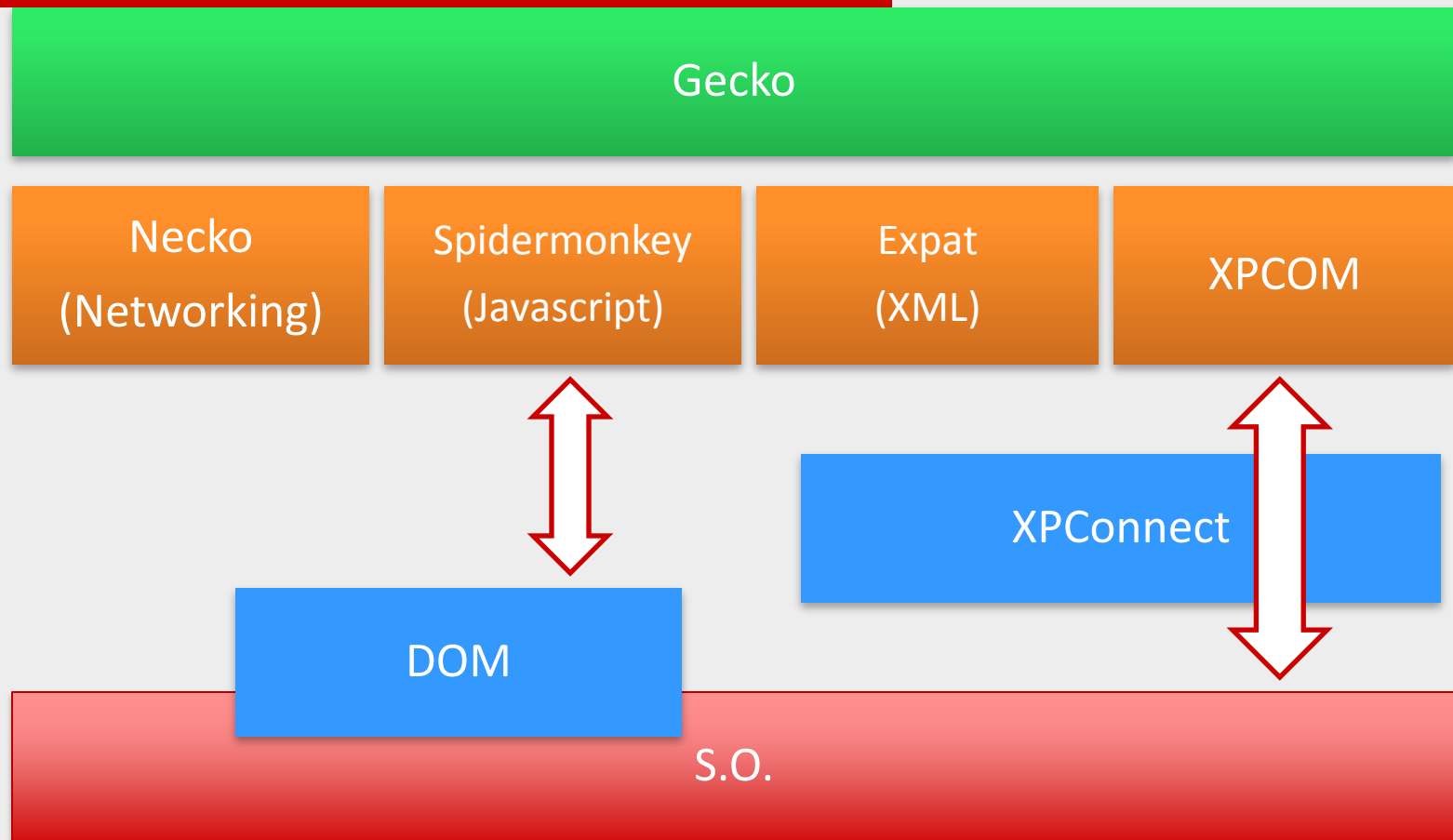
# Comunicarea cu platforma

---

- ❑ Partea dependenta de platforma (Gecko).
- ❑ Exporta functionalitatea prin interfeta XPCOM (Cross-Platform Component Object Model), similar cu COM-ul de la MS.
- ❑ Interfetele XPCOM pot fi accesate din cod nativ (C++ de exemplu) se poate crea propriul tau browser pe baza Gecko + Javascript , folosind XPConnect.
- ❑ O parte din obiectele browserului vor fi accesibile pur si simplu din DOM, dar XPCOM permite interactiunea cu multe alte obiecte (ex.: clientul de mail, meniurile, etc...)

# Comunicarea cu platforma

---





# Metadata: install.rdf

---

```
<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:em="http://www.mozilla.org/2004/em-rdf#">
  <Description about="urn:mozilla:install-manifest">
    <em:id>sample@sample.com</em:id>
    <em:version>1.0</em:version>
    <em:type>2</em:type>
    <!-- Target Application this extension can install into,
        with minimum and maximum supported versions. -->
    <em:targetApplication>
      <Description>
        <em:id>{ec8030f7-c20a-464f-9b0e-13a3a9e97384}</em:id>
        <em:minVersion>1.5</em:minVersion>
        <em:maxVersion>1.5</em:maxVersion>
      </Description>
    </em:targetApplication>
    <!-- Front End MetaData -->
    <em:name>My Extension</em:name>
    <em:description>A sample extension.</em:description>
    <em:creator>Your Name Here</em:creator>
    <em:homepageURL>http://www.mypage.com/</em:homepageURL>
  </Description>
</RDF>
```

# Metadata: chrome.manifest

---

<b>content</b>	MyExt	chrome/content/	
<b>locale</b>	MyExt	en-US	chrome/locale/en-US/MyExt/
<b>overlay</b>	chrome://browser/content/browser.xul chrome://MyExt/content/browser.xul		
<b>overlay</b>	chrome://browser/content/preferences/preferences.xul chrome://MyExt/content/preferences/preferences.xul		

## Chrome providers:

- content
  - main source file for a window description
  - \*.xul, \*.js
- locale
  - allows translators to plug in a different chrome package to translate an application
- skin
  - responsible for providing a complete set of files that describe the visual appearance of the chrome
  - images, css files

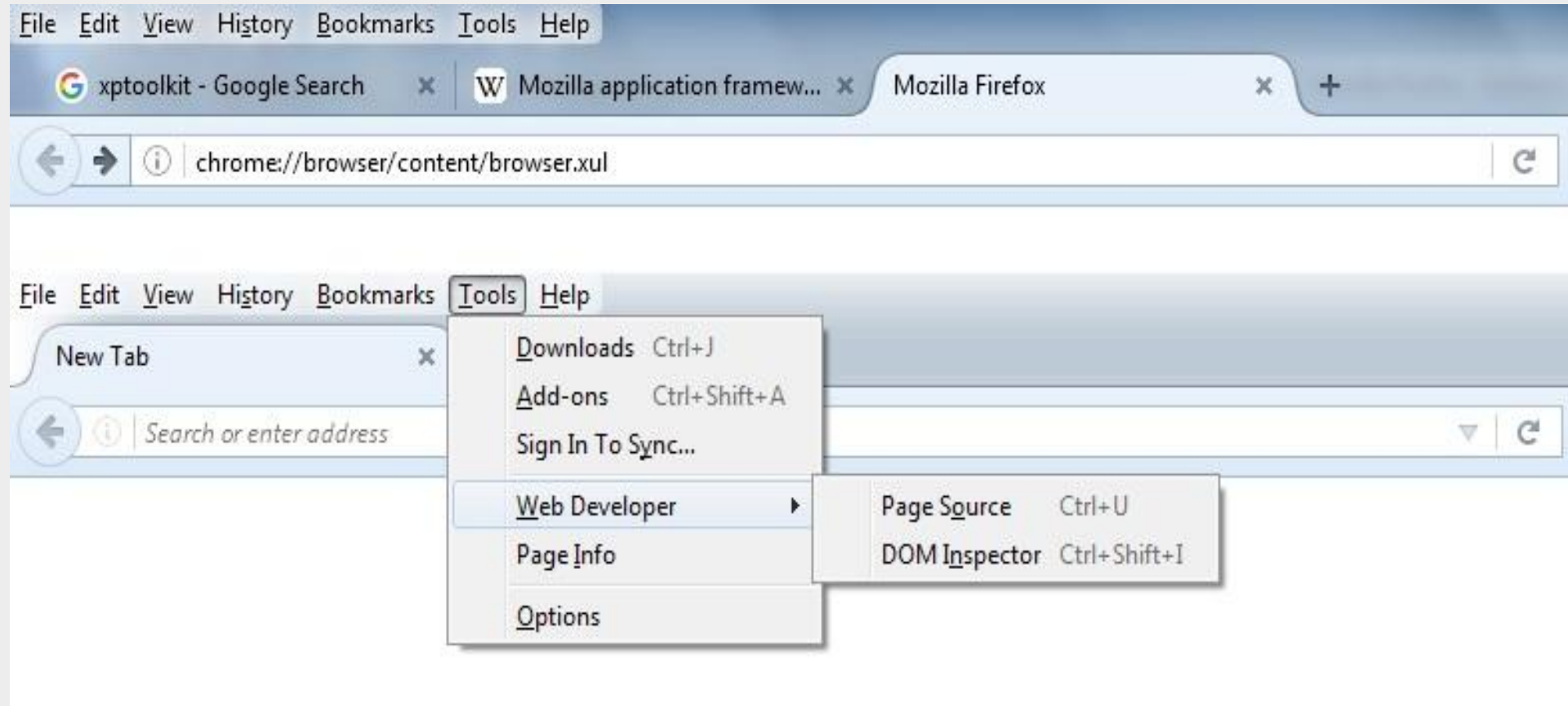
# The *chrome*

---

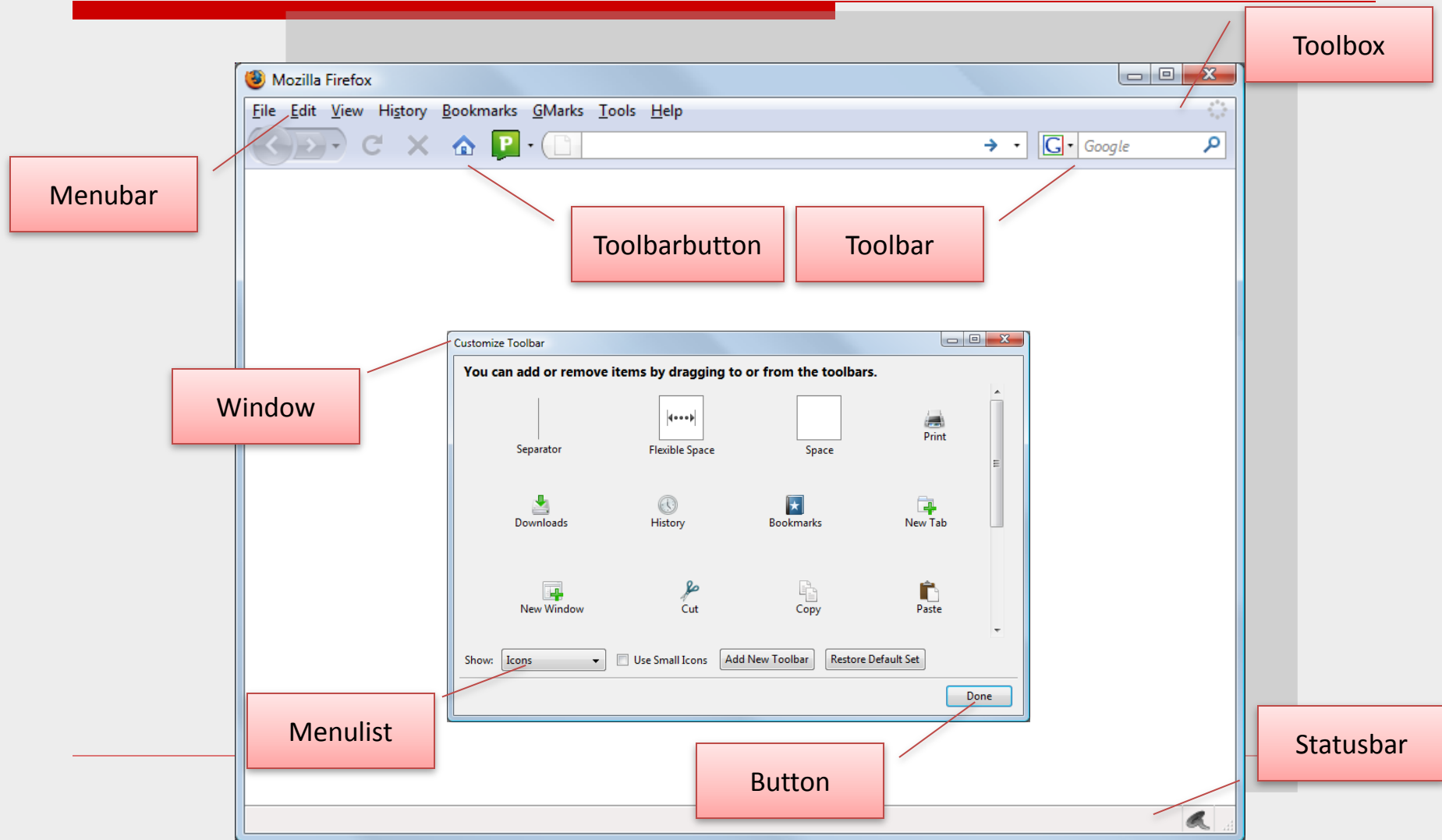
- Se refera la intreg pachetul UI
  - XUL, javascript, css,...
  
- Poate fi adresat printr-un URI
  - `chrome://browser/content/browser.xul`
  - `chrome://myextension/content/overlay.xul`

# The *chrome*

---



# Chrome UI



# XUL

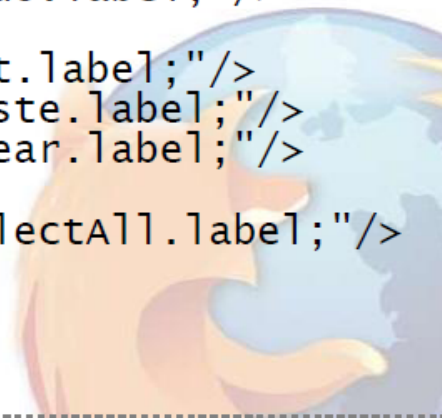
---

- XML User-interface Language
  - limbaj bazat pe XML
- set de controale UI
  - Layout : groupbox,vbox,hbox,grid,...
  - Input: textbox, checkbox,listbox, ...
  - Window : window, dialog, ...
- definește “logica” si nu “stilul”

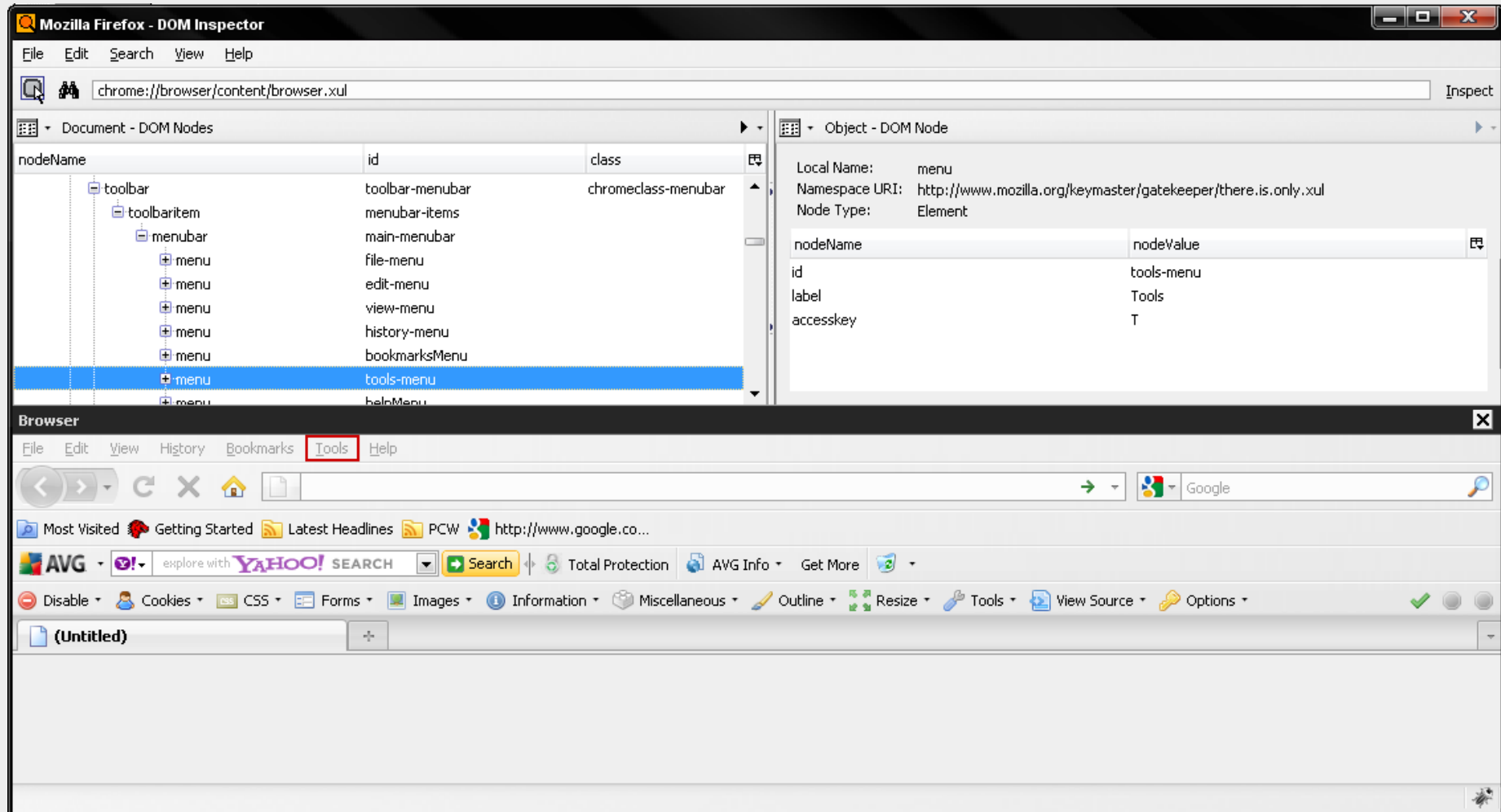
# XUL

---

```
<menubar>
  <menu label="&fileMenu.label;">
    <menupopup>
      <menuitem label="&fileMenu.new.label;" />
      <menuitem label="&fileMenu.open.label;" />
      <menuitem label="&fileMenu.save.label;" />
    </menupopup>
  </menu>
  <menu label="&editMenu.label;">
    <menupopup>
      <menuitem label="&editMenu.undo.label;" />
      <menuseparator />
      <menuitem label="&editMenu.cut.label;" />
      <menuitem label="&editMenu.paste.label;" />
      <menuitem label="&editMenu.clear.label;" />
      <menuseparator />
      <menuitem label="&editMenu.selectAll.label;" />
    </menupopup>
  </menu>
</menubar>
```



# DOM Inspector



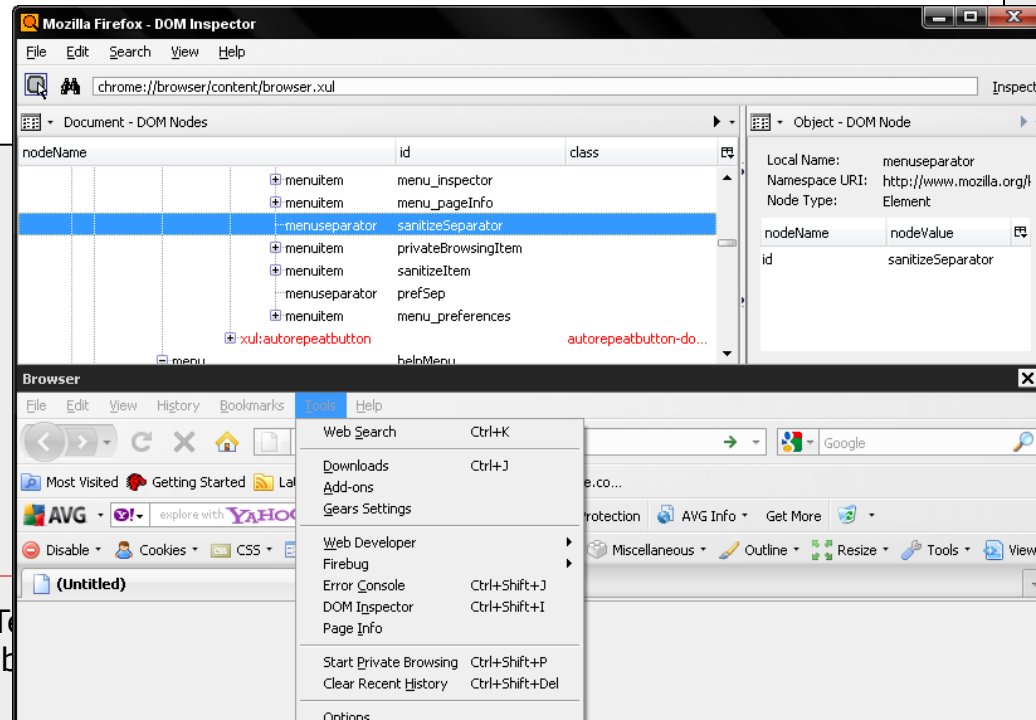


# content/menu\_overlay.xul

```
<?xml version="1.0"?>

<overlay id="simple_menuitem"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">

  <!-- Create a simple menu item -->
  <menupopup id="menu_ToolsPopup">
    <menuitem label="Talk to me" insertafter="sanitizeSeparator" />
  </menupopup>
</overlay>
```



# JavaScript

---

- ❑ Legarea se realizeaza din UI (XUL)
  - `<script`  
    `src="chrome://myextension/content/overlay.js"/>`
- ❑ Tratare evenimente aparute in UI
  - `onload`, `oncommand`, ...
- ❑ Manipulare arbore DOM
  - `window`, `document`, `content`, ...
- ❑ Comunicare cu obiecte XPCOM
  - `Preferences`, `LoginManager`, ...

# Make the extension say hello

```
<?xml version="1.0"?>
<overlay id="simple_menuitem"
        xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
<!-- Include the javascript code -->
<script type="application/x-javascript" src="chrome://myext/content/*.js" />

    <!-- Create a simple menu item -->
    <menupopup id="menu_ToolsPopup">
        <menuitem label="Talk to me" insertafter="sanitizeSeparator"
            oncommand="alert('Hello there!');" />
    </menupopup>
<!-- Add an icon to the bottom right of screen -->
<statusbar id="status-bar">
    <statusbarpanel class="statusbarpanel-iconic" id="myext_sbi"
        tooltip="&statusbar.tooltip.default;"
        onclick="myFunc();"
        src="chrome://myext/content/images/img18.png" />
</statusbar>
</overlay>
```

□ JavaScript: content/\*.js

# Localization

---

- ❑ Directoare individuale pt. fiecare idiom  
[content]\[locale]\<language>

- en-US, zh-CN, ja-JP, es-ES, ...

- ❑ Fisiererele pt. fiecare idiom constau in

- fisier DTD

- <!ENTITY menu.refresh "刷新">

- <!ENTITY menuitem.label "MyExtOptions...">

- <!ENTITY statusbar.tooltip.default "Some  
tooltip">

- fisier Properties

- signin=登录

# Icons + Styles

---

- ❑ de obicei in directorul  
[content]\[skin]
- ❑ imaginile sunt utilizate pt. icons sau grafica
- ❑ CSS ataseaza stiluri elementelor din UI. Adaugarea de stiluri unui buton:

```
#myext-toolbar-button
{
    list-style-image:
        url("chrome://myext/skin/icon24.png");
    -moz-image-region: rect(0px 24px 24px 0px);
}
```

# Piece it together

## Extension

```
+----- install.rdf
+----- chrome.manifest
```

```
+----- [content]
+----- *.xul
+----- *.js
```

```
+----- [locale]
+----- [en-us]
+----- *.dtd
+----- *.properties
```

```
+----- [skin]
+----- *.css
+----- *.png,*.jpg...
```

**Metadata**

**UI + Code**

**Localization**

**Icons + Styles**

# Exemple

---



## ☐ IE Tab

- Embeds **Internet Explorer** in a **Mozilla/Firefox tab**.

## ☐ LeechBlock

- productivity assistant tool
- automatically **block time-wasting sites** like social networks and media sites at given hours of the day

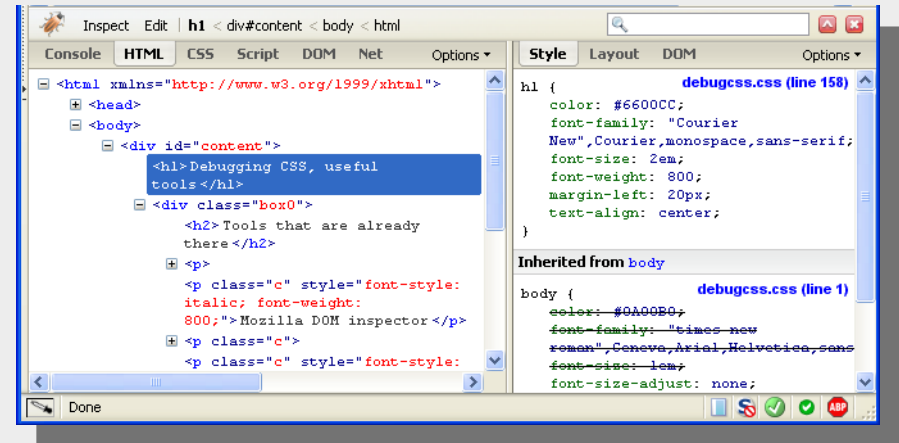
## ☐ Status-4-evar

- Show the status bar in Mozilla 11 or better.

# Exemple

## ❑ Firebug

- View, edit, and monitor CSS, HTML and Javascript live at any web pages



## ❑ TwitterFox

- Notifies you of your friends' tweets on Twitter.





# Bibliografie

---

## Extensii Mozilla

- <http://www-archive.mozilla.org/xpfe/>

## *Ce este un XUL*

- <http://andrei.clubcisco.ro/cursuri/f/f-sym/4ioc/labs/Lab%208%20XUL.pdf>

## *Metode de "accelerare" pt Mozilla*

- <https://www.youtube.com/watch?v=7B4L9mtENPE>

## *Probleme cu XPCOM*

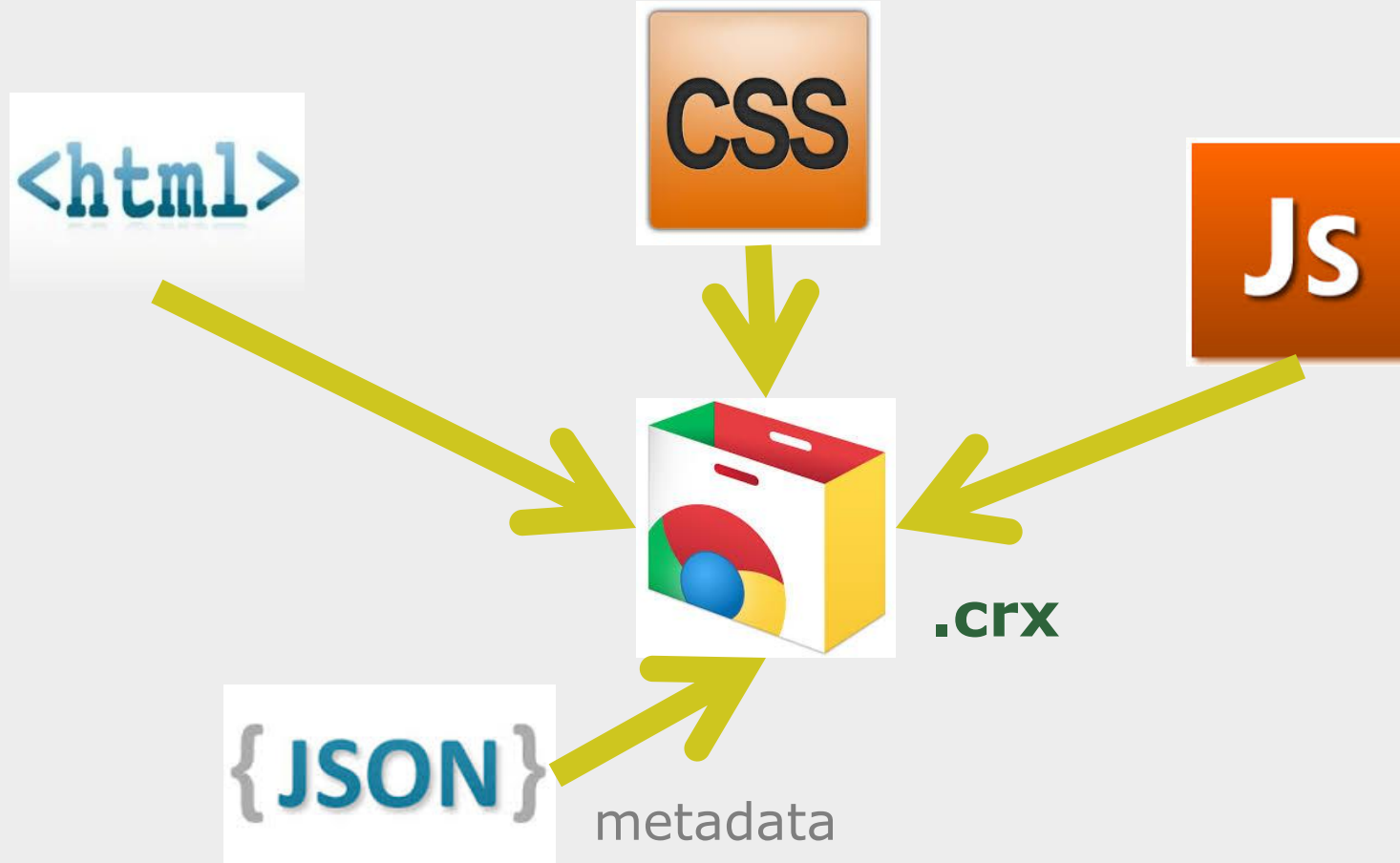
- <https://support.mozilla.org/en-US/questions/1029239>
- [https://www.youtube.com/watch?v=x\\_C2eaNj234](https://www.youtube.com/watch?v=x_C2eaNj234)

---

# Extensii Google Chrome

# Extensii Google Chrome

---



# Extensii Google Chrome

---



This **Google Mail Checker extension** uses a *browser action*.



This **Mappy extension** uses a *page action* and *content script* (code injected into a web page).



This **Set Page Color extension** features a browser action that, when clicked, shows a *popup*.

<https://developer.chrome.com/extensions/overview>

# Extensii Google Chrome

---

[https://developer.chrome.com/extensions/gets\\_tarted](https://developer.chrome.com/extensions/gets_tarted)

[https://developer.chrome.com/extensions/dev\\_guide](https://developer.chrome.com/extensions/dev_guide)

# Bibliografie

---

## Extensii Chrome

- ❑ <https://chrome.google.com/extensions>
- ❑ <http://www.go4it.ro/software/extensii-de-google-chrome-scurt-ghid-pe-intelesul-tutoror-5994491/>
- ❑ [https://www.youtube.com/watch?v=fZzO99do\\_pY](https://www.youtube.com/watch?v=fZzO99do_pY)
- ❑ <http://videotutorial.ro/cum-se-face-o-tema-pentru-browserul-google-chrome-tutorial-video/>
- ❑ <http://videotutorial.ro/setari-extensii-si-teme-pentru-cel-mai-rapid-browser-google-chrome-tutorial-video/>

---

# Extensii Internet Explorer

# Bibliografie

---

## Extensii Internet Explorer

### *Instalare/dezinstalare extensii*

- ❑ <http://www.digitalcitizen.ro/sunt-add-urile-din-internet-explorer-si-cum-functioneaza-ele>
- ❑ <http://www.digitalcitizen.ro/cum-elimini-activezi-sau-dezactivezi-add-uri-internet-explorer-11>

### *Creare extensii*

- ❑ <https://www.youtube.com/watch?v=YsB2uCF657E>
- ❑ [https://msdn.microsoft.com/en-us/library/aa753620\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa753620(v=vs.85).aspx)
- ❑ [https://msdn.microsoft.com/en-us/library/bb735854\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/bb735854(v=vs.85).aspx)
- ❑ [https://msdn.microsoft.com/en-us/library/aa753587\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa753587(v=vs.85).aspx)



# Curs 4+ - preview

---

Web programming - client-side

- Browser-ul Web
  - arhitectura, exemple,
  - extensibilitate: plugins, extensii
- tehnologii
  - **HTML, CSS**
  - Javascript
  - XML
  - AJAX
  - Web Workers, Web Storage

---

# **Limbaje de marcare**

## **- (X)HTML -**

# Limbaje de marcare

---

- ❑ set de coduri ce dau instructiuni referitoare la structura unui text si la modul de afisare a acestuia
- ❑ 1967 – *GenCode* – adnotarea manuscriselor
- ❑ 1970-1980 – TeX – carti de matematica
- ❑ Scribe, GML, SGML
- ❑ HTML, XML, XHTML
- ❑ text + instructiuni de marcare (*tag-uri*)
- ❑ prezentarea altor tipuri de informatii (imagini vectoriale, playlist-uri, servicii web, etc)

# Limbaaje de marcare

---

## **Limbaaje de marcare XML:**

- Extensible Stylesheet Language
- MathML
- MusiXML
- MXML
- RSS
- Scalable Vector Graphics
- Synchronized Multimedia Integration Language
- Web Services Description Language
- X3D
- XAML
- XHTML
- XPath
- ...

# Limbaje de marcare

---

## **Limbaje de marcare de uz general:**

- GML – predecesor al SGML
- SGML – predecesor al XML
- XML
- ...

## **Limbaje de marcare a documentelor:**

- HTML
- XHTML
- MathML
- TeX, LaTeX
- ...

## **Limbaje de marcare pentru mediatizarea conținutului:**

- Atom
- RSS
- SyncML
- ...

# Hypertext Markup Language (HTML)

---

- ❑ limbaj universal pentru publicarea hypertext-ului pe World Wide Web
- ❑ descrie continutul si structura informatiilor intr-o pagina web
- ❑ format neproprietar bazat pe SGML
- ❑ unelte de creare si redare:
  - editor text
  - unelte WYSIWYG (Dreamweaver, etc.)
- ❑ text + indicatii de redare (*tag*)
- ❑ numele fiecarui tag = *element*
  - sintaxa: `<element> continut </element>`
  - ex: `<p>Acesta este un paragraf</p>`

# Structura unei pagini HTML

---

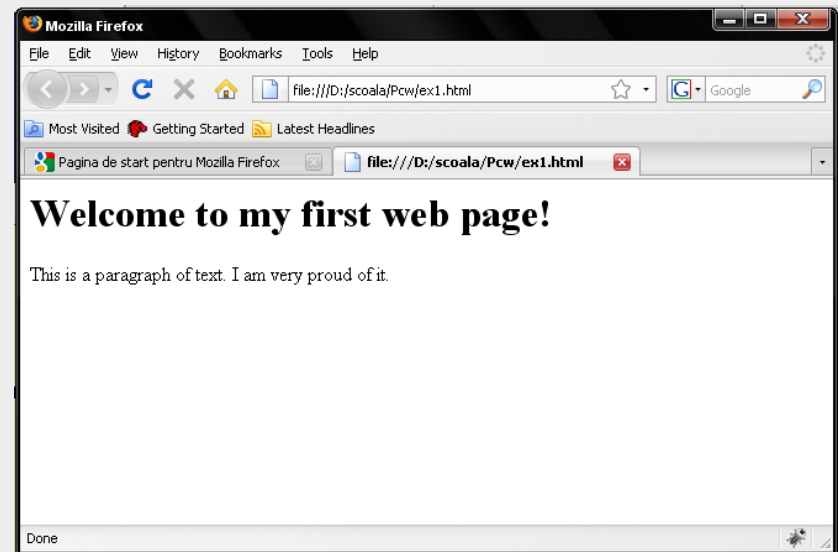
```
<html>

<head>
informatii despre pagina
</head>

<body>
continutul paginii
</body>

</html>
```

- **header** – descrie pagina
- **body** – continutul paginii
- fisier extensie **.html, .htm**



# HTML - Versiuni

---

- ❑ **GML** (IBM – Charles Goldfarb, '60) – Generalized Markup Language
- ❑ **SGML** (ISO 8879:1986) – Standard Generalized Markup Language
- ❑ **HTML Tags** – Tim Berners-Lee, 1991
- ❑ IETF – 1993: *"Hypertext Markup Language (HTML)" Internet-Draft*, T. Berners-Lee, D. Connolly
- ❑ Dave Raggett, 1993: *"HTML+ (Hypertext Markup Format)"*
- ❑ **HTML 2.0** – IETF HTML WG, 1995 (RFC 1866)
- ❑ **HTML 3.2** – Ian 1997 (W3C)
- ❑ **HTML 4.0** – Dec 1997
- ❑ **HTML 4.01** – Dec 1999
- ❑ **XHTML 1.0** – 2000
- ❑ **HTML 5**
  - 2004: Web Hypertext Application Technology Working Group (WHATWG)
  - October 2014, W3C Recommendation



# HTML vs XHTML

---

## XHTML – eXtensible HyperText Markup Language

- reformulare a HTML/4.01 in XML
- mai *extensibil* (namespaces – includerea de fragmente din alte limbaje bazate pe XML)
- *interoperabilitate* cu alte formate de date (bazate pe XML)
- divizare HTML in *componente reutilizabile* (XHTML Modularization)

## XHTML 1.0 (2000)

- procesare si mentenanta mai usoara
- elemente si attribute din HTML 4
- 3 “flavours”:
  - XHTML 1.0 *Strict* – HTML 4.01
  - XHTML 1.0 *Transitional*
    - browsere mai vechi (nu suporta style sheets)
    - include elemente prezentationale( ex. `body + bgcolor, center, font, etc.`)
  - XHTML 1.0 *Frameset*

# De ce XHTML?

---

- **99%** din paginile HTML de pe Web au macar o eroare!
  - browsere care “iarta”
- W3C: manipularea “draconica” a erorilor => XHTML
  - MIME type: *application/xhtml+xml*
  - XHTML 1.0: *application/html*
- WHAT WG (Mozilla, Opera):
  - alta viziune asupra Web-ului
  - *backward compatibility* => HTML 5
- <http://diveintohtml5.org/past.html>

# De ce XHTML?

---

- ❑ limbaj mai rigid si mai structurat
- ❑ interoperabilitate peste diferite browsere web
- ❑ mai probabil ca paginile web construite cu XHTML sa fie afisate corect in viitor
- ❑ poate fi usor interschimbat cu alte date XML: SVG, MathML, MusicML, etc.

# HTML vs XHTML

---

## Diferente:

- ❑ elementele XHTML trebuie sa fie **corect imbricate**
- ❑ elementele XHTML trebuie intotdeauna **inchise**
- ❑ elementele XHTML trebuie scrise cu **litere mici**
- ❑ documentele XHTML trebuie sa contina **un element root**

# HTML vs XHTML

---

## Diferente:

☐ elementele XHTML trebuie sa fie **corect imbricate**

☐ ele **inc** Gresit:  
`<b><i>This text is bold and italic</b></i>`

☐ ele **mi** Corect:  
`<b><i>This text is bold and italic</i></b>`

☐ documentele XHTML trebuie sa contina **un element root**

# HTML vs XHTML

---

## Diferente:

- ❑ elementele XHTML trebuie sa fie **corect imbricate**



ele

inc



ele

mi



doc

ele

```
<ul>
  <li>Cafea</li>
  <li>Ceai
    <ul>
      <li>Ceai negru</li>
      <li>Ceai verde</li>
    </ul>
  <li>Lapte</li>
</ul>
```

# HTML vs XHTML

---

## Diferente:

- ❑ elementele XHTML trebuie sa fie **corect imbricate**



ele

inc



ele

mi



doc

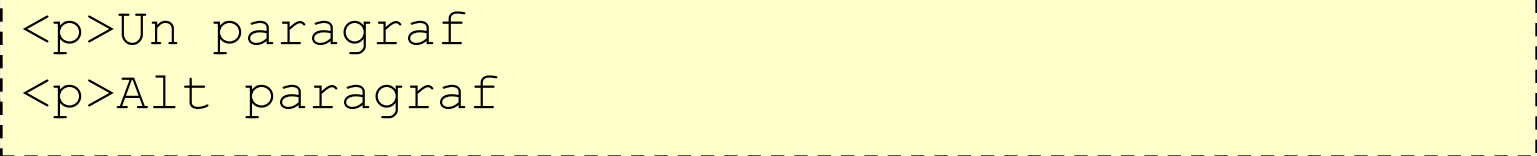
ele

```
<ul>
  <li>Cafea</li>
  <li>Ceai
    <ul>
      <li>Ceai negru</li>
      <li>Ceai verde</li>
    </ul>
  </li>
  <li>Lapte</li>
</ul>
```

# HTML vs XHTML

---

## Diferente:

- ☐ **elemente imbricate**  

- ☐ elementele XHTML trebuie intotdeauna **inchise**
- ☐ elementele XHTML trebuie scrise cu **litere mici**
- ☐ documentele XHTML trebuie sa contina **un element root**



# HTML vs XHTML

---

## Diferente:

- ☐ **elemente imbricate**  

```
<p>Un paragraf</p>  
<p>Alt paragraf</p>
```
- ☐ elementele XHTML trebuie intotdeauna **inchise**
- ☐ elementele XHTML trebuie scrise cu **litere mici**
- ☐ documentele XHTML trebuie sa contina **un element root**

# HTML vs XHTML

---

## Diferente:

- ☐ A break: `<br>`
- ☐ A horizontal rule: `<hr>`
- ☐ An image: ``
- ☐ elementele XHTML trebuie intotdeauna **inchise**
- ☐ elementele XHTML trebuie scrise cu **litere mici**
- ☐ documentele XHTML trebuie sa contina **un element root**

# HTML vs XHTML

---

## Diferente:

- ☐ A break: `<br />`
- ☐ A horizontal rule: `<hr />`
- ☐ An image: ``
- ☐ elementele XHTML trebuie intotdeauna **inchise**
- ☐ elementele XHTML trebuie scrise cu **litere mici**
- ☐ documentele XHTML trebuie sa contina **un element root**

HT

Dife

Gresit:

```
<BODY>
```

```
    <P>This is a paragraph</P>
```

```
</BODY>
```

```
<Body>
```

```
    <P>This is a paragraph</P>
```

```
</Body>
```

Corect:

```
<body>
```

```
    <p>This is a paragraph</p>
```

```
</body>
```

☐ elementele XHTML trebuie scrise cu **litere mici**

☐ documentele XHTML trebuie sa contina **un element root**

# HTML vs XHTML

---

## Diferente:

☐ elementele XHTML trebuie sa fie **corect imbricate**

☐ elementele XHTML trebuie sa fie **inchise**

☐ elementele XHTML trebuie sa fie **mici**

☐ documentele XHTML trebuie sa contina **un element root**

```
<html>
<head> ... </head>
<body> ... </body>
</html>
```

# HTML vs XHTML

---

Alte reguli de sintaxa XHTML:

- ❑ Numele atributelor trebuie sa fie scrise cu **litere mici**
- ❑ Valorile atributelor trebuie sa fie scrise intre **ghilimele**
- ❑ Minimizarea atributelor este **interzisa**
- ❑ Atributul *id* **inlocuieste** atributul *name*
- ❑ XHTML DTD defineste elemente **obligatorii**

# HTML vs XHTML

---

Alte reguli de sintaxa XHTML:

- ❑ Numele atributelor trebuie sa fie scrise cu **litere mici**
- ❑ Valorile atributelor trebuie sa fie scrise intre **ghilimele**
- ❑ Minim: 

Gresit:	<code>&lt;table WIDTH="100%"&gt;</code>
Corect:	<code>&lt;table width="100%"&gt;</code>
- ❑ Atributul `width` trebuie scris cu **litere mici**
- ❑ XHTML **NU** definește elemente **obligatorii**

# HTML vs XHTML

---

Alte reguli de sintaxa XHTML:

- ❑ Numele atributelor trebuie sa fie scrise cu **litere mici**
- ❑ Valorile atributelor trebuie sa fie scrise intre **ghilimele**
- ❑ Minim: **Gresit:**  
`<table width=100%>`  
**Corect:**  
`<table width="100%">`
- ❑ Atribu
- ❑ XHTML



# HTML vs XHTML

---

## Alte reguli de sintaxa XHTML:

### Gresit:

```
<input checked>  
<input readonly>  
<input disabled>  
<option selected>
```

### Corect:

```
<input checked="checked"/>  
<input readonly="readonly"/>  
<input disabled="disabled"/>  
<option selected="selected"/>
```

❑ Valorile atributelor  
**ghilimele**

❑ Minimizarea atributelor este **interzisă**

❑ Atributul *id* **înlocuiește** atributul *name*

❑ XHTML DTD definește elemente **obligatorii**

# HTML vs XHTML

## Alte reguli de sintaxa XHTML:

Gresit:

<input  
<input  
<input  
<option

□ Val  
intr

□ Min

□ Atri

□ XH

HTML	XHTML
compact	compact="compact"
checked	checked="checked"
declare	declare="declare"
readonly	readonly="readonly"
disabled	disabled="disabled"
selected	selected="selected"
defer	defer="defer"
ismap	ismap="ismap"
nohref	nohref="nohref"
noshade	noshade="noshade"
nowrap	nowrap="nowrap"
multiple	multiple="multiple"
noresize	noresize="noresize"

/>  
/>  
"/>

orii

# HTML vs XHTML

---

## Alte reguli de sintaxa XHTML:

Gresit:

```

```

Corect:

```

```

```

```

## ghilimele

- ❑ Minimizarea atributelor este **interzisa**
- ❑ Atributul *id* **inlocuieste** atributul *name*
- ❑ XHTML DTD definește elemente **obligatorii**

# HTML vs XHTML

---

Alte reguli de sintaxa XHTML:

- ❑ Numele atributelor trebuie sa fie scrise cu **litere mici**
- ❑ Valorile atributelor trebuie sa fie scrise intre **ghilimele**
- ❑ ~~Minimizarea atributelor este interzisa~~  
DOCTYPE, head, title, body
- ❑ ~~Atributul `xmlns` trebuie sa aiba valoarea `http://www.w3.org/1999/xhtml`~~
- ❑ XHTML DTD defineste elemente **obligatorii**

# Structura unui document XHTML

---

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd" >

<html xmlns="http://www.w3.org/1999/xhtml">

<head>
    <title>Titlul paginii</title>
</head>

<body>
</body>

</html>
```

XHTML DTD (Document Type Definition)

- obligatorie (prima linie dintr-un document XHTML)
- nu este element XHTML
- descrie sintaxa permisa pentru marcasele XHTML (Strict, Transitional, Frameset)

# Validare pagini XHTML

- ❑ trebuie adaugata definitia DTD ca prima linie a documentului
- ❑ W3C XHTML Validator, Tidy

The screenshot shows the W3C Markup Validation Service interface. The browser title is "The W3C Markup Validation Service - Mozilla Firefox". The address bar shows "http://validator.w3.org/#validate\_by\_uri-with\_options". The main heading is "W3C® Markup Validation Service" with the tagline "Check the markup (HTML, XHTML, ...) of Web documents". There are three tabs: "Validate by URI", "Validate by File Upload", and "Validate by Direct Input". The "Validate by URI" tab is active. Below it, there's a "Validate by URI" section with a text input field for the address, containing "www.ace.tuiasi.ro". A "More Options" section is expanded, showing various settings: "Character Encoding" set to "(detect automatically)", "Document Type" set to "(detect automatically)", "List Messages Sequentially" selected, "Show Source" checked, "Show Outline" checked, "Clean up Markup with HTML Tidy" checked, "Validate error pages" checked, and "Verbose Output" checked. A "Check" button is at the bottom.

The screenshot shows the W3C Markup Validator results page. The browser title is "[Invalid] Markup Validation of http://www.ace.tuiasi.ro/ - W3C Markup Validator - Mozilla Firefox". The address bar shows "http://validator.w3.org/check?uri=www.ace.tuiasi.ro&charset=(detect+automatically)&doctype=Inline&group=". The main heading is "Errors found while checking this document as XHTML 1.0 Transitional!". Below it, a table shows the validation results: "Result: 6 Errors, 1 warning(s)", "Address: http://www.ace.tuiasi.ro/", "Encoding: utf-8", "Doctype: XHTML 1.0 Transitional", "Root Element: html", and "Root Namespace: http://www.w3.org/1999/xhtml". An "Options" section is visible with checkboxes for "Show Source", "Show Outline", "List Messages Sequentially" (selected), "Group Error Messages by Type", "Validate error pages", "Verbose Output", and "Clean up Markup with HTML Tidy". A "Revalidate" button is at the bottom right. A "Potential Issues" section is also visible, stating "No DOCTYPE found! Checking with default XHTML 1.0 Transitional Document Type." and explaining that no DOCTYPE Declaration could be found or recognized in this document.

side.

# HTML 5

---

## Premisa:

- WWW uses a mixture of features introduced by
  - various specifications,
  - software products such as web browsers
  - common practice
  
- detailed processing models to encourage more interoperable implementations
- extends, improves and rationalizes HTML
- markup and APIs for complex web applications
- features for web apps running on low-powered devices (smartphones, tablets)

# HTML 5

---

## Features:

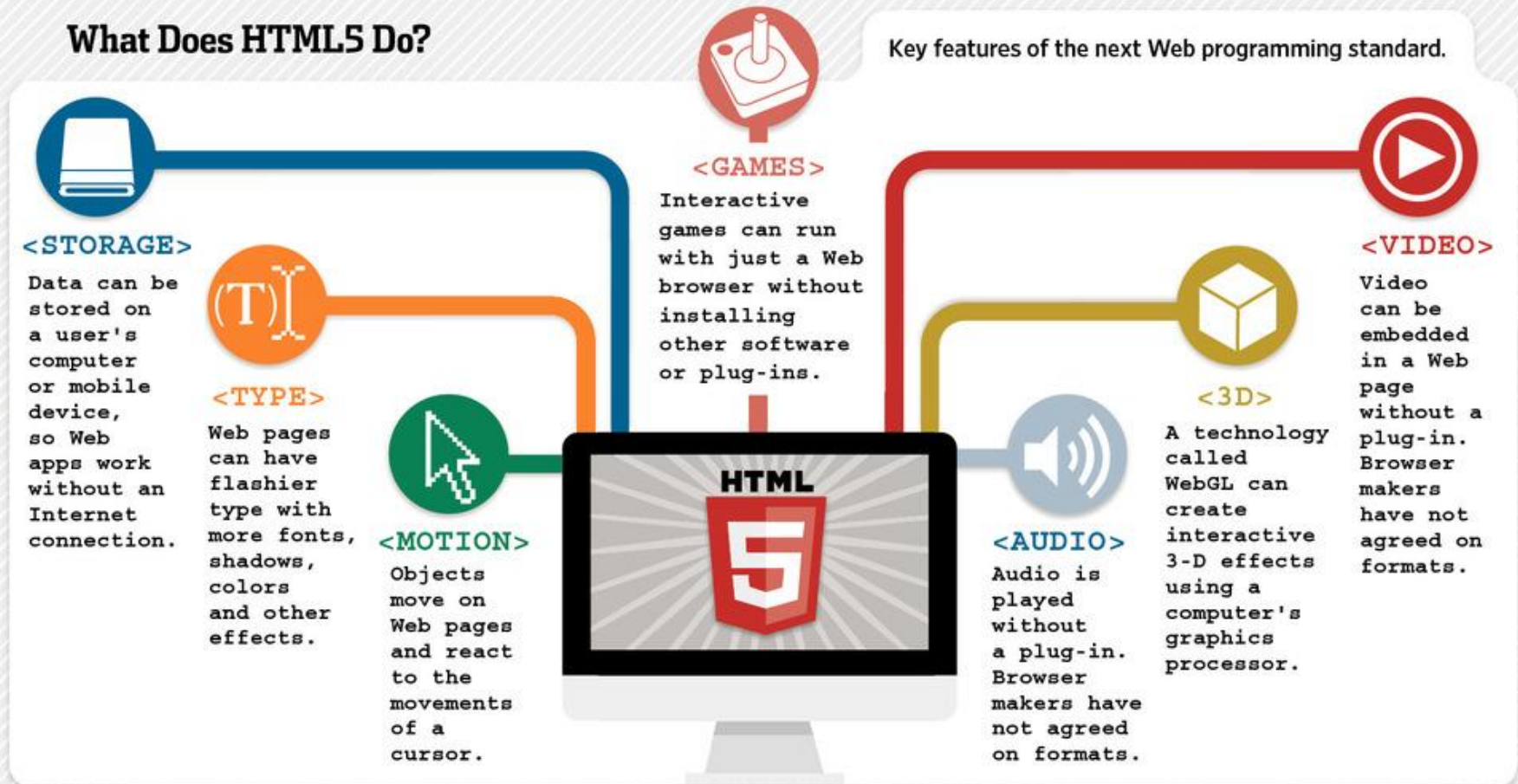
- syntactic – to make it easy to include and handle multimedia and graphical content on the web
  - <video>, <audio>, <canvas>
- structural - to enrich the semantic content of documents
  - <main>, <section>, <article>, <header>, etc.
- new attributes
  - charset (<meta>), async (<script>)
- inline SVG and MathML in *text/html*
- APIs and DOM – integrated in the specs



# HTML 5; 5.1 sept 2016; 5.2 -2017

## What Does HTML5 Do?

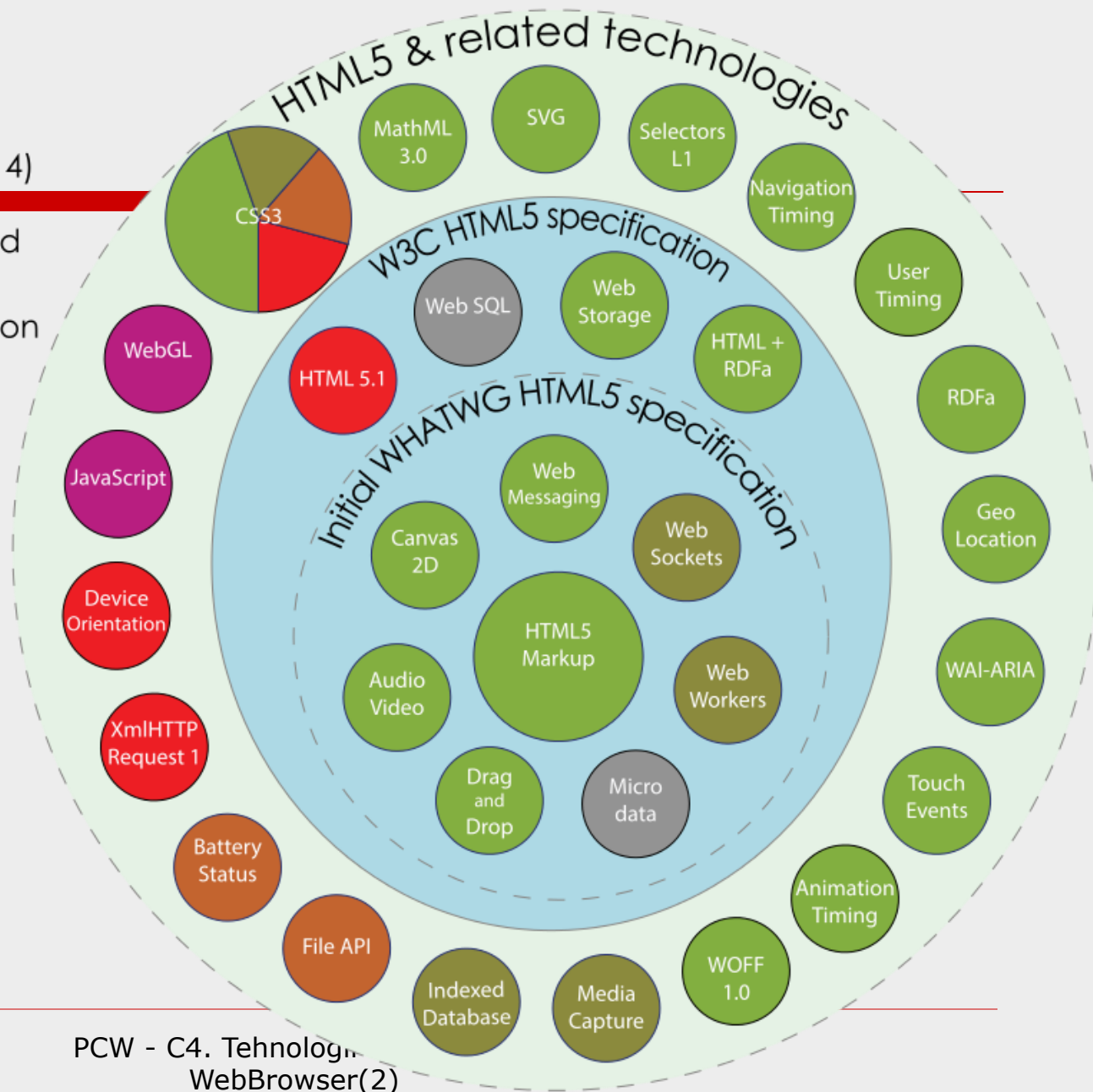
Key features of the next Web programming standard.



# HTML5

Taxonomy & Status (October 2014)

- Recommendation/Proposed
- Candidate Recommendation
- Last Call
- Working Draft
- Non-W3C Specifications
- Deprecated or inactive



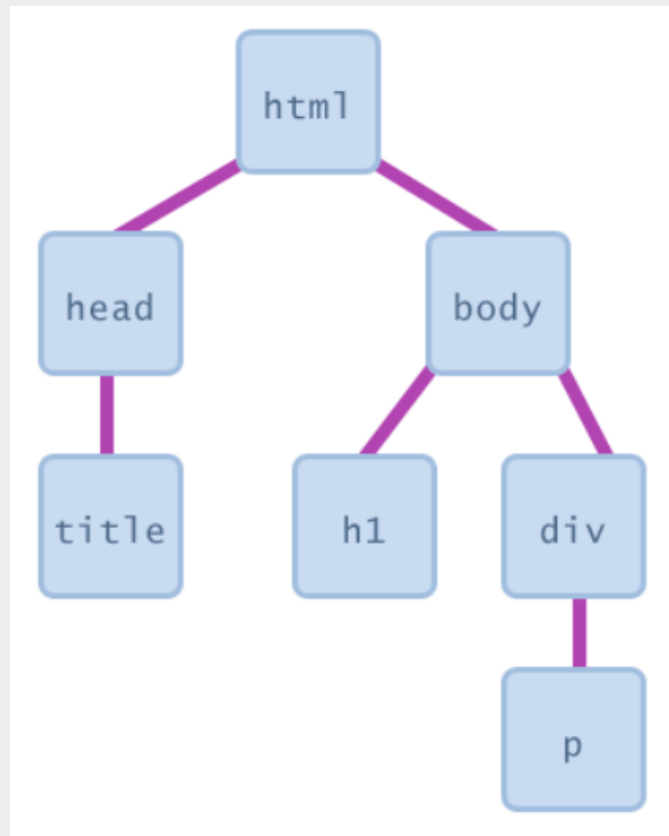
---

# DOM/HTML

# Document Object Model (DOM)

---

- ❑ o reprezentare a documentelor (X)HTML/XML sub forma unui set de obiecte
- ❑ cross-platform
- ❑ language independent
- ❑ permite accesarea si modificarea *dinamica* a *continutului*, *structurii* si *stilului* unui document



# Document Object Model (DOM)

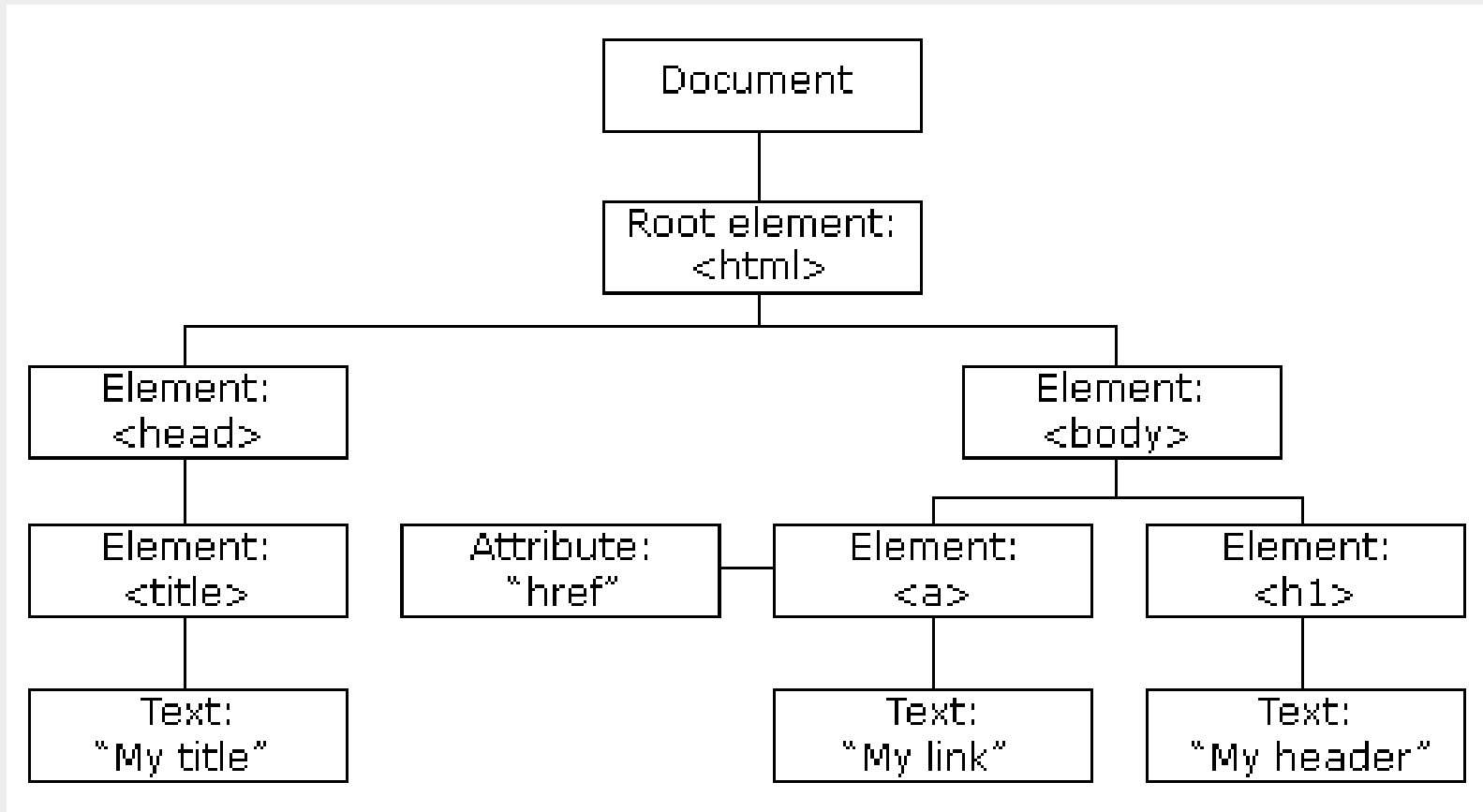
---

- Desenati arborele DOM al urmatorului document HTML :

```
<html>
  <head>
    <title>My Title</title>
  </head>
  <body>
    <p>Some text</p>
    <a href="link_url">My
link</a>
    <h1>My header</h1>
  </body>
</html>
```

# DOM/HTML

---



# Document Object Model (DOM)

---

- ❑ Standardizare (W3C) – 3 parti:
  - Core DOM – model standard pt. orice document structurat
  - XML DOM – model standard pt. documente XML
  - HTML DOM - model standard pt. documente HTML
  
- ❑ DOM Level 1 (1998)
  - core elements
- ❑ DOM Level 2 (2000)
  - getElementById
  - event model
  - suport pt. spatii de nume XML
  - suport pt. CSS
- ❑ DOM Level 3 (2004)
  - suport pt. Xpath
  - tratare evenimente tastatura
  - interfata pt. serializarea documentelor in format XML
- ❑ DOM Level 4 (2015 – Proposed Recommendation)

# Obiecte DOM globale

---

- ❑ window : fereastra browser-ului
- ❑ navigator : info despre browser-ul folosit
- ❑ screen : info despre zona de pe ecran ocupata de browser
- ❑ history : lista a paginilor vizitate de utilizator
- ❑ document : pagina HTML curenta



# Tratarea evenimentelor

---

```
<h2 onclick="myFunction();" >Click me!</h2>
```

- elementele HTML au attribute speciale denumite *evenimente*
- funcțiile Javascript pot fi utilizate pentru tratarea evenimentelor
  - funcția se va executa la interacțiunea cu elementul respectiv
  - attribute HTML de tip eveniment :  
[onabort](#), [onblur](#), [onchange](#), [onclick](#), [ondblclick](#),  
[onerror](#), [onfocus](#), [onkeydown](#), [onkeypress](#), [onkeyup](#),  
[onload](#)

# document object si getElementById

---

```
<h2 onclick="makeRed();">Sell</h2>
```

```
<p id="announce">Get it while it's hot!</p>
```

```
function makeRed() {  
    var para = document.getElementById("announce");  
    para.style.color = "red";  
}
```

**Sell**

Get it while it's hot!

- ❑ metoda `getElementById` a obiectului `document` returneaza un obiect reprezentand un element HTML cu atributul `id` furnizat (`null` daca nu este gasit)
- ❑ obiectele DOM pentru toate elementele HTML contin urmatoarele proprietati:  
**`className`, `id`, `style`, `title`**

# DOM/HTML

## □ Prop <html>

- X. <body>

- X.

- X. <p id="intro">Hello World!</p>

- X.

- X. <script type="text/javascript">

```
document.getElementById("intro").childNodes[0].nodeValue;
```

```
document.write("<p>The text from the intro
```

## □ Metc paragraph: " + txt + "</p>");

- X. </script>

- X.

```
Script </body>
```

- X. </html>

- X.removeChild(node) - remove a child node from x

# Bibliografie

---

## HTML 5.1

- <https://www.w3.org/TR/html/>
- <http://dhark.com/html5intro/index.html>

## HTML 5.2

- <https://w3c.github.io/html/introduction.html>