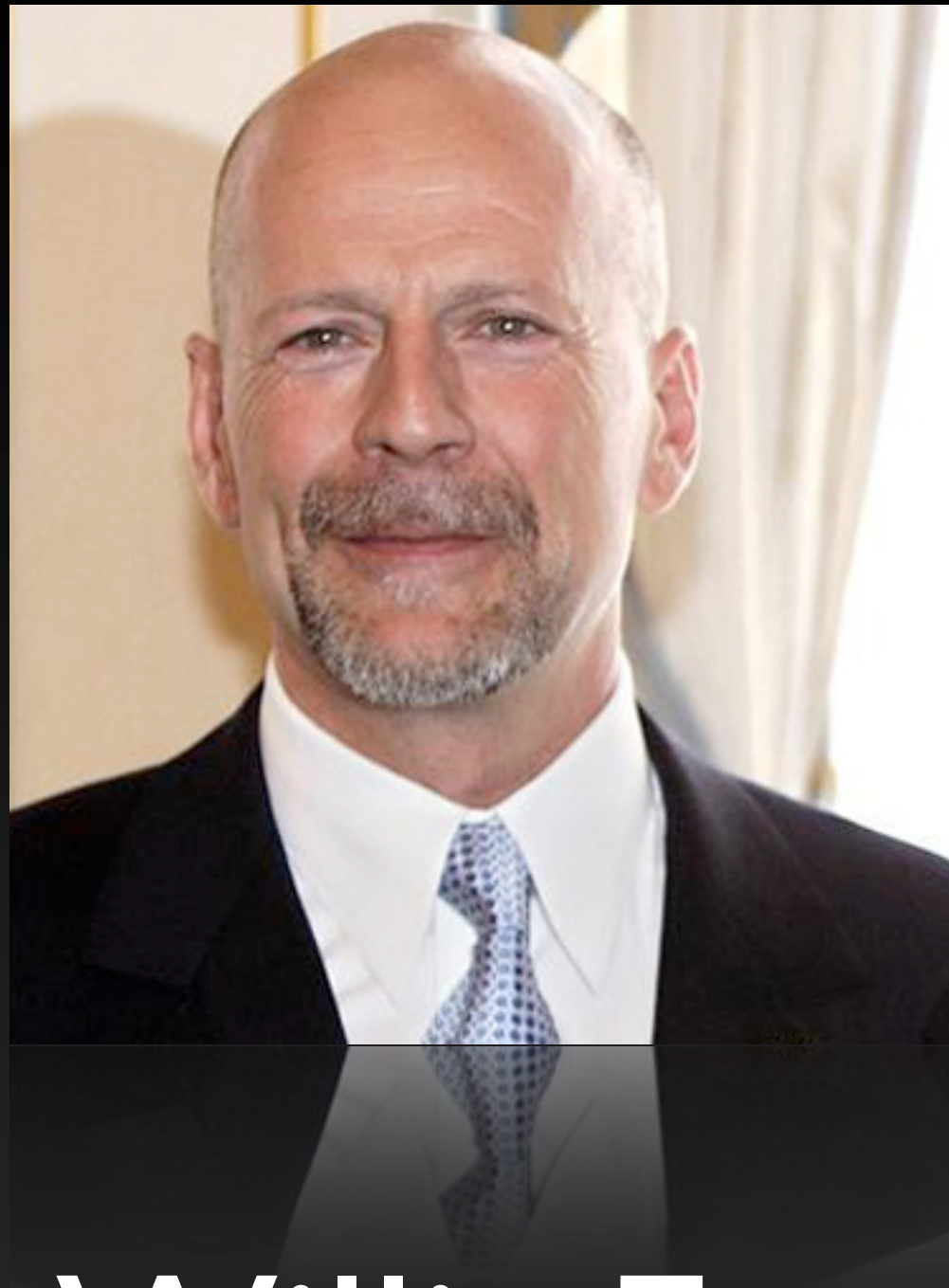


# How to Write Your First Extension



# Bruce Willis Extension

(Francois Mori/AP Photo)

# How to Write Your First Extension

1. Create development profile
2. Configuration settings
3. Pointing extension to your dev directory
4. Creating folder structure & files
5. Packaging & installing
6. Distributing your add-on

# Create Development Profile

Keep a separate Firefox profile for  
extension development

## **Windows:**

Start menu > Run (Windows key + R).

Write `firefox -P` and press click OK.

## **Mac OS X** (in Terminal):

Type in `/Applications/Firefox.app/  
Contents/MacOS/firefox -  
profilemanager`

## **Linux** (in a terminal):

Use `cd` to navigate to your Firefox directory and then enter `./firefox -profilemanager`



## Introduction

Firefox stores information about your settings and preferences in your personal profile.

If you are sharing this copy of Firefox with other users, you can use profiles to keep each user's information separate. To do this, each user should create his or her own profile.

If you are the only person using this copy of Firefox, you must have at least one profile. If you would like, you can create multiple profiles for yourself to store different sets of settings and preferences. For example, you may want to have separate profiles for business and personal use.

To begin creating your profile, click Continue.

Cancel

Go Back

Continue

Cancel

Go Back

Continue

To begin creating your profile, click Continue.



## Conclusion

If you create several profiles you can tell them apart by the profile names. You may use the name provided here or use one of your own.

Enter new profile name:

dev

Your user settings, preferences and other user-related data will be stored in:

/Users/robertnyman/Library/Application Support/Firefox/Profiles  
/5m3z8wx9.dev

Choose Folder...

Use Default Folder

Click Done to create this new profile.

Cancel

Go Back

Done

CANCEL

GO BACK

DONE

Click Done to create this new profile.

# Pointing Extension to Your Development Directory

Find your Profile directory

## **Windows:**

Win XP: C:\Documents and Settings\[user name]  
\Application Data\Mozilla\Firefox\Profiles

Vista: C:\Users\[user name]\AppData\Roaming

**Mac OS X** (~ = /Users/[user name]):

~/Library/Application Support/Firefox/Profiles/

## **Linux:**

~/.mozilla/

Go to [Dev Profile]/extensions/ folder

Create empty file (no file extension!):

`brucewillis@robertnyman.com`



Point to your extension code, i.e. file path:

**Windows:**

E.g: `C:\dev\brucewillis\`

**Mac/Linux:**

E.g: `~/dev/brucewillis/`



# Configuration Settings

1. Open Firefox:

- Have your development profile as default
- or Through Profile Manager

2. Type in `about:config`



### This might void your warranty!

Changing these advanced settings can be harmful to the stability, security, and performance of this application. You should only continue if you are sure of what you are doing.

☒ Show this warning next time

I'll be careful, I promise!

## Recommended settings:

```
javascript.options.showInConsole = true
```

```
nglayout.debug.disable_xul_cache = true
```

```
browser.dom.window.dump.enabled = true
```

- Optional settings:

```
javascript.options.strict = true
```

```
extensions.logging.enabled = true
```

Filter: <input type="text" value="javascript.options.showInConsole"/>				<input type="button" value="Show All"/>
Preference Name	Status	Type	Value	
javascript.options.showInConsole	user set	boolean	true	

Copy  
Copy Name  
Copy Value

New ▶

Toggle

Reset











String




Integer

Boolean

# Creating Folder Structure & Files



- ▼ chrome
  - ▼ content
    -  browser.xul
    -  bruceWillis.js
    -  preferences.xul
  - ▼ locale
    - ▼ en-US
      -  translations.dtd
  - ▼ skin
    -  skin.css
    -  status-bar.png
    -  toolbar-button.png
  -  chrome.manifest
- ▼ defaults
  - ▼ preferences
    -  prefs.js
  -  install.rdf

 install.rdf  
 prefs.js  
 preferences

- Automatic creation of extension
- Extension Developer extension

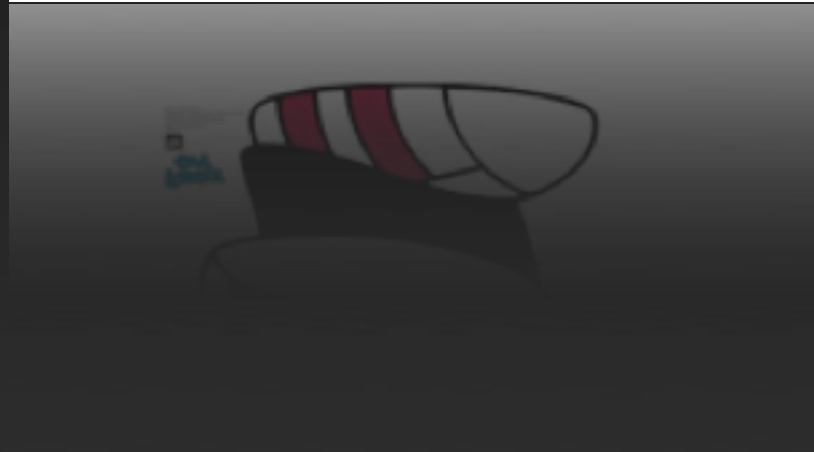
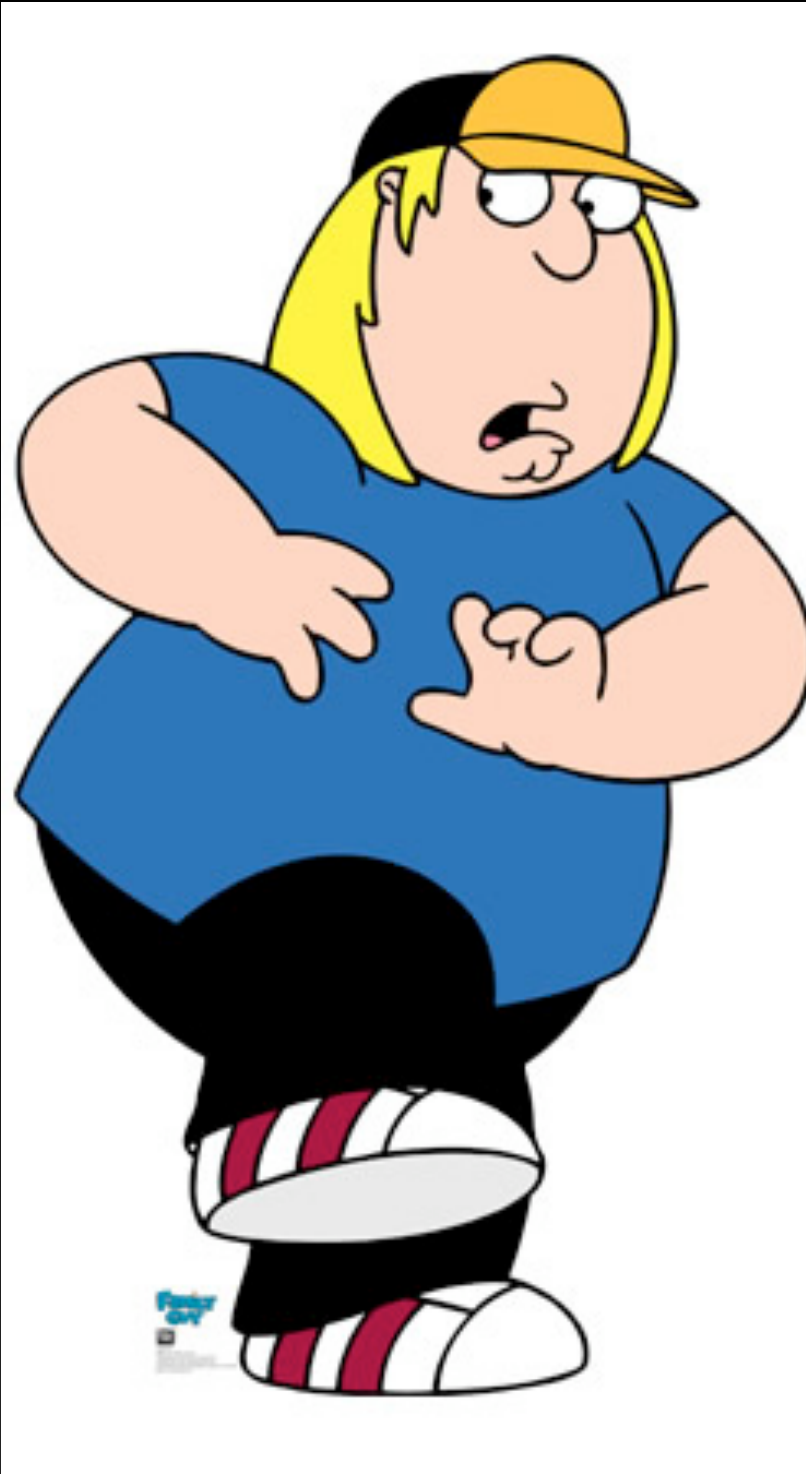
**install.rdf**

Information, IDs and metadata

```
<?xml version="1.0"?>
<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>brucewillis@robertnyman.com</em:id>
    <em:name>Bruce Willis demo extension</em:name>
    <em:version>1.0</em:version>
    <em:type>2</em:type>
    <em:creator>Robert Nyman</em:creator>
    <em:description>Finds document headings and replaces them with Die
Hard movie titles</em:description>
    <em:homepageURL>http://www.robertnyman.com/</em:homepageURL>
    <em:optionsURL>chrome://brucewillis/content/preferences.xul</
em:optionsURL>

    <em:targetApplication>
      <Description>
        <em:id>{ec8030f7-c20a-464f-9b0e-13a3a9e97384}</em:id>
        <em:minVersion>2.0</em:minVersion>
        <em:maxVersion>3.1b2</em:maxVersion>
      </Description>
    </em:targetApplication>
  </Description>
</RDF>
```



In the Description node

**em:id**

Your unique developer id, of your own choosing. Has to be the same as the pointer file you previously created, i.e. `brucewillis@robertnyman.com`

**em:name**

The name of your extension.

**em:version**

Current version of your extension.

**em:type**

The type declares that is an extension, as opposed to, for instance, a theme.

**em:creator**

You!

**em:description**

Describes your extension functionality. Will be shown in the Tools > Add-ons window.

**em:homepageURL**

The URL of your extension's web site.

**em:optionsURL**

The URL to where you will have your file for editing options/preferences.



In the Description/  
em:TargetApplication node

**em:id**

The actual id of Firefox: {ec8030f7-c20a-464f-9b0e-13a3a9e97384}. Exchange this to develop for another app, like Thunderbird.

**em:minVersion**

The minimum version number of Firefox to run the extension.

**em:maxVersion**

The maximum version number of Firefox to run the extension.

Valid alternatives for Firefox, Thunderbird etc and their corresponding versions

```
<?xml version="1.0"?>
<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>brucewillis@robertnyman.com</em:id>
        <em:name>Bruce Willis demo extension</em:name>
        <em:version>1.0</em:version>
        <em:type>2</em:type>
        <em:creator>Robert Nyman</em:creator>
        <em:description>Finds document headings and replaces them with Die
Hard movie titles</em:description>
        <em:homepageURL>http://www.robertnyman.com/</em:homepageURL>
        <em:optionsURL>chrome://brucewillis/content/preferences.xul</
em:optionsURL>

        <em:targetApplication>
            <Description>
                <em:id>{ec8030f7-c20a-464f-9b0e-13a3a9e97384}</em:id>
                <em:minVersion>2.0</em:minVersion>
                <em:maxVersion>3.1b2</em:maxVersion>
            </Description>
        </em:targetApplication>
    </Description>
</RDF>
```

More information in Install Manifests

**chrome.manifest**

# File references and usage

**content**      brucewillis      chrome/content/

**content**      brucewillis      chrome/content/ contentaccessible=yes

overlay chrome://browser/content/browser.xul chrome://  
brucewillis/content/browser.xul

**locale** brucewillis    en-US    chrome/locale/en-US/

**skin**    brucewillis    classic/1.0    chrome/skin/

style chrome://global/content/customizeToolbar.xul chrome://  
brucewillis/skin/skin.css

```
# Content pointer
```

```
content      brucewillis      chrome/content/
```

```
# Make content accessible from web pages in Firefox 3
```

```
content      brucewillis      chrome/content/ contentaccessible=yes
```

```
# Overlay browser skin
```

```
overlay chrome://browser/content/browser.xul chrome://  
brucewillis/content/browser.xul
```



```
# Language versions
```

```
locale brucewillis en-US chrome/locale/en-US/
```

```
# Setting reference to extension skin
```

```
skin    brucewillis classic/1.0 chrome/skin/
```

```
# Adding CSS to available toolbar buttons
```

```
style chrome://global/content/customizeToolbar.xul
```

```
chrome://brucewillis/skin/skin.css
```

More information in [Chrome Manifest](#)

Are you with  
me so far?

**chrome content folder**

XUL, JavaScript and content CSS

- `browser.xul`
- `bruce-willlis.css`
- `bruceWilllis.js`
- `preferences.xul`

- `browser.xul`
- `bruce-willlis.css`
- `bruceWilllis.js`
- `preferences.xul`



```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://brucewillis/skin/skin.css" type="text/
css"?>
<!DOCTYPE brucewillis SYSTEM "chrome://brucewillis/locale/
translations.dtd">
<overlay id="sample" xmlns="http://www.mozilla.org/keymaster/
gatekeeper/there.is.only.xul">
    <script src="bruceWillis.js" />

    <menupopup id="menu_ToolsPopup">
        <menuitem label="&runbrucewillis;" key="brucewillis-run-
key" oncommand="linkTargetFinder.run()" />
    </menupopup>

    <keyset>
        <key id="brucewillis-run-key" modifiers="accel alt shift"
key="B" oncommand="bruceWillis.run()" />
    </keyset>

    <statusbar id="status-bar">
        <statusbarpanel id="brucewillis-status-bar-icon"
class="statusbarpanel-iconic" src="chrome://brucewillis/skin/status-
bar.png" tooltiptext="&runbrucewillis;" onclick="bruceWillis.run()" />
    </statusbar>


    <toolbarpalette id="BrowserToolbarPalette">
        <toolbarbutton id="brucewillis-toolbar-button"
label="Bruce Willis" tooltiptext="&runbrucewillis;"
oncommand="bruceWillis.run()" />
    </toolbarpalette>
</overlay>
```

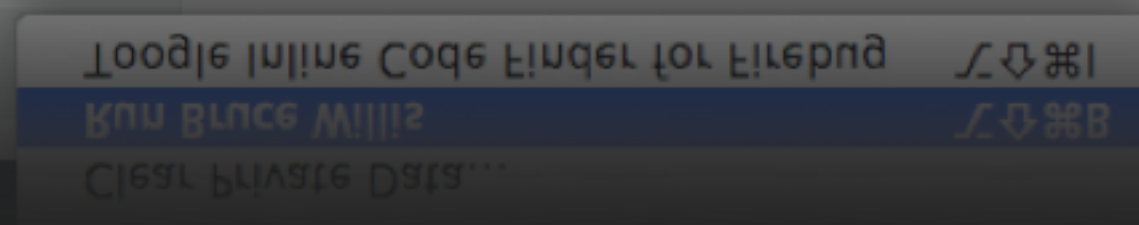
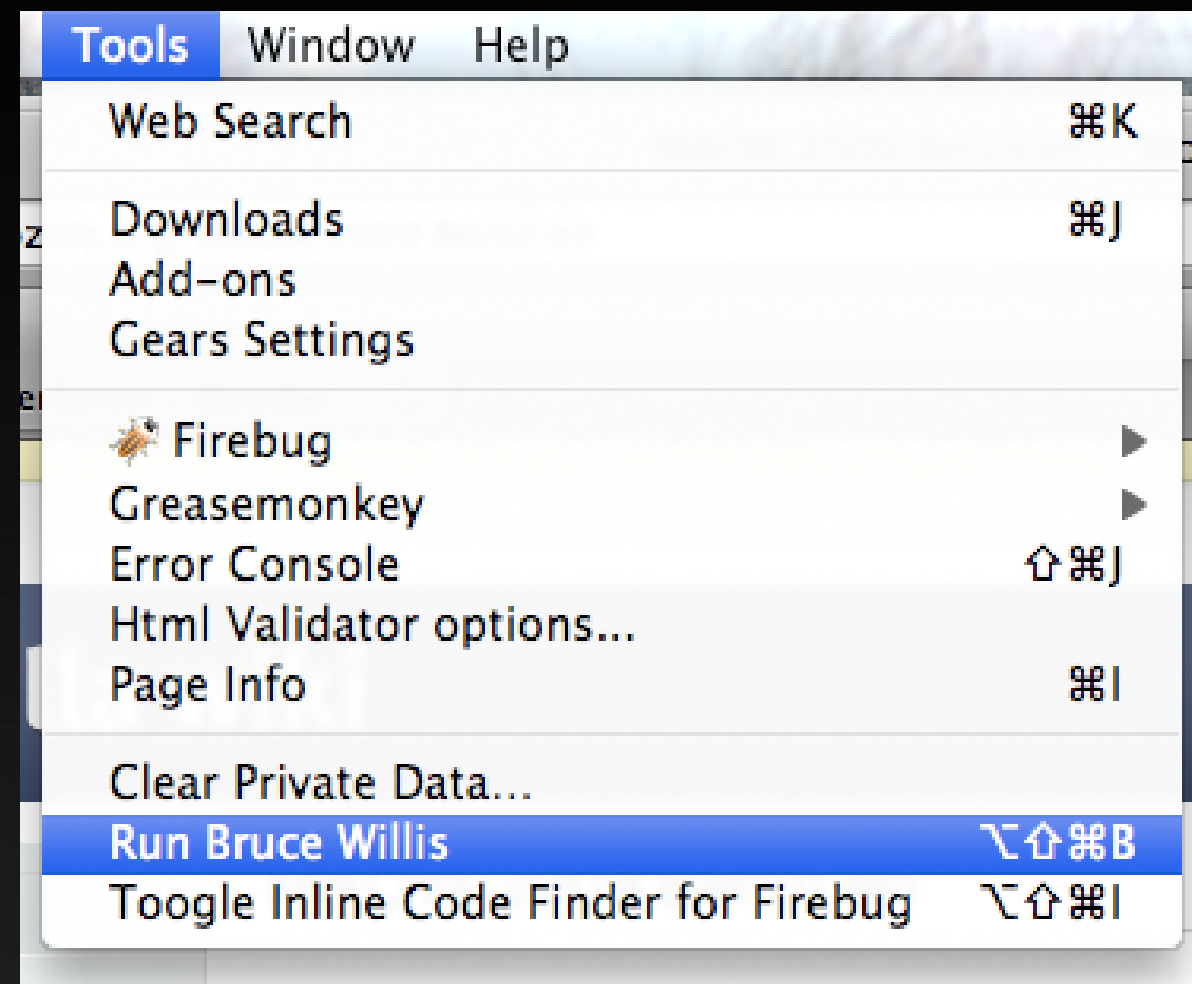
```
<?xml version="1.0"?>  
<?xml-stylesheet href="chrome://brucewillis/skin/skin.css" type="text/  
css"?>  
<!DOCTYPE brucewillis SYSTEM "chrome://brucewillis/locale/  
translations.dtd">
```

```
<overlay id="sample" xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">  
  <script src="bruceWillis.js" />  
  <!-- Content -->  
</overlay>
```

```
<!-- Tools menu option and keyboard shortcut -->
<menupopup id="menu_ToolsPopup">
  <menuitem label="&runbrucewillis;" key="brucewillis-run-
key" oncommand="linkTargetFinder.run()" />
</menupopup>

<keyset>
  <key id="brucewillis-run-key" modifiers="accel alt shift"
key="B" oncommand="bruceWillis.run()" />
</keyset>
```





```
<!-- Statusbar icon -->  
<statusbar id="status-bar">  
    <statusbarpanel id="brucewillis-status-bar-icon"  
class="statusbarpanel-iconic" src="chrome://brucewillis/skin/status-  
bar.png" tooltiptext="&runbrucewillis;" onclick="bruceWillis.run()" />  
</statusbar>
```

wiki.mozilla.org



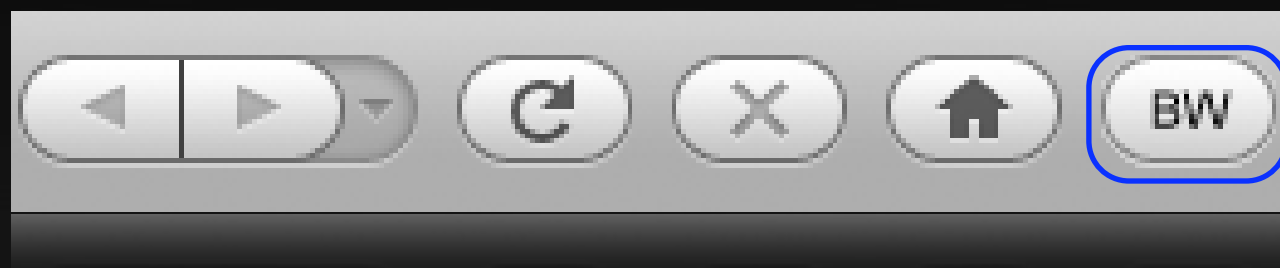
Bruce Willis



0 errors / 0 warnings

```
<!-- Firefox toolbar button -->  
<toolbarpalette id="BrowserToolbarPalette">  
    <toolbarbutton id="brucewillis-toolbar-button"  
label="Bruce Willis" tooltiptext="&runbrucewillis;"  
oncommand="bruceWillis.run()" />  
</toolbarpalette>
```





- `browser.xul`
- `bruce-willlis.css`
- `bruceWilllis.js`
- `preferences.xul`

```
h1, h2, h3, h4 {  
    font-family: Georgia, Times, serif;  
}
```

- `browser.xul`
- `bruce-willlis.css`
- `bruceWilllis.js`
- `preferences.xul`

```

var bruceWillis = function () {
    var prefManager = Components.classes["@mozilla.org/preferences-service;
1"].getService(Components.interfaces.nsIPrefBranch);
    return {
        init : function () {
            gBrowser.addEventListener("load", function () {
                var autoRun = prefManager.getBoolPref("extensions.brucewillis.autorun");
                if (autoRun) {
                    bruceWillis.run();
                }
            }, false);
        },

        run : function () {
            var head = content.document.getElementsByTagName("head")[0],
                style = content.document.getElementById("bruce-willis-style"),
                h1 = content.document.getElementsByTagName("h1"),
                h2 = content.document.getElementsByTagName("h2"),
                h3 = content.document.getElementsByTagName("h3"),
                h4 = content.document.getElementsByTagName("h4");

            if (!style) {
                style = content.document.createElement("link");
                style.id = "brucewillis-style";
                style.type = "text/css";
                style.rel = "stylesheet";
                style.href = "chrome://brucewillis/content/bruce-willis.css";
                head.appendChild(style);
            }

            for (var i=0, il=h1.length; i<il; i++) {
                h1[i].textContent = "Die Hard 1";
            }

            for (var i=0, il=h2.length; i<il; i++) {
                h2[i].textContent = "Die Hard 2";
            }

            for (var i=0, il=h3.length; i<il; i++) {
                h3[i].textContent = "Die Hard 3";
            }

            for (var i=0, il=h4.length; i<il; i++) {
                h4[i].textContent = "Die Hard 4";
            }
        }
    };
}();
window.addEventListener("DOMContentLoaded", bruceWillis.init, false);

```

```
// Structure - Yahoo JavaScript Module Pattern
var bruceWillis = function () {
    var prefManager = Components.classes["@mozilla.org/preferences-
service;1"].getService(Components.interfaces.nsIPrefBranch);
    return {
        init : function () {
            // init method
        },

        run : function () {
            // run method
        }
    };
}();
window.addEventListener("DOMContentLoaded", bruceWillis.init, false);
```

```
// init method
init : function () {
    gBrowser.addEventListener("load", function () {
        var autoRun =
prefManager.getBoolPref("extensions.brucewillis.autorun");
        if (autoRun) {
            bruceWillis.run();
        }
    }, false);
}
```

```
// run method
run : function () {
    var head = content.document.getElementsByTagName("head")[0],
        style = content.document.getElementById("bruce-willis-style"),
        h1 = content.document.getElementsByTagName("h1"),
        h2 = content.document.getElementsByTagName("h2"),
        h3 = content.document.getElementsByTagName("h3"),
        h4 = content.document.getElementsByTagName("h4");

    if (!style) {
        style = content.document.createElement("link");
        style.id = "brucewillis-style";
        style.type = "text/css";
        style.rel = "stylesheet";
        style.href = "chrome://brucewillis/content/bruce-willis.css";
        head.appendChild(style);
    }

    for (var i=0, il=h1.length; i<il; i++) {
        h1[i].textContent = "Die Hard 1";
    }

    for (var i=0, il=h2.length; i<il; i++) {
        h2[i].textContent = "Die Hard 2";
    }

    for (var i=0, il=h3.length; i<il; i++) {
        h3[i].textContent = "Die Hard 3";
    }

    for (var i=0, il=h4.length; i<il; i++) {
        h4[i].textContent = "Die Hard 4";
    }
}
```



```
// Run when DOM has loaded  
window.addEventListener("DOMContentLoaded", bruceWillis.init, false);
```

- browser.xul
- bruce-willlis.css
- bruceWilllis.js
- preferences.xul

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin/" type="text/css"?>
<prefwindow
    title="Bruce Willis Preferences"
    xmlns="http://www.mozilla.org/keymaster/gatekeeper/
there.is.only.xul">

    <prefpane label="Bruce Willis Preferences">
        <preferences>
            <preference id="brucewillis-autorun"
name="extensions.brucewillis.autorun" type="bool"/>
        </preferences>

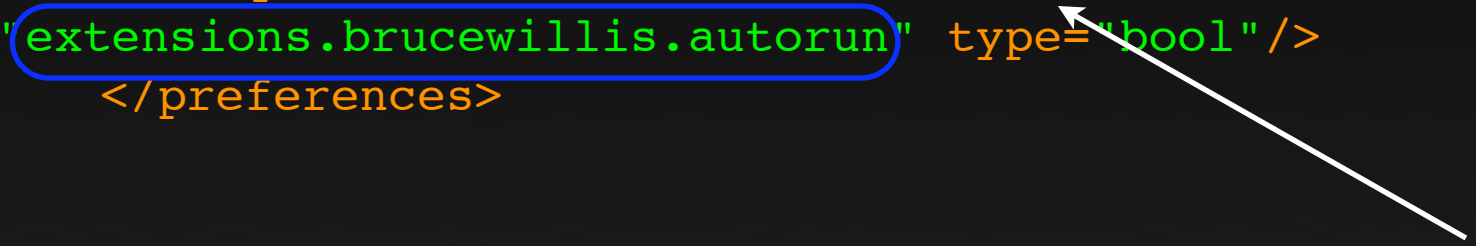
        <groupbox>
            <caption label="Settings"/>
            <grid>
                <columns>
                    <column flex="4"/>
                    <column flex="1"/>
                </columns>
                <rows>
                    <row>
                        <label control="autorun" value="Autorun"/>
                        <checkbox id="autorun"
preference="brucewillis-autorun"/>
                    </row>
                </rows>
            </grid>
        </groupbox>

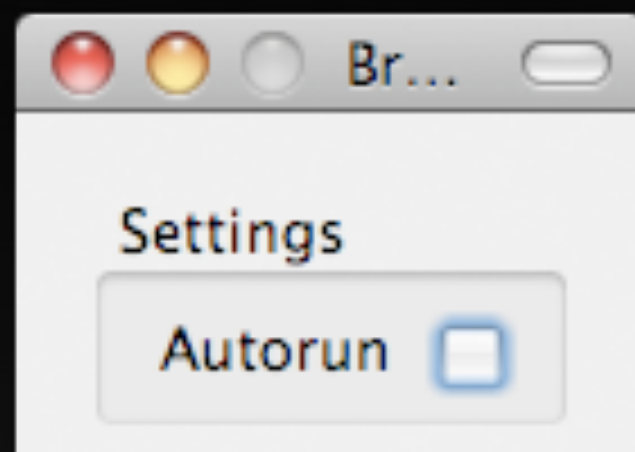
    </prefpane>

</prefwindow>
```

```
<!-- Connect element to extension preference -->
<preferences>
  <preference id="brucewillis-autorun"
name="extensions.brucewillis.autorun" type="bool"/>
</preferences>

<checkbox id="autorun" preference="brucewillis-autorun"/>
```

A blue oval highlights the string 'extensions.brucewillis.autorun' in the XML code. A white arrow points from this oval to the 'preference=' attribute in the '<checkbox id="autorun" preference="brucewillis-autorun"/>' line below.



**chrome locale folder**

Offering localization

translations.dtd:

```
<!ENTITY runbrucewillis "Run Bruce Willis">
```



**chrome skins folder**

Skin your extension with CSS and images

- `skin.css`
- `status-bar.png`
- `toolbar-button.png`

```
/* skin.css */
```

```
/* Style in View > Toolbars > Customize */
```

```
#brucewillis-toolbar-button {
```

```
    list-style-image: url(chrome://brucewillis/skin/toolbar-  
button.png);
```

```
}
```

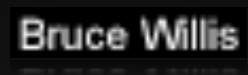
```
#brucewillis-status-bar-icon {
```

```
    width: 83px;
```

```
    margin: 0 5px;
```

```
}
```

```
/* status-bar.png */
```



```
/* toolbar-button.png */
```



**preferences - defaults folder**

Setting default extension values



```
// prefs.js
```

```
pref("extensions.brucewillis.autorun", false);
```

# Packaging & Installing

An XPI is just a zipped file

ZIP the contents of your extension folder  
(only the contents, NOT the folder itself)

## **Windows:**

Mark all files, right-click and choose:

Send To > Compressed

Rename ZIP file to .xpi

## **Mac OS X/Linux** (in Terminal):

Navigate to your extension files. Type in:

```
zip -r BruceWillis.xpi *
```

Drag your XPI file into any Firefox - Voilà!

What it looks like



## Firefox & Thunderbird

Looking for Firefox or Thunderbird? You'll find them and a whole lot more at [Mozilla.com](http://Mozilla.com).

### Mozilla Foundation

The Mozilla Foundation provides overall support for Mozilla open source software projects.

[Learn more about the Foundation...](#)

### Mozilla Store

Get [Firefox on CD](#), [t-shirts](#), and more at the new [Mozilla Store](#).



### Other Mozilla Software

[SeaMonkey 1.1.15](#)

[Camino 1.6.6](#)

[Lightning 0.9](#)

[Sunbird 0.9](#)

[Bugzilla](#)

[All projects...](#)

### Announcements

Mar 31 [Mozilla Project Celebrates Ten Years](#)

Nov 26 [Mozilla Foundation announces Directed Giving pr...](#)

Sep 17 [Mozilla Launches Internet Mail and Communicatio...](#)

### mozillaZine News

Mar 15 [Mozilla Firefox 3.1 Beta 3 Released](#)

Feb 04 [Mozilla Firefox 3.0.6 Released](#)

Dec 08 [Mozilla Firefox 3.1 Beta 2 Released](#)

### Developer News

Mar 17 [about:mozilla - Firefox, Weave, AMO, Calendar, ...](#)

Mar 16 [about:addons - Fx 3.5 Add-on Compatibility, Dev...](#)

Mar 12 [Firefox 3.1 Beta 3 now available for download](#)

# Before





### Die Hard 3

Looking for Firefox or Thunderbird? You'll find them and a whole lot more at [Mozilla.com](http://Mozilla.com).

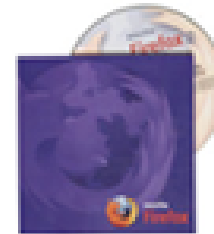
### Die Hard 2

The Mozilla Foundation provides overall support for Mozilla open source software projects.

[Learn more about the Foundation...](#)

### Die Hard 2

Get [Firefox on CD](#), [t-shirts](#), and more at the new [Mozilla Store](#).



### Die Hard 2

[SeaMonkey 1.1.15](#)

[Camino 1.6.6](#)

[Lightning 0.9](#)

[Sunbird 0.9](#)

[Bugzilla](#)

[All projects...](#)

### Die Hard 2



Mar 31 [Mozilla Project Celebrates Ten Years](#)

Nov 26 [Mozilla Foundation announces Directed Giving pr...](#)

Sep 17 [Mozilla Launches Internet Mail and Communicatio...](#)

### Die Hard 2



Mar 15 [Mozilla Firefox 3.1 Beta 3 Released](#)

Feb 04 [Mozilla Firefox 3.0.6 Released](#)

Dec 08 [Mozilla Firefox 3 .1 Beta 2 Released](#)

### Die Hard 2



Mar 17 [about:mozilla - Firefox, Weave, AMO, Calendar, ...](#)

Mar 16 [about:addons - Fx 3.5 Add-on Compatibility, Dev...](#)

Mar 12 [Firefox 3.1 Beta 3 now available for download](#)

# After

# Distributing Your Add-On

addons.mozilla.org  
or through secure location (SSL)

Successfully Getting your Addon Reviewed

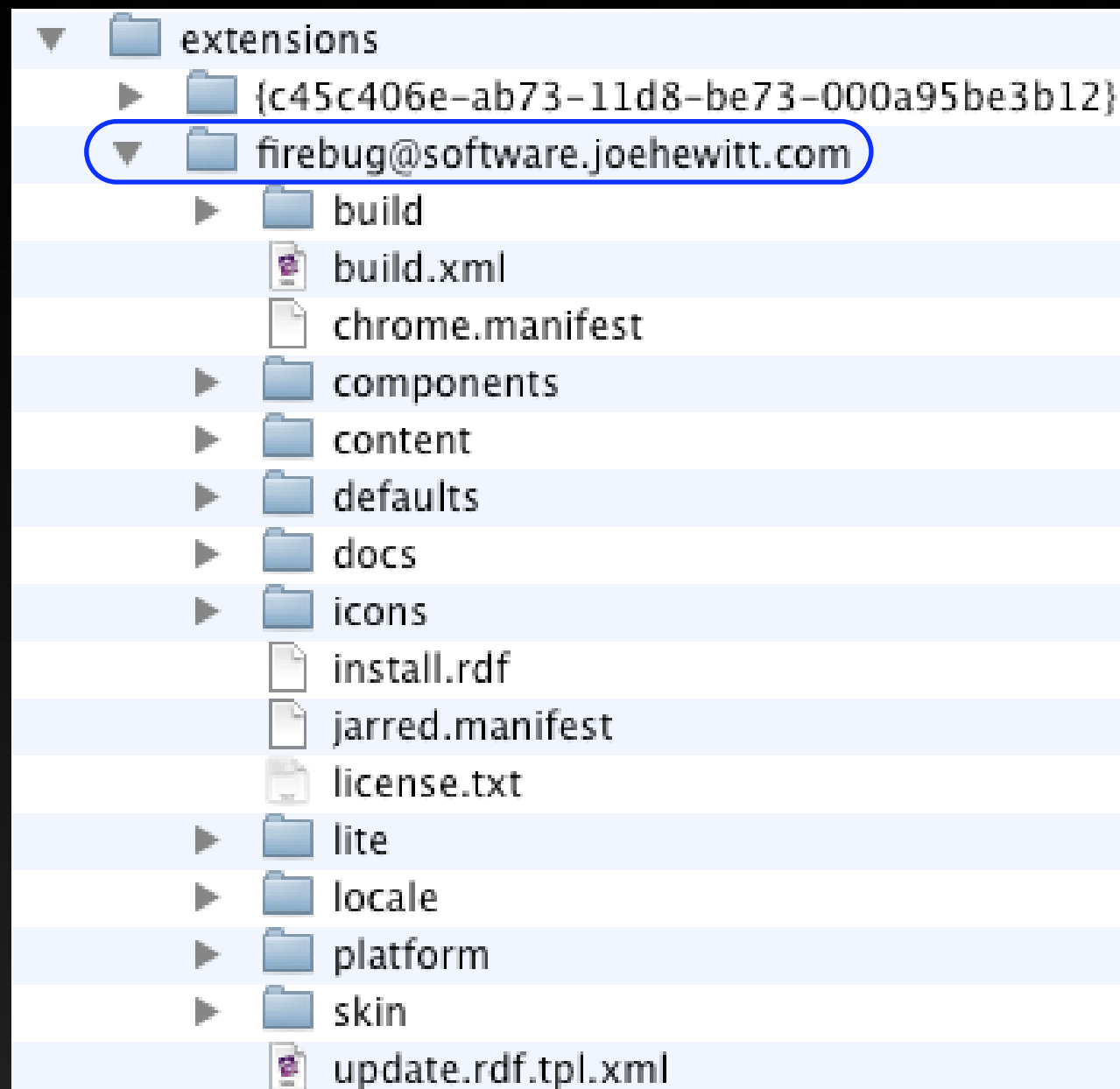
# Downloads

- Download Bruce Willis extension
- Download source files
- All code at:

[bitbucket.org/robnymann/maow-berlin/](https://bitbucket.org/robnymann/maow-berlin/)

Learn more

Learn by looking at others



In your Profile folder



What you've learned

1. Create development profile
2. Configuration settings
3. Pointing extension to your dev directory
4. Creating folder structure & files
5. Packaging & installing
6. Distributing your add-on

That's it!

# Robert Nymman

[robert@robertnymman.com](mailto:robert@robertnymman.com)