

# Taller de desarrollo de extensiones

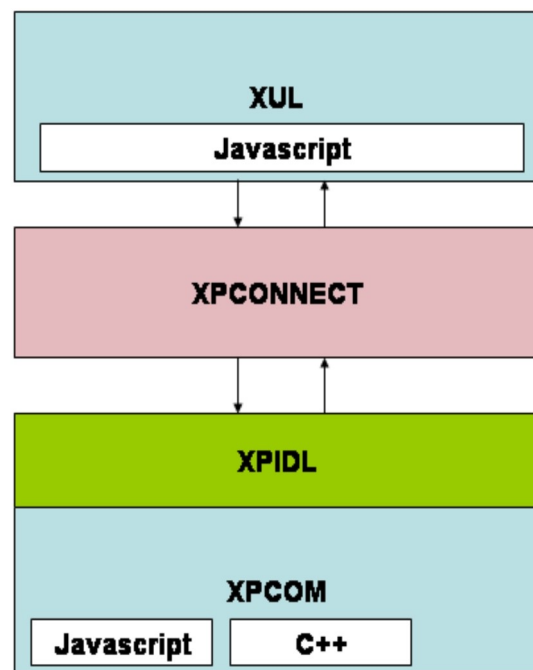
Universidad Católica de Costa Rica

26 de octubre del 2013



## Arquitectura

El front-end del Firefox es muy similar a una página web: CSS, JavaScript, y un lenguaje llamado XUL en vez de HTML. Este front-end se conoce internamente como *chrome*, y corre JavaScript privilegiado.



El JavaScript en el *chrome* tiene acceso a componentes, los cuales exponen APIs privilegiados como acceso a disco, historial, ejecución de librerías, etc. También ha tenido acceso muy sencillo al contenido de las páginas web cargadas en Firefox:

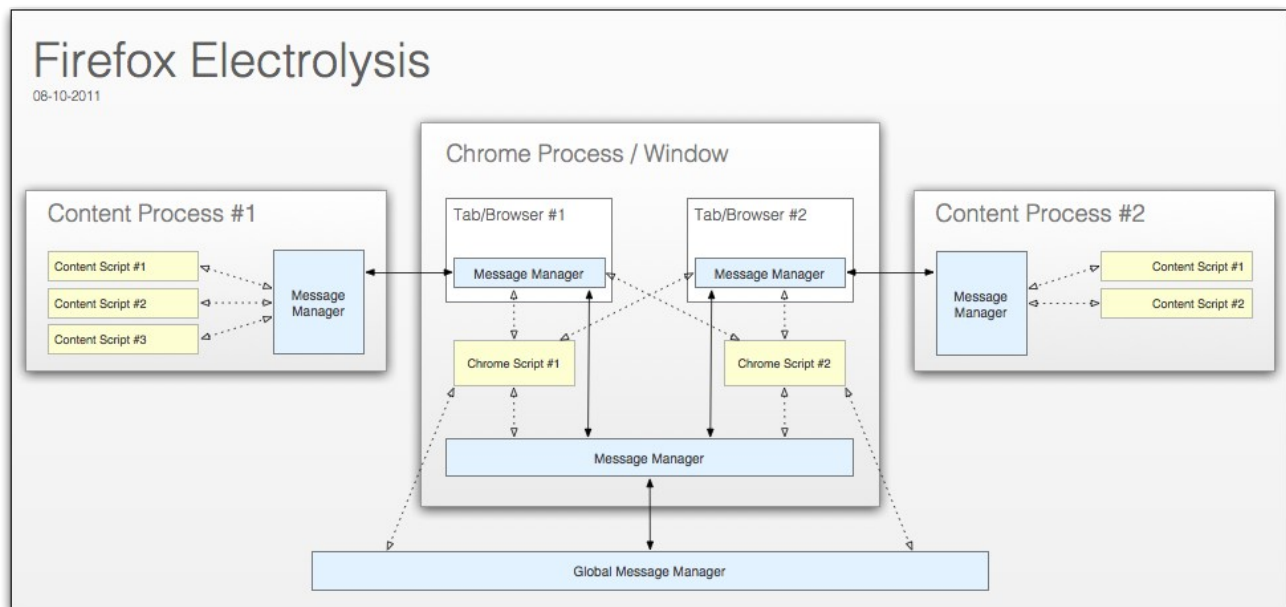
```
document.getElementById("content").contentDocument
```

La seguridad entre código privilegiado y no privilegiado se protege por medio de [wrappers](#).

## Electrolysis (Firefox multi-proceso)

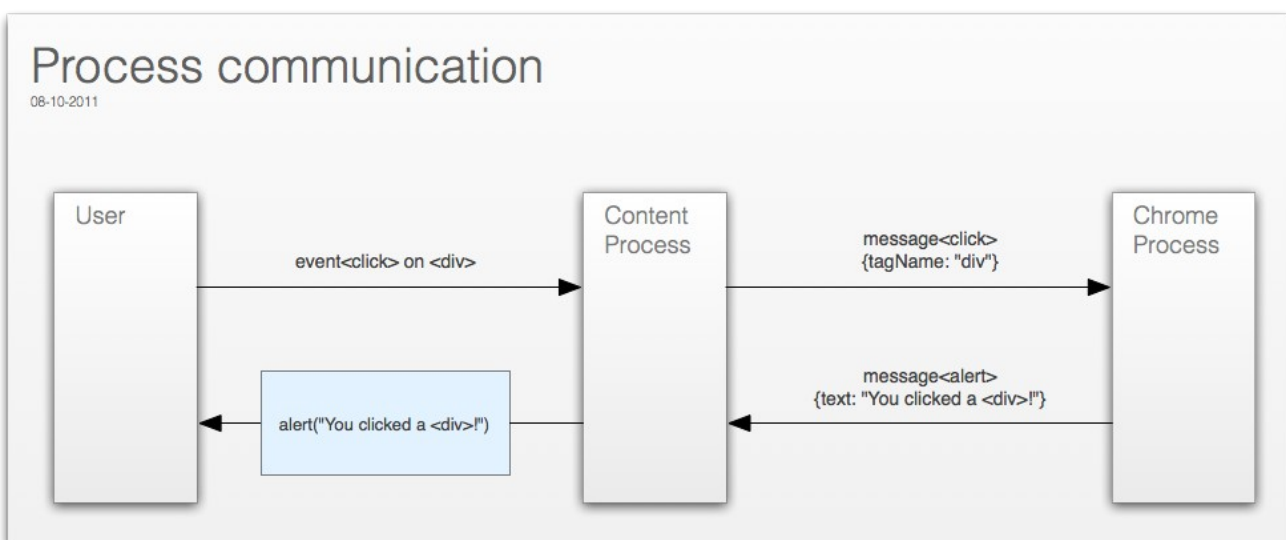
Está actualmente deshabilitado en Firefox para escritorio, pero el plan es activarlo en menos de un año. Divide el Firefox en varios procesos, con un proceso principal y procesos de

contenido.



Antes, el JavaScript del chrome podía acceder el JavaScript de cualquiera de las pestañas abiertas directamente, lo cual hacía muy fácil crear *userscripts*. Debido a la separación de procesos, ahora hay que utilizar el API del Message Manager para poder comunicarse con el contenido de las páginas.

La comunicación entre procesos es asíncrona y solo por medio de texto (strings). El Message Manager está diseñado para enviar mensajes en formato JSON.



Los ejemplos del laboratorio utilizan el esquema de mensajes para que continúen

funcionando con Electrolysis.

## *Userscripts*

Modifican las páginas web y son los complementos más sencillos y más comunes. Muchos complementos más complejos empezaron como *userscripts*.

Hay varias maneras de crearlos:

- [Greasemonkey](#). Actualmente no funciona con Electrolysis.
- [Add-ons SDK](#). Diseñado para Electrolysis, pero todavía no tiene soporte completo.
- ¡A pie! Así lo vamos a hacer en este taller.

## *Código*

- [install.rdf](#)
- [chrome.manifest](#)
- [bootstrap.js](#)
- [Message Manager](#).
- [Services.jsm](#).

## *Taller*

Hacer un *userscript* funcional basado en los ejemplos vistos en la clase. Al final veremos demos de los que deseen presentar lo que hicieron, y el mejor demo se llevará un premio.

## *Más desarrollo de extensiones*

- [Add-ons SDK](#).
- [XUL School](#).
- [Mozilla Developer Network](#).
- [addons.mozilla.org](#)

## *Mozilla Costa Rica*

- [mozilla-costarica.org](#)
- [facebook.com/MozillaCostaRica](#)
- [twitter.com/MozillaCR](#)