Hybrid Desktop/Web applications with WebKitGTK+
Xan López
What this talk is not about
Transformative effect of the Web in apps
Facebook, Twitter, Google...
It's about...
Web as a platform is winning
Why?
Reach a massive audience
Internet Users in the World by Geographic Regions - 2010

Source: Internet World Stats - www.internetworldstats.com/stats.htm
Estimated Internet users are 1,966,514,816 on June 31, 2010
Copyright © 2010, Miniwatts Marketing Group
Reasonable learning curve
If the trend continues...
The Web will be the dominant platform in the future
Not GNOME
You might say
Not very different from now
You'd be right!
We can try to do something about it
Step 1
WebKit + GNOME
WebKitGTK+
Use the platform
Native widgets
Browsers, IM clients, Email clients, Help display, ...
Problem
No easy way to access and modify web content
Step 2
DOM
JavaScript bindings
var e = document.getElementById("foo");
e.style.color = "red";
Actually...
“The DOM is designed to be used with any programming language. In order to provide a precise, language-independent specification to the DOM interfaces, we have chosen to define the specification in OMG IDL”

DOM Level 1 Specification
OMG IDL!

OMG!!1!
Object Management Group
Interface Definition Language
module core {

    interface {
        CustomToJS,
        GenerateNativeConverter,
        CustomMarkFunction,
        InlineGetPropertySlot
    } Document : Node {

        // DOM Level 1 Core
        readonly attribute DocumentType doctype;
        readonly attribute [V8Custom] DOMImplementation implementation;
        readonly attribute Element documentElement;

        [ReturnsNew] Element createElement(in [ConvertNullToNullString] DOMString tagName)
            raises (DOMException);
        DocumentFragment createDocumentFragment();
        [ReturnsNew] Text createTextNode(in DOMString data);
        [ReturnsNew] Comment createComment(in DOMString data);
        [ReturnsNew] CDATASection createCDATASection(in DOMString data)
            raises(DOMException);
        [OldStyleObjC, ReturnsNew] ProcessingInstruction createProcessingInstruction(in DOMString target,
                                                                                       in DOMString data)
            raises (DOMException);
        [ReturnsNew] Attr createAttribute(in DOMString name)
            raises (DOMException);
        [ReturnsNew] EntityReference createEntityReference(in DOMString name)
            raises(DOMException);
        NodeList getElementsByTagName(in DOMString tagname);

        // Introduced in DOM Level 2:

        [OldStyleObjC, ReturnsNew] Node importNode(in [Node importedNode,
                                                    in boolean deep)
            raises (DOMException);
        [OldStyleObjC, ReturnsNew] Element createElementNS(in [ConvertNullToNullString] DOMString namespaceURI,
                                                        in [ConvertNullToNullString] DOMString qualifiedName)
            raises (DOMException);
        [OldStyleObjC, ReturnsNew] Attr createAttributeNS(in [ConvertNullToNullString] DOMString namespaceURI,
                                                        in [ConvertNullToNullString] DOMString qualifiedName)
            raises (DOMException);

    }
Access the DOM from GObject/C (and other languages)
DEMO
Work in progress
Unstable releases already available with GObject DOM bindings (since 1.3.0)
Next stable release 1.4.0 (around March next year with GNOME 3.0)
Join us!
#webkit & #webkit-gtk at Freenode IRC
webkit.org and webkitgtk.org
That's it!