

# Mozilla projects

## Browser Components

### **Embedding**

The Embedding project is focused on allowing easy reuse of the Mozilla browser components and providing an SDK for an embeddable browser and HTML editor.

## Browser Infrastructure

### New Layout (Gecko)

The goal of the New Layout project is to create a fast, small, standards-based layout engine designed for performance and portability.

# Embedding Mozilla

## Goals:

**To provide an embeddable browser and HTML editor SDK.**

## Platform Support:

Windows (32bit), Linux , Mac 8.6+ (OSX isn't there yet)

The current api set for embedding can be found here (<http://www.mozilla.org/projects/embedding/PublicAPIs.html>). Note that nsIWebshell is deprecated and will eventually be completely removed from the tree. If you're using the M15 milestone for embedding, you're using code that was partially converted to use the new apis and that code is incomplete. That's not to say that the current embedding api is complete, but it does reflect the direction we're taking things.

# Embedding Mozilla

## Topics of Interest:

- **General API and usage documentation**

  - Historical Docs:

    - API documentation

    - WebShell Information (depricated)

- General UI overriding and usage documentation.
- Listeners/progress/callback handling.
- Roll your own browser
- Roll your own editor/composer
- How to run the smoketests
- For Hire - Looking for someone who knows mozilla
- Changing browser settings using nsIPref
- **Embedding examples**
- Embedding QA
  - smoketests
- The tracking bug for API related changes is 70229.
- The API review meeting notes are here.

# Embedding Mozilla

## Contributors:

These contributors are closest to the day-to-day embedding activity. There are certainly many other contributors.

- Chris Blizzard
- Adam Lock
- Rick Potts
- Doug Turner
- Judson Valeski
- Conrad Carlen
- Radha Kulkarni
- Chak Nanga
- Dan Matejka

# Embedding Examples

There are many examples of products already using Gecko for embedding. Here's a partial list (email me with others that I missed).

## Linux:

- Galeon browser
- Gateway/AOL/Netscape's Instant AOL device
- gtk widget (sample widget on mozilla)
- gtkEmbed (sample app on mozilla)
- SashXB (part of GNOME)
- Skipstone browser (uses thegtk widget)
- Eazel/Nautilus (part of GNOME )
- ActiveState's Komodo
- XParts (uses gtk widget)
- Nokia multimedia device

## Mac:

- Powerplant sample (sample app on mozilla)**
- wKiosk (Kiosk type browser)**

# Embedding Examples

## Windows:

- k-meleon browser (download it from zdnet; their real site (<http://k-meleon.org>) is flakey).
- winEmbed (sample app on mozilla)
- mfcEmbed(sample app on mozilla)
- ActiveX control (sample control on mozilla)
- Neoplanet (a Gecko "add-on" for their browser)
- Allaire's HomeSite (integrated Gecko browser)
- Bradsoft CSS builder (uses mozilla ActiveX embedding control)
- ActiveState's Komodo
- RoboForm (uses mozilla ActiveX embedding control)
- NetCaptor (uses mozilla ActiveX embedding control)

## Java:

- Webclient (java browsing object(s) on mozilla)

# Links

- Mozilla Embedding プロジェクト

<http://www.mozilla.org/projects/embedding/>

<http://www.mozilla.org/projects/embedding/PublicAPIs.html>

<http://www.mozilla.org/projects/embedding/examples.html>

# X2Web Macintosh版

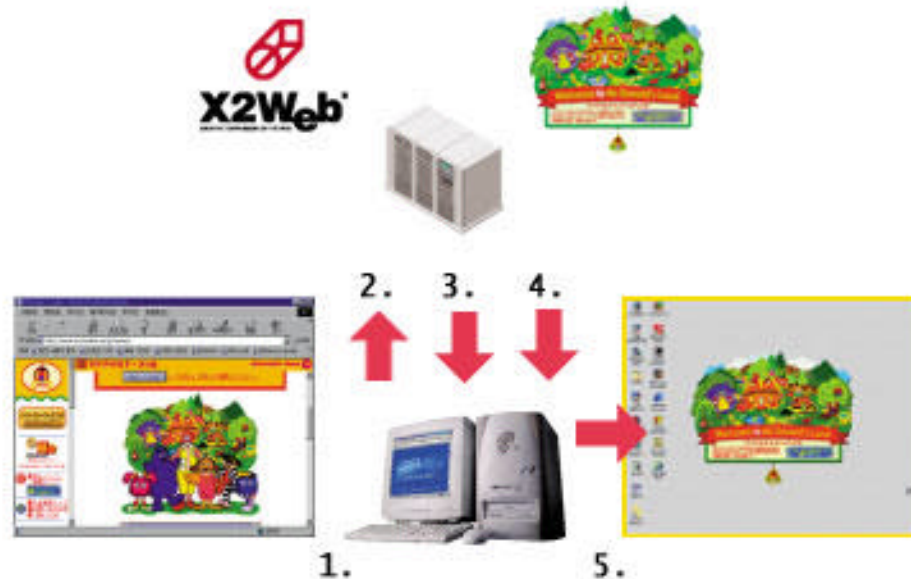


## X2Web Generator ( Windows/Macintosh )

X2Webコンテンツの作成からWeb上への公開まで、X2Webに関する制作環境をシームレスに提供する、専用のデザイン&開発ツールです。特別なプログラムの知識を必要としない操作性とMacintosh/Windowsに対応したマルチプラットフォーム仕様で、幅広いクリエイターの方にご使用いただけます。

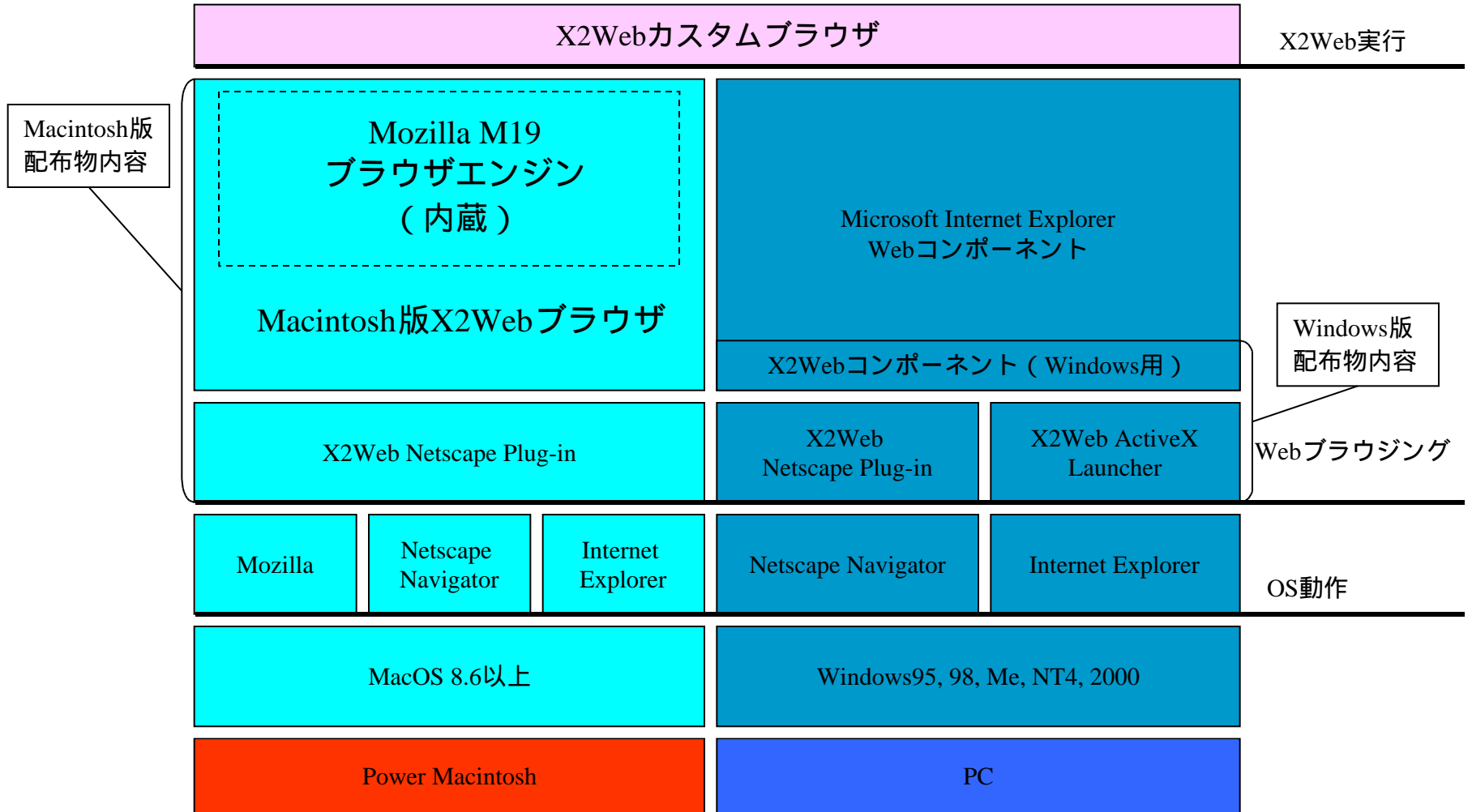
# X2Web Macintosh版

## 【動作概念】

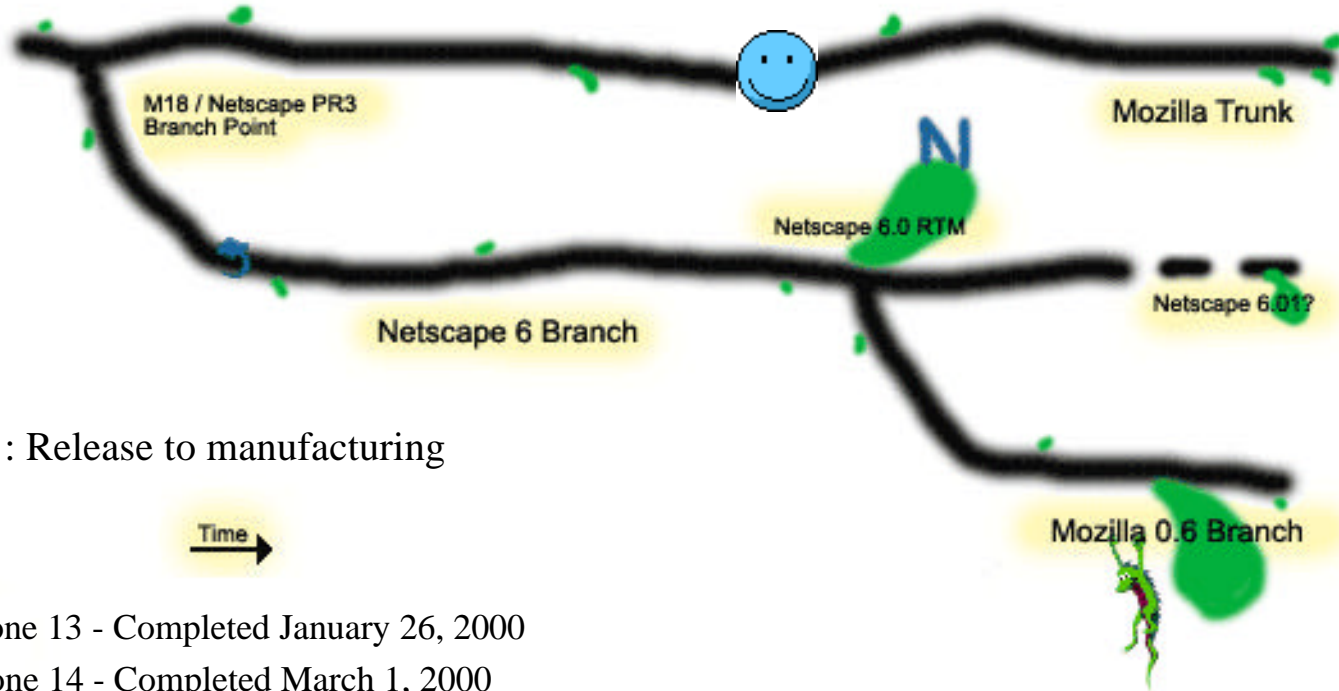


1. 普通のブラウザでX2Webサイトへアクセス
2. X2WebサイトをPCが確認
3. サーバからX2Webコンポーネントをブラウザが自動的にゲット  
( Macintoshは事前にプラグインのインストールが必要 )
4. サーバよりX2Webブラウザデータが配信される
5. X2Webブラウザがデスクトップ上で実行

# X2Webのアーキテクチャ



# 当時のロードマップ



RTM : Release to manufacturing

Milestone 13 - Completed January 26, 2000

Milestone 14 - Completed March 1, 2000

Milestone 15 - Completed April 18, 2000

Milestone 16 - Completed June 13, 2000

Milestone 17 - Completed August 7, 2000

Milestone 18 - Completed October 12, 2000

Mozilla 0.6 - Completed December 6, 2000

# エンベッディング

- モジラをエンベッディングしよう！
- エンベッディングとは
- モジラはそれ自体、単なるウェブブラウザというよりは、ウェブコミュニケーションの為の統合環境です。ウェブブラウザ、HTMLエディタ、メールニュースクライアントなど、種々のパーツを含んでいます。
- また、それらの外観であるユーザーインターフェースパーツを、XULスクリプトでコントロールする為のさまざまな仕掛けをも含みます。（XMLパーサー、XMLアプリケーション、など）

# エンベッディング

- つまり、モジラ全体を捕らえると、それは、ウェブブラウザライブラリの外側に（実際にはウェブブラウザ部分は、単一のライブラリになってはいたませんが）、モジラアプリケーションの固有の仕様を実現する為のアプリケーション部分という鎧をまとっているようなものです。
- エンベッディングとは、モジラの中から、このウェブブラウザ部分のみを取り出して、これを独自の仕様のアプリケーション外枠の中に埋め込んで使用するものです。

# エンベッディング

- モジラ全体をその仕様のあるがままに受け入れて使用するのではなく、独自に作成したアプリケーションフレームワーク、または、既存の何だかのアプリケーションフレームワークから起こしたアプリケーションに、パーツとしてモジラブラウザを埋め込みたい場合は、このエンベッディングという手法を採用することになります。

## 既存のフレームワークによるエンベディング

- Macintoshのモジラには、"PPBrowser" という、CodeWarriorのPowerPlantをフレームワークとして使用して、これにモジラブラウザを埋め込んだ例があります。
- `mozilla/embedding/browser/powerplant/PPBrowser.mcp`
- モジラ本体をビルド後、上記プロジェクトによりビルドできます。
- Conrad Carlen氏によりチェックインされたもので、同氏と他の方々により、管理進行されています。

## 既存のフレームワークによるエンベディング

- モジラのMacintoshでのエンベディングを行う場合、間違いなくこのソースが、唯一無二の貴重なお手本となるでしょう。

## 独自のフレームワークによるエンベディング

- "PPBrowser" は、PowerPlantフレームワークにモジラブラウザをエンベディングした例ですが、当ページでは、"PPBrowser" をバイブルとして、独自のアプリケーションへのモジラブラウザのエンベディングを試みます。
- "marbrow" が、この例として作成したものです。

