

# Mozilla und das Mozilla-Projekt



Vortrag für das IFC-Seminar

von  
Elmar Ludwig  
Universität Osnabrück

Wintersemester 2001/02



# Inhaltsverzeichnis

<b>Überblick</b>	<b>2</b>
Mozilla . . . . .	3
Gecko . . . . .	4
Bugzilla . . . . .	5
Tinderbox . . . . .	6
Bonsai . . . . .	7
<b>Architektur</b>	<b>8</b>
XPCOM . . . . .	8
XUL . . . . .	9
<b>Literaturverzeichnis</b>	<b>10</b>

# Überblick

Im April 1998 veröffentlichte Netscape den (fast) kompletten Quellcode ihres gleichnamigen Web-Browsers unter einer „Open-Source“ Lizenz und übergab die Entwicklung in die Hände der „Mozilla Organisation“, die anfangs zwar praktisch nur aus Programmierern von Netscape bestand, die aber auch nach außen hin offen für andere Entwickler sein sollte.

Ich möchte nun im ersten Teil meines Vortrags vorstellen, was sich in der Zwischenzeit daraus entwickelt hat. Der Schwerpunkt soll dabei auf den verschiedenen dort entstandenen Projekten liegen, die speziell für Softwareentwicklung auch von allgemeinem Interesse sind:

- Mozilla - the current Mozilla browser
- Gecko - an embeddable HTML layout engine
- Bugzilla - web-based bug tracking database
- Tinderbox - compile/build monitoring system
- Bonsai - CVS (version control) tracking tool

Im zweiten Teil werde ich dann näher auf einige wichtige Komponenten der Architektur eingehen (vor allem XPCOM und XUL) und ein paar einfache Beispiele aus diesem Bereich zeigen (wenn die Zeit reicht).

Worum es **nicht** in diesem Vortrag geht:

- Wie bediene ich einen Web-Browser?
- Wie installiere ich das Mozilla? (etc.)
- Aber Programm ... ist doch viel besser!

## Mozilla

Man muß zunächst unterscheiden zwischen **Mozilla.org** und **Mozilla**:

- Mozilla.org ist die halb-offizielle Bezeichnung für die „Mozilla Organisation“, die die Projektverwaltung übernommen hat. Teile davon sind immer noch bei Netscape angesiedelt.

Mozilla.org betreibt Dinge wie den Web-Server, das CVS-Repository (Versionskontrolle für Quellcode), Bugzilla, entwirft die Zeitrahmen für Software-Releases, veranstaltet Konferenzen etc.

- Mozilla ist der Name für das „wichtigste“ Software-Produkt, die Applikations-Plattform, auf der dann Programme wie der Mozilla-Browser (genauer: die Navigator-Applikation) laufen.

Mozilla ist im Prinzip eine „Internet-Application Platform“, deren einzige Aufgabe darin besteht, anderen Applikationen eine Ausführungsumgebung zur Verfügung zu stellen.

Man kann das im Prinzip in etwa mit dem *Applet-Viewer* für Java vergleichen (von der Architektur her).

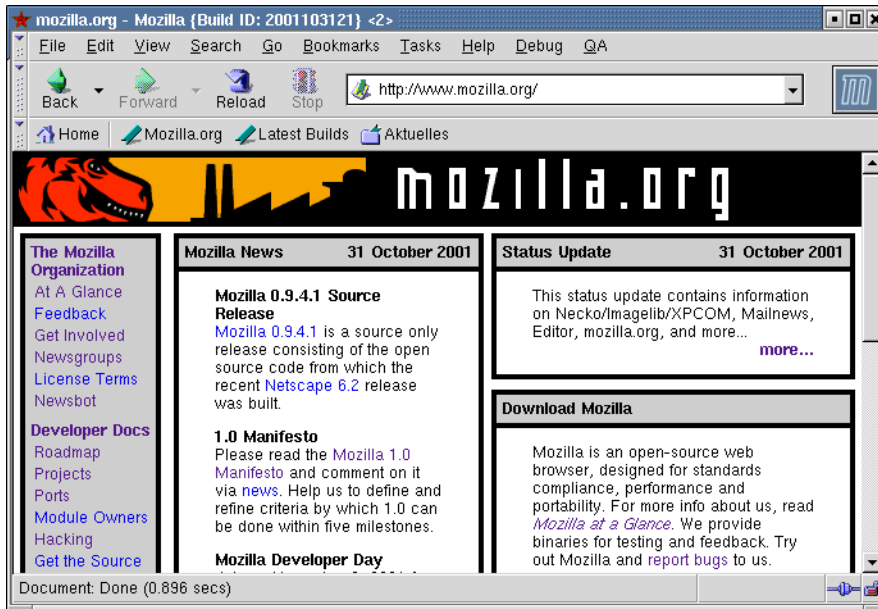
Auf dem Mozilla-Plattform laufen dann Applikationen wie z.B.:

- Navigator (`chrome://navigator/content/navigator.xul`)
- Editor (`chrome://editor/content/editor.xul`)
- Mail/News (`chrome://messenger/content/messenger.xul`)
- IRC-Client (`chrome://chatzilla/content/chatzilla.xul`)
- JavaScript-Debugger (`chrome://venkman/content/venkman.xul`)
- ...

Man kann weitere Applikationen zum Teil direkt aus dem Internet laden und ausführen oder auch lokal installieren.

Beispiel zeigen...

## Gecko



*„And the beast shall come forth surrounded by a roiling cloud of vengeance. The house of the unbelievers shall be razed and they shall be scorched to the earth. Their tags shall blink until the end of days.“*

from The Book of Mozilla, 12:10

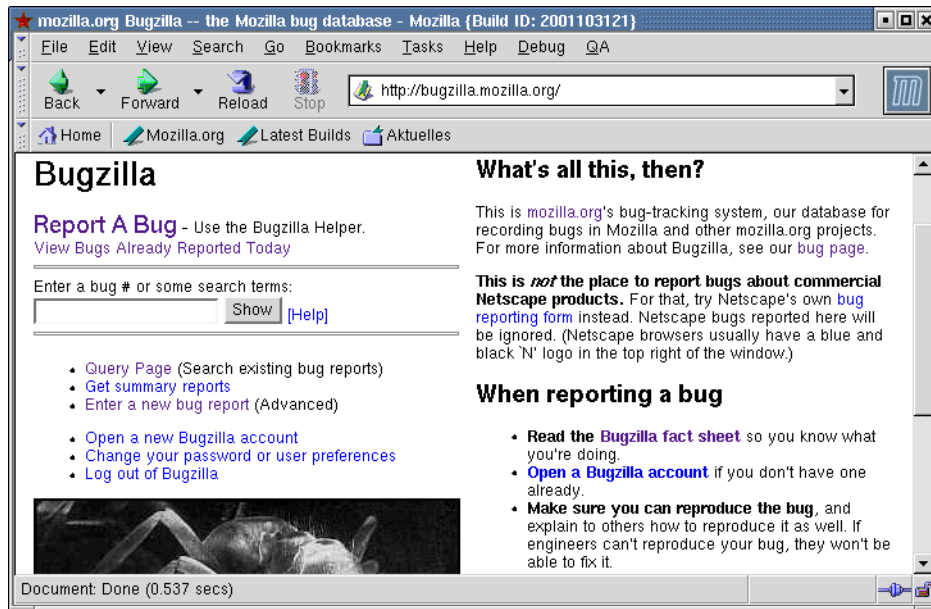
**Gecko** ist die HTML-Layout Komponente in Mozilla, und ist (als echte Komponente eben) so eingerichtet, daß sie auch in anderen Programmen (Beispiele unter Linux dafür sind GtkEmbed oder Galeon) für eigene Zwecke wiederverwendet werden kann.

Der Internet-Explorer erlaubt (natürlich nur unter Windows) so etwas ähnliches.

Gecko soll die „üblichen“ Standards des W3C unterstützen, vor allem:

- HTML 4.0
- CSS 1 und CSS 2 (bisher nur teilweise)
- XML (und XSLT)
- PNG bzw. MNG (für Animationen)
- SVG (noch in Entwicklung)

# Bugzilla



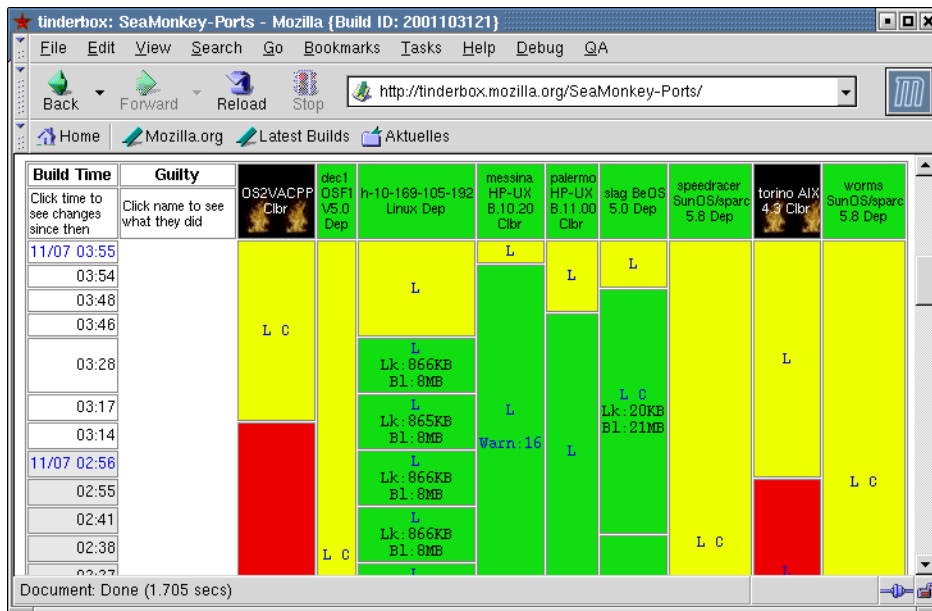
**Bugzilla** ist ein Web-basiertes Werkzeug zur Verwaltung von Fehlerberichten oder Problemfällen. Jeder einzelne Bericht kann Attachments (wie zum Beispiel Testfälle, Screenshots oder Patches) enthalten und hat eine eigene CC-Email-Liste für Personen, die bei Änderungen automatisch informiert werden.

Das kann heißen, daß sich der Status eines Bugs ändert, ein Patch oder Testfall hinzukommt, oder einfach nur jemand einen Kommentar abgibt.

Speziell diese Komponente (die an sich überhaupt nichts mit Mozilla zu tun hat) kann für andere Software-Projekt ein sehr hilfreiches Werkzeug sein.

Bugzilla ist z.B. auch als Bug-Tracker für GNOME und XMMS im Einsatz.

# Tinderbox



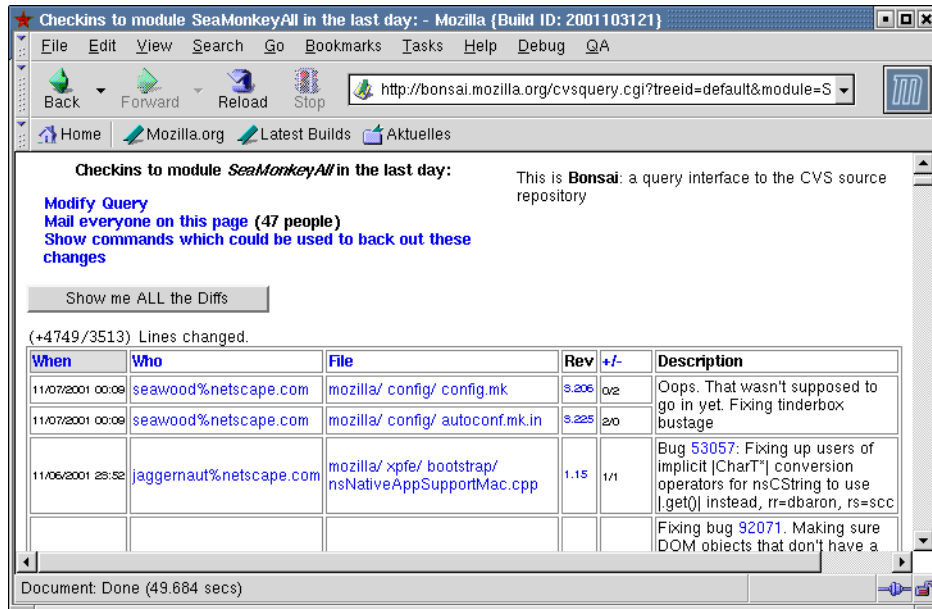
**Tinderbox** ist ein System zur (halb-)automatischen Überwachung von Übersetzungsvorgängen (compile/build). Die Tinderbox-Klienten machen nichts anderes als permanent die aktuelle Version eines Programms (aus dem CVS-System) zu übersetzen und eventuell vorhandene automatische Testfälle anschließend abzuarbeiten.

Der Tinderbox-Server sammelt diese Informationen ein und stellt den aktuellen (oder auch vergangenen) Status auf einer Web-Seite dar, von der aus auch die Build-Ausgaben und Fehlermeldungen angesehen werden können.

Die Klienten können natürlich weltweit verteilt sein und verschiedene Betriebssysteme (bzw. Versionen) verwenden.



# Bonsai



**Bonsai** ist „nur“ ein Web-Frontend für das CVS-System (*Concurrent Version Control System*), das inzwischen bei sehr vielen großen Projekten zur Verwaltung des Programmtextes eingesetzt wird. Es gibt am 19.12.2001 hier im IFC-Seminar einen Vortrag über Versionskontrolle mit CVS von Sascha Teichmann.

Mit Bonsai kann man vor allen leicht ansehen, was sich gerade im Quellcode „so tut“ (vor allem spannend, wenn es Fehler bei der automatisierten Übersetzung gibt) bzw. bei Bedarf auch Änderungen schnell rückgängig machen.

Außerdem kann man spezifische Versionen einzelner Dateien aus dem Haupt-Zweig oder aus anderen *Branches* ansehen.

# Architektur

Eigentlich müsste man hier wesentlich tiefer einsteigen, wenn man die Details der Architektur verstehen möchte, wofür hier aber die Zeit fehlt. Daher werde ich jetzt nur einen kurzen Überblick geben und statt dessen einige Beispiele zeigen (bzw. vorführen).

## XPCOM

**XPCOM** ist die Abkürzung für *Cross-Platform Component Object Model*.

Wer auf der Windows-Plattform schon mal mit COM (dem *Component Object Model* von Microsoft) zu tun hatte, wird das alles sehr vertraut finden.

XPCOM ist COM, allerdings in einer neuen – und portablen – Implementierung.

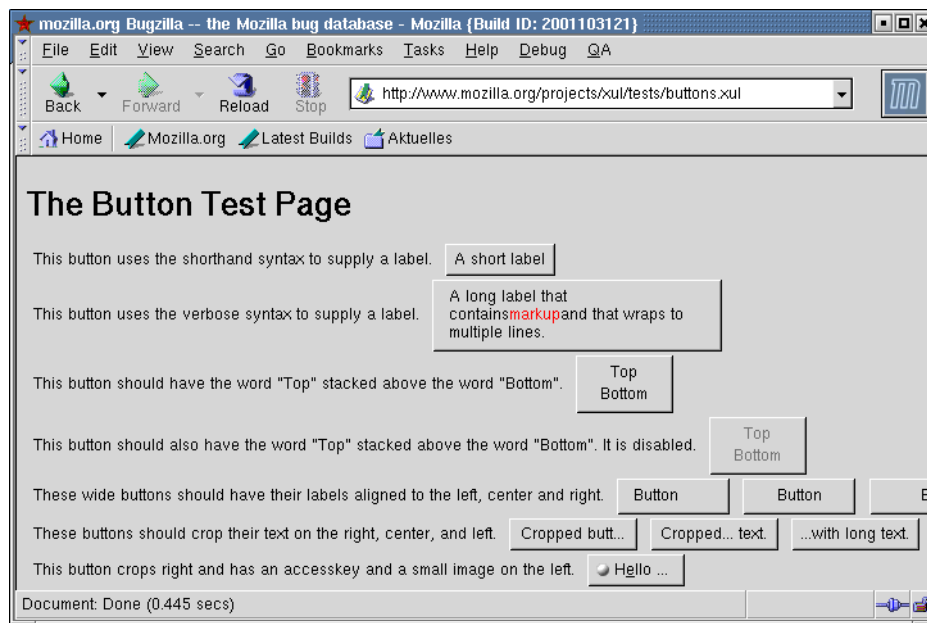
Worum geht es bei XPCOM (bzw. COM):

- eigenständige, wiederverwendbare *Komponenten* (Objekte)
- Idee von *Nachrichten* statt Funktionsaufrufen
- Entkoppelung von Schnittstelle und konkreter Implementierung (*Interface-based Programming*)
- sprach-unabhängige Definition des Interfaces einer Komponente ( $\Rightarrow$  IDL)

Sich bei der Implementation der Komponenten-Idee sehr eng an dem COM aus der Windows-Welt zu orientieren, halte ich persönlich eher für einen Schritt zurück, auch wenn dafür die „Lernkurve“ für Entwickler, die bereits mit COM gearbeitet haben, sehr flach bleibt. . .

„*Those who fail to learn from <insert term here> are doomed to reinvent it poorly.*“

# XUL



**XUL** ist die *XML User Interface Language*, in der – zusammen mit einer erweiterten Version von *JavaScript*, (mittlerweile als Version 1.5 von der ECMA standardisiert) – ein großer Teil der Benutzeroberfläche implementiert ist. XPCOM Komponenten können von JavaScript aus angesprochen werden.

Damit erreicht man vor allem:

- Plattform-unabhängige Realisierung der Benutzeroberfläche
- Die Oberfläche ist bei Bedarf komplett (und relativ leicht) anpaßbar bzw. sogar komplett austauschbar
- Komponenten können - potentiell - direkt über das Internet geladen werden, da kein Programmcode maschinenspezifisch übersetzt werden muß

Leider muß man (mindestens im Moment noch) gewisse Abstriche bei der Geschwindigkeit hinnehmen...

# Literaturverzeichnis

- [Mozilla.org] Homepage des Mozilla-Projekts <<http://www.mozilla.org/>>.
- [MozillaZine] Nachrichten und Diskussionen rund um das Mozilla-Projekt  
<<http://www.mozillazine.org/>>.
- [MozDev] Diverse externe Projekte, die auf Mozilla aufbauen  
<<http://www.mozdev.org/>>.