

- **Fix the web with Greasemonkey**
- SS08 (23.04.2008)
  - » Aaron Müller (SEB6)

# Greasemonkey

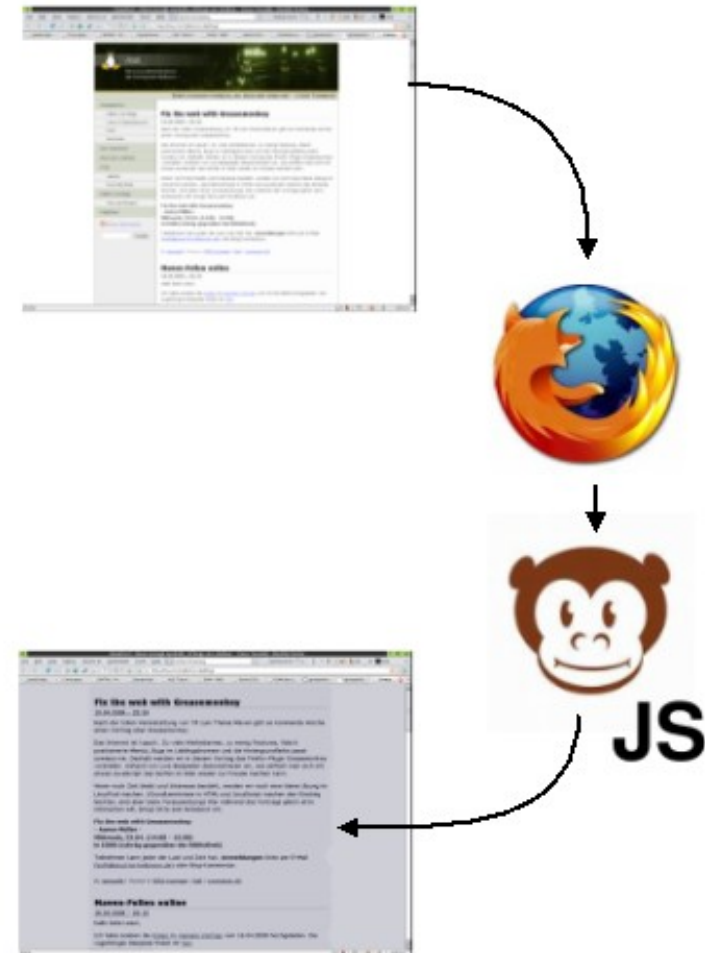
*Because it's your web.*

- Active Browsing
- Greasemonkey
  - » Einsatzgebiete, Beispiele
  - » Scripts installieren und verwalten
- Technik
  - » Firefox-Extension
  - » DOM
  - » JS-Schnittstelle
- Eigene Scripts
  - » Tools
  - » Aufbau eines Scripts
  - » Common Tasks
  - » Live-Scripting
- Alternativen
  - » Opera, Turnabout
  - » Privoxy
- Ressourcen

# Active Browsing

*Das Konzept hinter Greasemonkey.*

- Normale Anfrage:
  - » konsumieren
- Active Browsing:
  - » **manipulation** der Seite durch Browser oder Proxy
  - » **verwenden** statt konsumieren



# Greasemonkey

*So siehts aus.*

## ■ Firefox-Plugin

- » Verwalten von „UserScripts“
- » Namespaces

## ■ JavaScript

- » Scriptsprache für Manipulation

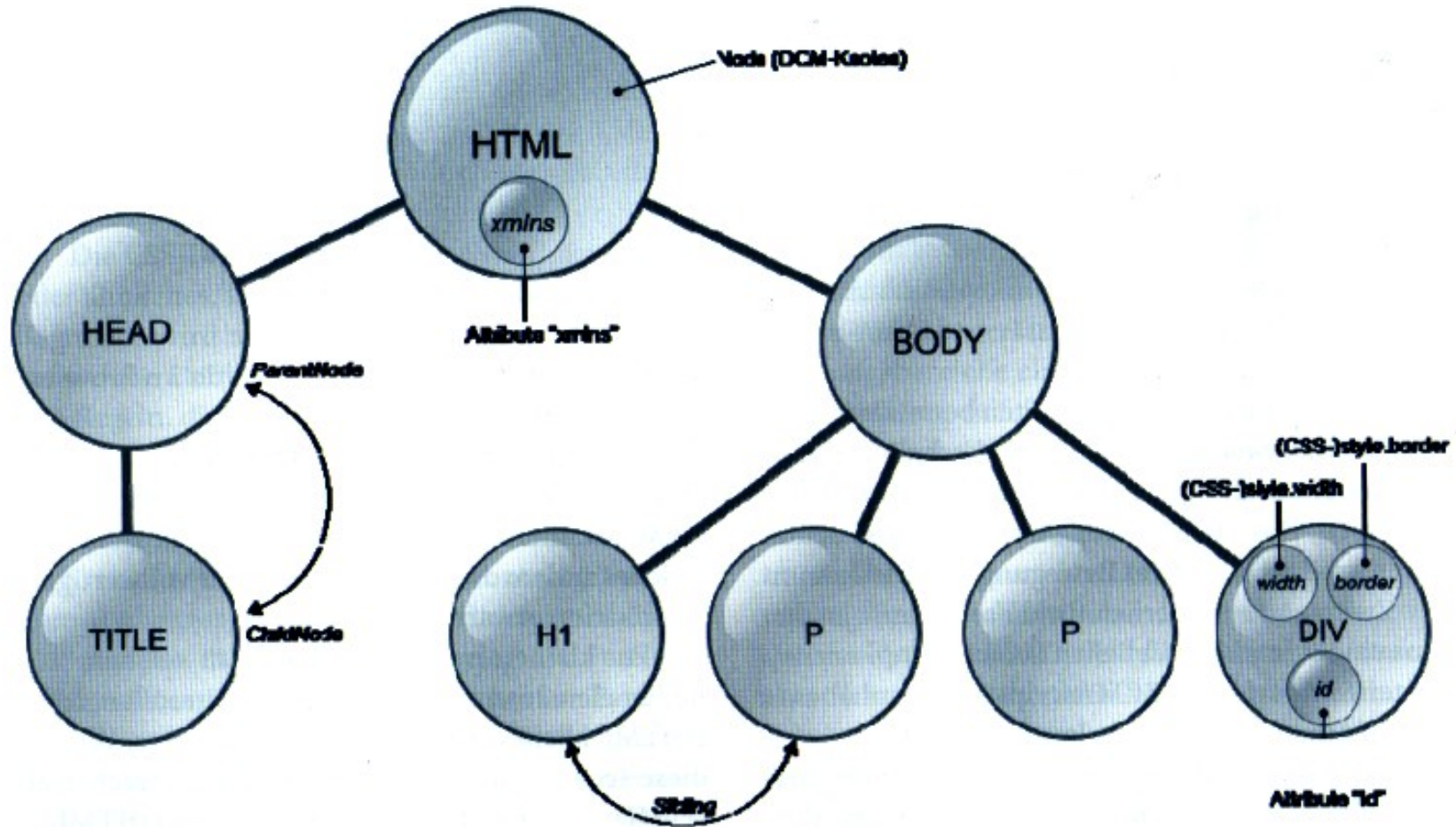
## ■ Beispiele

- » YouTube Video vergrößern
- » Werbung entfernen
- » Seite lesbar machen
- » Informationen zusammentragen
- » Shortcuts
- » Formulare ausfüllen
- » Mashups, Features
- » ...

- Zutaten
  - » Firefox
  - » Greasemonkey Extension
- DOM
  - » Knoten
  - » Attribute
- JS-API
  - » Manipulation
  - » Events
- CSS

# Document Object Model

... auch DOM-Baum genannt



## ■ Node-Methoden

- » `.(get|set|remove) Attribute()`
- » `.(append|remove) Child()`

## ■ Node-Attribute

- » `.innerHTML`
- » `.style`
- » `.parentNode`
- » `.firstChild, .lastChild`

## ■ Globale Objekte

- » `window`
- » `document`
  - `.createElement()`
  - `.createTextNode()`
  - `.getElementsByTagName()`
  - `.getElementById()`

## ■ Events

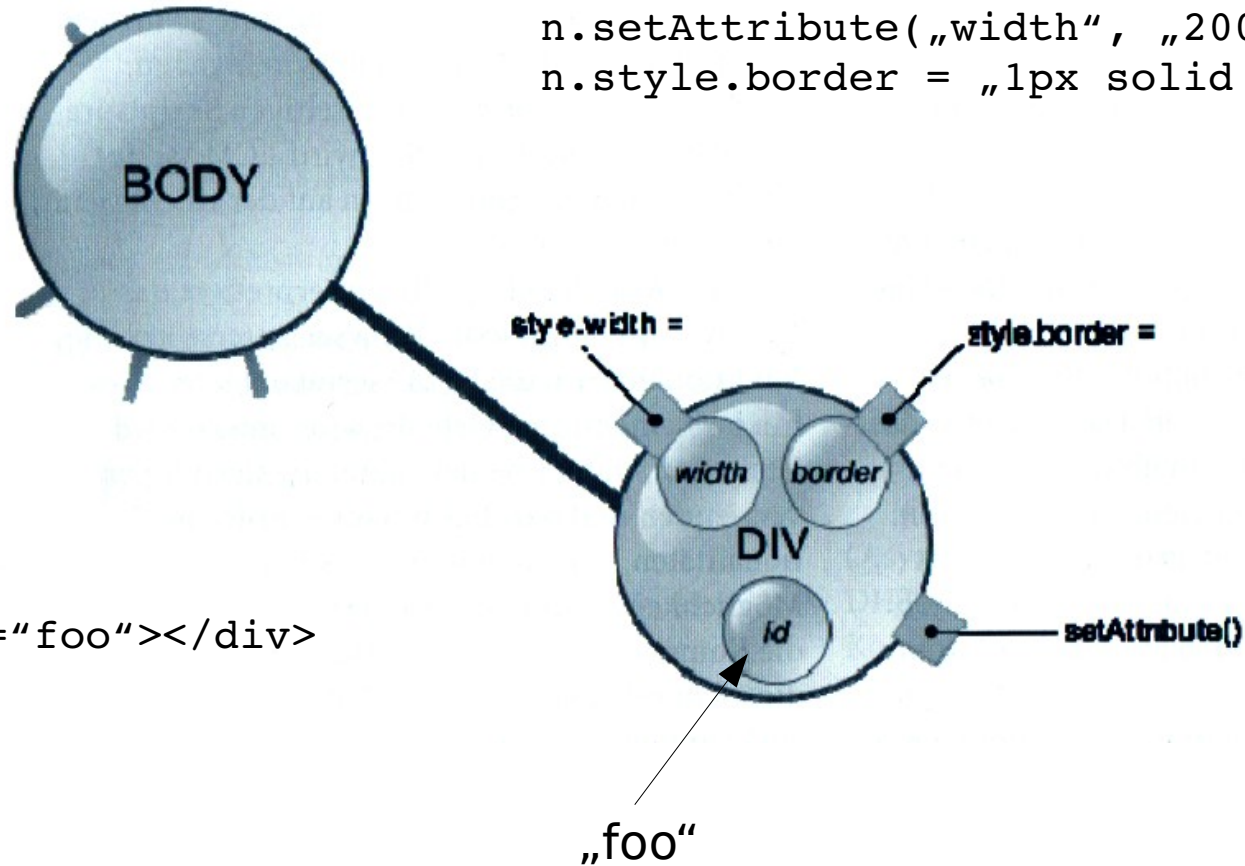
- » `onClick(), onKeyUp(), ...`

# DOM-Manipulation

*Den Baum bearbeiten.*

```
var n = document.getElementById("foo");

n.setAttribute("width", "200px");
n.style.border = "1px solid black";
```



```
...
<body>
  <div id="foo"></div>
...
```

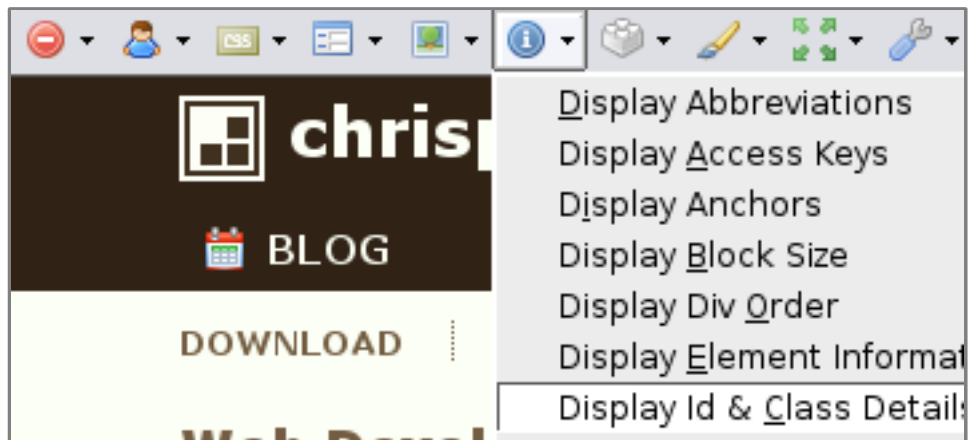


# Eigene Scripts schreiben

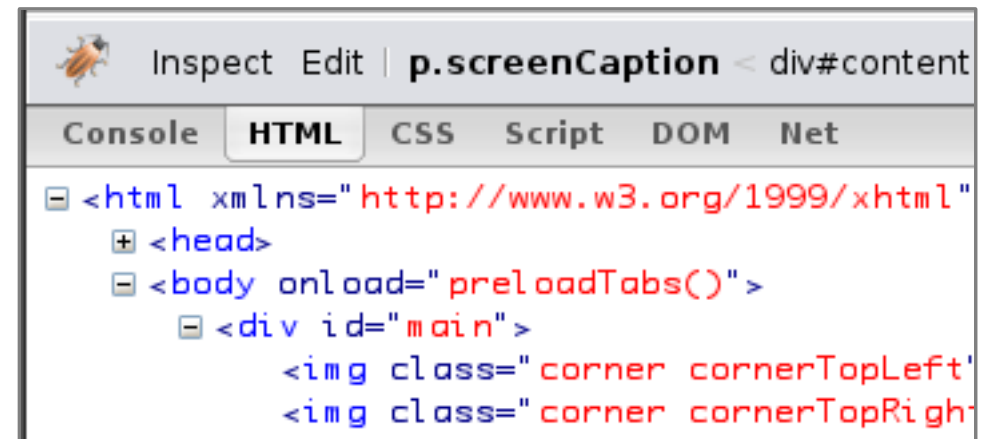
Das Handwerkszeug

9

- WebDeveloper  
Toolbar
  - » von Chris Pederick



- FireBug
  - » von Joe Hewitt



# Aufbau

*Ein simples UserScript*

10

```
// ==UserScript==
// @name           SitePoint
// @namespace      http://aaron-mueller.de/
// @description    Removes annoying Banners and extend the
//                 content.
// @include        http://www.sitepoint.com/*
// @include        http://sitepoint.com/*
// @exclude        http://sitepoint.com/forum/*
// ==/UserScript==

var ads = document.getElementById("articlebanner")
ads.style.display = "none";

document.getElementById("maincontent").style.width = "100%";
document.getElementById("content").style.width = "100%";
```

# „Patterns“ I

*Elemente selektieren*

11

- Für alle H3, ...

```
var e = document.getElementsByTagName(„h3“)  
for (var i=0; i<e.length; i++) {  
    e[i].style.backgroundColor = „green“  
}
```

- Für das Element mit der ID „ads“, ...

```
var e = document.getElementById(„ads“)  
e.style.backgroundColor = „green“
```

# „Patterns“ II

*Elemente verstecken oder entfernen*

12

- Element aus dem DOM-Baum löschen

```
var e = document.getElementById(„ads“)  
e.parentNode.removeChild(e)
```

- Element ein/ausblenden

```
var e = document.getElementById(„ads“)  
e.style.display = „none“
```

```
// wieder einblenden  
e.style.display = „“
```

## ■ Einen neuen Link einfügen

```
var nav = document.getElementById(„navBar“)  
if (nav) {  
    var link = document.createElement(„a“)  
    link.innerHTML = „Zur Hauptseite“  
    link.setAttribute(„href“, „/“)  
  
    nav.parentNode.insertBefore(link, nav)  
}
```

# „Patterns“ IV

*Elemente verändern/verschieben*

14

## ■ CSS-Stil ändern/erweitern

```
e.style.left = „0px“ // left: 0px  
e.style.backgroundColor = „green“ // background-color: green
```

## ■ Attribute hinzufügen/ändern/löschen

```
e.setAttribute(„href“, „http://google.com/“)  
e.removeAttribute(„href“)
```

## ■ Maus

- » Element angeklickt?
- » Maus über Element?

```
e.addEventListener(  
  „click“,  
  function() {  
    /* ... */  
  },  
  false  
)
```

## ■ Tastatur

- » Taste gedrückt?

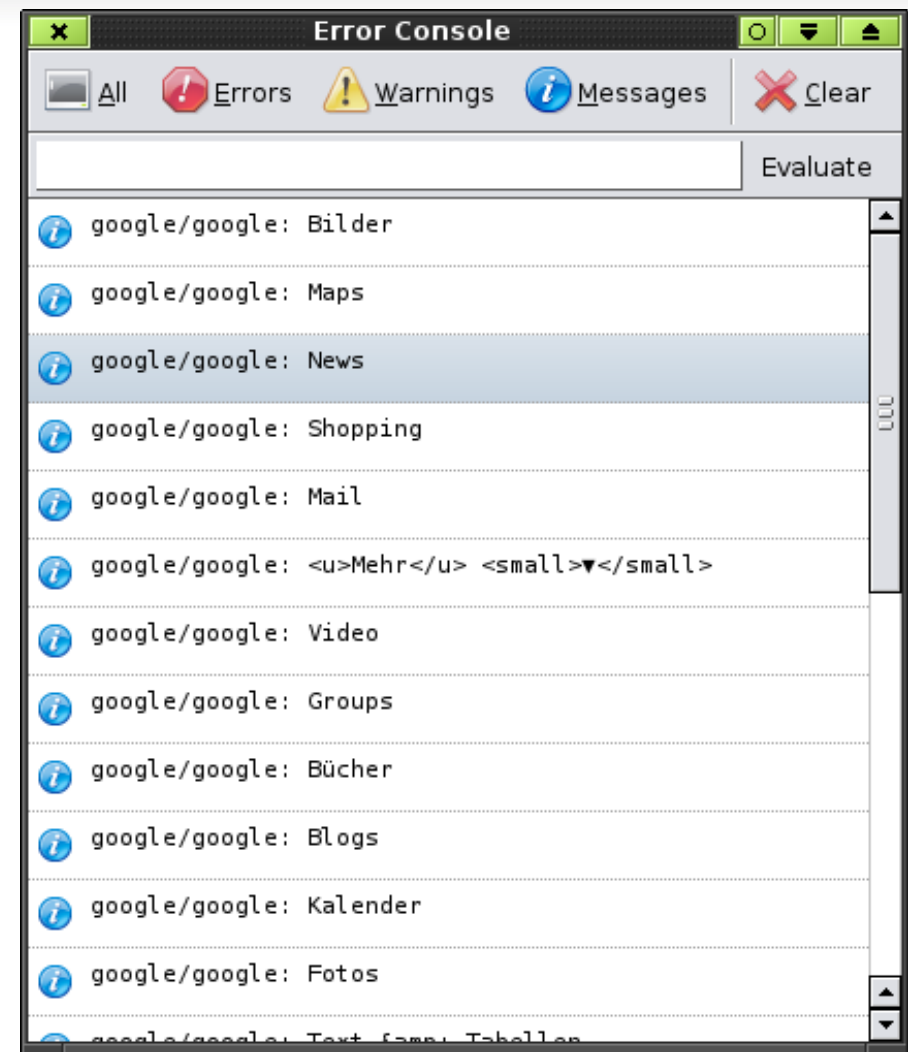
```
e.addEventListener(  
  „keydown“,  
  function(evt) {  
    if (evt.which == 72) {  
      /* g wurde gedrückt */  
    }  
  },  
  false  
)
```

```
< href=“/home“ accesskey=“1“>  
Back to the Homepage  
</a>
```

# Debugging & Logging

*alert() is obsolete and System.out.println() is not available.*

- GM\_log()
  - » Ausgabe auf der JavaScript Error-Console
  - » Zu finden unter [Extras]-> [Fehlerconsole]





# Live Scripting

*Wir schreiben selbst Skripte*

- [linux.hs-heilbronn.de](http://linux.hs-heilbronn.de)
- [youtube.com](http://youtube.com)

## ■ Andere Browser

### » Opera

- Standardmäßig integriert
- Kompatibel zu Greasemonkey Scripts

### » InternetExplorer

- Turnabout

## ■ Proxy

### » Privoxy

- Browserunabhängig
- Kein Plugin
- Filter, Blocker

## ■ IPTables

- » „Spaß mit offenen AccessPoints“

- [http://developer.mozilla.org/en/docs/Gecko\\_DOM\\_Reference](http://developer.mozilla.org/en/docs/Gecko_DOM_Reference)
- [http://developer.mozilla.org/en/docs/Core\\_JavaScript\\_1.5\\_Guide](http://developer.mozilla.org/en/docs/Core_JavaScript_1.5_Guide)
- [http://wiki.greasespot.net/Main\\_Page](http://wiki.greasespot.net/Main_Page)
- <http://diveintogreasemonkey.org/>
- <http://userscripts.com/>
- <http://www.getfirebug.com/>
- <http://chrispederick.com/work/web-developer/>

**<!--Happy  
Scripting-->**

