

Unobtrusive Javascript - Javascript unaufdringlich

Dirk Ginader

<http://blog.ginader.de>



WEB STANDARDS
WEBKRAUTS

Webstandards für ein besseres Web

Webkrauts

Die Webkrauts setzen sich dafür ein, die Vorteile der Webstandards auch im deutschsprachigen Raum stärker zur Geltung zu bringen.

Wir leisten Aufklärungsarbeit durch Veröffentlichungen im Netz und in anderen Medien.

Wir bringen das Web einen Schritt weiter.



WEB STANDARDS
WEBKRAUTS



WEB STANDARDS
WEBKRAUTS

Webstandards für ein besseres Web

Schichten einer Webseite

1. Markup: (X)HTML
Hiermit wird der Inhalt der Seite ausgezeichnet
2. Design: CSS
Hiermit wird das Aussehen der Seite bestimmt
3. Verhalten: Javascript
Hiermit werden Interaktionen mit der Seite definiert

Vermischung → Böse!



Trennung der Schichten

- Das Markup gehört in die Seite. Als einzige der Schichten
- Das Design wird in eine CSS-Datei ausgelagert und verlinkt:

```
<link href="standard.css" rel="stylesheet" type="text/css" media="screen, projection" />
```

- Das Verhalten wird in eine JS-Datei ausgelagert und verlinkt:

```
<script type="text/javascript" src="jquery.js"></script>
```



Auch kein CSS im Javascript

- CSS gehört nicht ins HTML
- Javascript gehört auch nicht ins HTML
- CSS gehört aber auch nicht ins Javascript:

Böse:

```
document.getElementById("bar").style.color = "red";
```

Statt dessen:

```
<style>
```

```
.active{color:red;}
```

```
</style>
```

```
document.getElementById("bar").style.className = "active";
```



Vorteile der Trennung

- Bessere Wartbarkeit
- Unabhängigkeit von Browsertechniken
- Reduzierung von Barrieren
- Besseres Caching
- Bessere Indizierbarkeit durch Suchmaschinen



Der Javascript Aufruf

- **Ganz früher:**

```
<body onload="machwas () ">
```

Nachteil: wird zu spät ausgeführt

- **Später:**

```
<script type="text/javascript">
```

```
  //<![CDATA[
```

```
htmlLoaded();
```

```
  //]]>
```

```
</script>
```

```
</body>
```

Vorteil: Wird direkt ausgeführt

Nachteil: Immer noch Javascript im Markup



DOMloaded

- Nicht auf Bilder warten und kein Javascript im HTML
- Crossbrowser
- Eingebaut in jQuery, Mootools, YUI, etc.
`$(document).ready(function() {machwas();});`
- Geht auch mit Prototype dank Low Pro:
`Event.onReady(function() {machwas();});`
- Oder Standalone:
`DomLoaded.load(function() {machwas();});`



Beispiele

- <https://www.closefinance.de>
(Unobtrusive JS Prä-Domloaded)
- <http://www.baenninger.info>
(Unobtrusive JS mit Ajax mit jQuery)
- <http://baustelle.onvert.de>
(Unobtrusive JS mit jQuery)



Beispiel

Beispiel aus dem Leben:

2 Tage vor dem Barcamp Frankfurt rief mich Ansgar an...

<http://www.best-of-accessibility.de/index.php/anmeldung>

Das jQuery-Script aus meinem Beispiel ist inzwischen online. Mission erfüllt... ;-)



Beispiel

Komplexeres Beispiel aus dem Leben:

Während des Barcamp Frankfurt fragte mich Dirk Jesse ob ich ihn jQuery-technisch etwas unterstützen könnte...

<http://www.yaml.de>

Die vorgestellte neue jQuery Tabs Extension „YAML Tabs“ wird in Kürze zusammen mit der neuen YAML-Version released.



Links

- Präsentation zu Progression Enhancement:
<http://icant.co.uk/articles/HighlandFling.pdf>
- Barrierefreies Javascript:
<http://ichwill.net/> (de)
<http://onlinetools.org/articles/unobtrusivejavascript/> (en)

