

Virtuelle Grenzen überwinden
Technische Barrieren abbauen

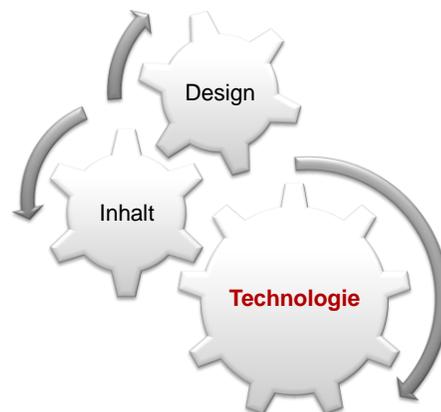


29. Februar 2008
Marcel Albertin, CTO

www.namics.com

Agenda.

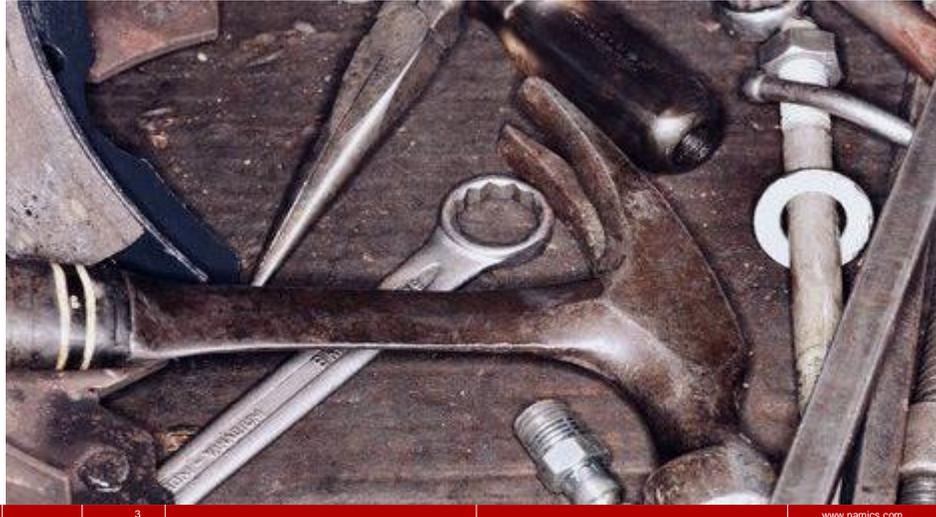
- » Voraussetzungen schaffen
- » Umsetzung durchführen
- » Erfolgskontrolle gewährleisten



2

www.namics.com

Voraussetzungen schaffen.



3

www.namics.com

Voraussetzungen schaffen.

Know-how Aufbau: CSS

- » Verwendung von Cascading Stylesheet (CSS)
 - Trennung von Inhalt & Design
- » Herausforderungen
 - Medienspezifischer Einsatz funktioniert - wenn überhaupt - nur eingeschränkt mit Hilfsmitteln
 - Darstellung in verschiedenen Browsern
 - Endgeräte-Unabhängigkeit
 - Lesbarkeit bei ausgeschaltetem CSS muss gewährleistet sein

4

www.namics.com

Voraussetzungen schaffen.

Know-how Aufbau: Beispiel CSS

» Media Typen CSS 2

- all: Ausgabe auf allen Endgeräten
- aural: Ausgabe für Synthesizer
- braille: Ausgabe für Braille-Zeile
- embossed: Ausgabe für Braille Drucker
- handheld: Ausgabe auf Handheld-Geräten
- print: Ausgabe für Print
- projection: Projizierte Präsentationen
- screen: Farbige Bildschirmausgabe
- tty: Für Ausgabegeräte mit fest-definierter Zeichendarstellung
- tv: Darstellung auf dem Fernseh

```
@media print {
  BODY { font-size: 10pt }
}
@media screen {
  BODY { font-size: 12pt }
}
@media screen, print {
  BODY { line-height: 1.2 }
}
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE>link to a target medium</TITLE>
<LINK rel="stylesheet" type="text/css"
  media="print, handheld" href="foo.css">
</HEAD>
<BODY>
<>The body...
</BODY>
</HTML>
```

Voraussetzungen schaffen.

Know-how Aufbau: Lesbarkeit



- [Zum Inhalt springen](#)
- [Zur Navigation springen](#)

[drucken](#) [schliessen](#)

 Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

[Bundesverwaltung admin.ch](#) ⇌
Eidgenössisches Departement für Verteidigung,
Bevölkerungsschutz und Sport

- [Startseite](#)
- [Übersicht](#)
- [Kontakt](#)
- [Wo finde ich was?](#)

- [Français](#)
- [Italiano](#)
- [English](#)

- [Aktuell](#)
- [Themen](#)
- [Dokumentation](#)
- [Das VBS](#)

Stichworte

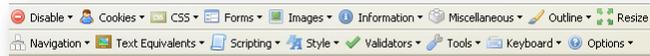
- [Alarmierung](#) ⇌
- [Aufgebotsdaten \(Schulen, WK\)](#) ⇌

Voraussetzungen schaffen.

Know-how Aufbau: Toolbars

» Toolbars

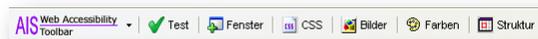
- Firefox: Developer-Toolbar, Accessibility-Toolbar



- Firefox: Firebug



- Internet Explorer: AIS Accessibility Toolbar



7

www.namics.com

Voraussetzungen schaffen.

Know-how Aufbau: AJAX

» AJAX (Asynchronous JavaScript and XML)

- Asynchrone Datenübertragung zwischen Browser und Webserver → Keine Page-Reloads mehr notwendig, sondern nur noch partielles Nachladen bei Bedarf

» Herausforderungen

- Technisch: Screenreader muss das automatische Update lesen können
- Bedienbarkeit: “Überforderung der Benutzer”

» Zwischenlösungen parat haben

» Zukunft: Role-Model

- W3C: Embedding Accessibility Role and State Metadata in HTML Documents

8

www.namics.com

Voraussetzungen schaffen.

Know-how Aufbau: AJAX Zwischenlösungen

- » Buttons anbieten, welche das automatische Update (XHR) verhindern
- » Einschalten von Alerts ermöglichen, welche über neue Informationen in Kenntnis setzen
- » Bei automatischen Updates sollte keine Fokusänderung vorgenommen werden
- » Neue Informationen sollten optisch ausgezeichnet werden (z.B. 5 Sekunden fett, kein Blinken)
- » Die Code-Strukturierung mit Überschriften etc. ist essentiell
- » Neue Content-Elemente sollten upgedated und nicht hinzugefügt werden. Wenn es nicht anders geht, dann als "parent". Das heisst, eindeutig zum vorherigen Element dazugehörend auszeichnen

Voraussetzungen schaffen.

Know-how Aufbau: PDF

- » Generierung von PDF-Dateien
 - PDF-Dateien müssen gemäss P028 barrierefrei zur Verfügung gestellt werden, sofern keine Alternative besteht
- » Herausforderungen
 - Einsatz einer Dokumentvorlage mit hierarchischer Strukturierung
 - Überarbeitung von nicht barrierefreien Dokumenten
 - Verwendung der richtigen Tools
 - Schulung der Autoren

Voraussetzungen schaffen.

Auswahl des CMS

- » Passendes CMS für die Anforderungen finden
 - Front-/Backend
- » Herausforderungen
 - Quellcode des Rich-Text-Editor
 - Interpretation von leeren ALT-Attributen
 - Auszeichnung externer Quellen aus einem Repository
 - Ausgabe von validem (X)HTML

Umsetzung durchführen.



Umsetzung durchführen.

Quellcode Strukturierung

» Listen

- Konsequenter Einsatz von Listen
- Verschachtelte Navigation
- Überspringmöglichkeit von Linklisten

```
<ul>
  ■ <li> Mehr über uns </li>
  ■ <li> Adressen und Telefonnummern </li>
</ul>
```

» Tabellen

- Keine Layout-Tabellen
- SUMMARY-Attribut oder noch besser Beschreibung im Inhalt
- Verknüpfung von Überschriften und Zellen
- Lineare Visualisierung

RTV	Preis	Profischnitt	Konzept/Design	Technik	Pflege	sonst. oder Kundespezifische
Bereitstellung äquivalenter Alternativen						
1	1	Alternativen für Bildschirmtext	-	-	+	-
1	1	Alternativen für Grafiken und Bilder	-	-	+	-

Umsetzung durchführen.

Einsatz von Scripts

» Einsatz von Java-Script

- Gemäss Richtlinien muss es auch ohne funktionieren!
- Kontakt-Link MUSS ohne Java-Script aufrufbar sein
- Menschen, welche Hilfsmittel verwenden, lassen sich nicht daran erkennen, ob sie Java-Script einsetzen oder nicht
- Tastaturzugänglichkeit
- Einsatz von Java-Script darf die Verwendung nicht stören → Kein Zwang Java-Script abzustellen

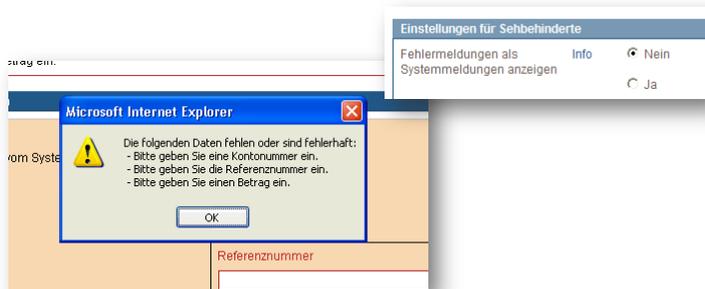
```
...
<script type="text/vbscript">
<!--
document.write("Hello World")
!-->
</script>
</noscript>Your browser does not support VBScript!</noscript>
...
```

» Verzicht auf Java-Applets

Umsetzung durchführen.

Einsatz von Scripts: Beispiele

- » Alerts anstatt "Einblendungen"
- » Session Ablauf ankündigen und Benutzer-Aktion einfordern
- » Beispiel: Credit Suisse Direct Net



15

www.namics.com

Umsetzung durchführen.

Frames

- » Einsatz von Frames
 - Diverse Usability-Probleme
 - Back-Button funktioniert eingeschränkt
 - Bookmarking nur mit Hilfsmitteln
 - <noscript> für Frames und
 - für i-frames zwischen Tags

```
<FRAMESET cols="20%, 80%">
<FRAMESET rows="100, 200">
  <FRAME src="contents_of_frame1.html">
  <FRAME src="contents_of_frame2.gif">
</FRAMESET>
<FRAME src="contents_of_frame3.html">
<NOFRAMES>
  <P>This frameset document contains:
  <UL>
    <LI><A href="contents_of_frame1.html">Some neat contents</A>
    <LI><IMG src="contents_of_frame2.gif" alt="A neat image">
    <LI><A href="contents_of_frame3.html">Some other neat contents</A>
  </UL>
</NOFRAMES>
</FRAMESET>
```

```
<iframe src="http://www.my_site.com"
base="zone1" height="350" width="450">
  Leider unterstützt Ihr Browser keine
  lokalen Frames.
</iframe>
```

16

www.namics.com

Umsetzung durchführen.

Tastaturzugänglichkeit

- » Geräteunabhängigkeit
 - Auf Mouse-over verzichten
 - onfocus verwenden
 - onclick-Events durch onkey + onlick-Events ersetzen
- » DropDowns
 - dürfen nicht automatisch weiterleiten → Button zur Auswahl nicht vergessen
- » Flyouts
 - müssen auf allen Ebenen mit der Tastatur zugänglich sein

Umsetzung durchführen.

Tastaturzugänglichkeit: Beispiele

- » Standard-HTML DropDown
 - tastaturzugänglich
 - ohne Java-Script funktionstüchtig
- » DropDown mit CSS realisiert (Layer)
 - tastaturzugänglich ab Internet Explorer 7 und Firefox 2
 - ohne Java-Script funktionstüchtig



Umsetzung durchführen.

Tabulatorenreihenfolge

- » Saubere Strukturierung ist Schritt 1, danach kommt die Tabulatorenreihenfolge!
- » Die Tabulatorenreihenfolge muss der visuellen Logik entsprechen
- » Index muss nicht abfolgend sein


```

<a href="Eins.html" title="Hier gehts lang" tabindex="1">Link 1</a>
<a href="Zwei.html" title="Hier gehts lang" tabindex="2">Link 2</a>
<a href="Drei.html" title="Hier gehts lang" tabindex="3">Link 3</a>
    
```

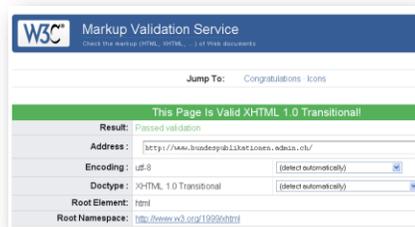
Erfolgskontrolle gewährleisten.



Erfolgskontrolle gewährleisten.

Validität des Quellcodes

- » HTML-Validität ist der erste Schritt zur Barrierefreiheit!
- » Quellcode-Überprüfung mit W3C-Validator
 - Richtlinien fordern 100% Kompatibilität
 - Browser haben nach wie vor Probleme mit validem Quellcode
 - Abwägung der “Fehler”
 - Ausnahme Amaya



Erfolgskontrolle gewährleisten.

Barrierefreier Quellcodes

- » Für Entwickler
 - Lokal:  A-Prompt (kostenlos)
<http://www.wob11.de/apromptprogramm.html>
- » Für Redaktoren & Entwickler
 - Website (ASP oder Inhouse): Watchfire
<http://www.watchfire.com>



Erfolgskontrolle gewährleisten.

Testen

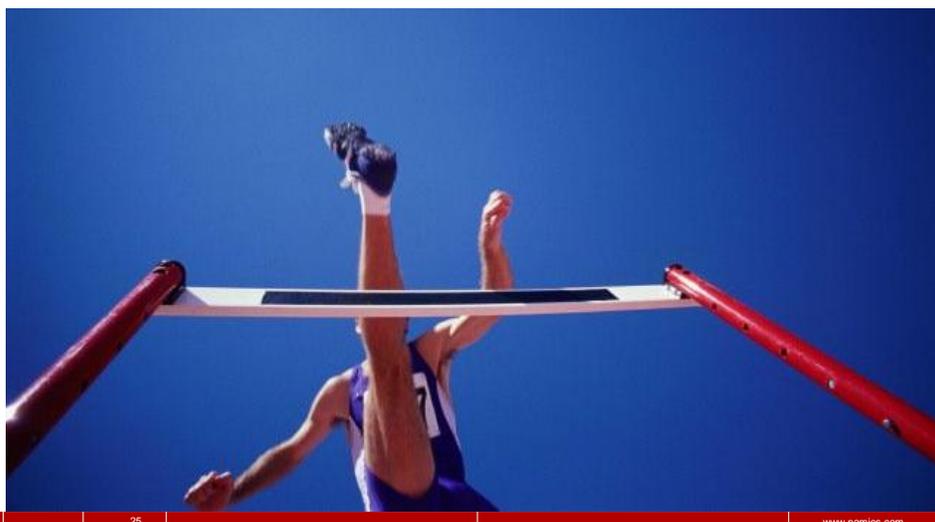
- » Je früher, desto besser!
- » Immer auch von Externen testen lassen, die eigenen Entwickler werden es immer bedienen können
- » Die Zielgruppe MUSS testen – das sind die zukünftigen Benutzer
- » Tools der Zielgruppe zum Testen verwenden

Erfolgskontrolle gewährleisten.

Weiterentwicklung absichern

- » Erstellung von Guidelines für Entwickler
 - Definition der Qualitätsstandards
 - Anleitung für das Testing
 - Definition von Standardprozeduren und Dokumente
- » Zusätzliche Hilfestellungen für den Know-how-Transfer anbieten
 - Wiki
 - Blog

Technische Barrieren abbauen.



25

www.namics.com

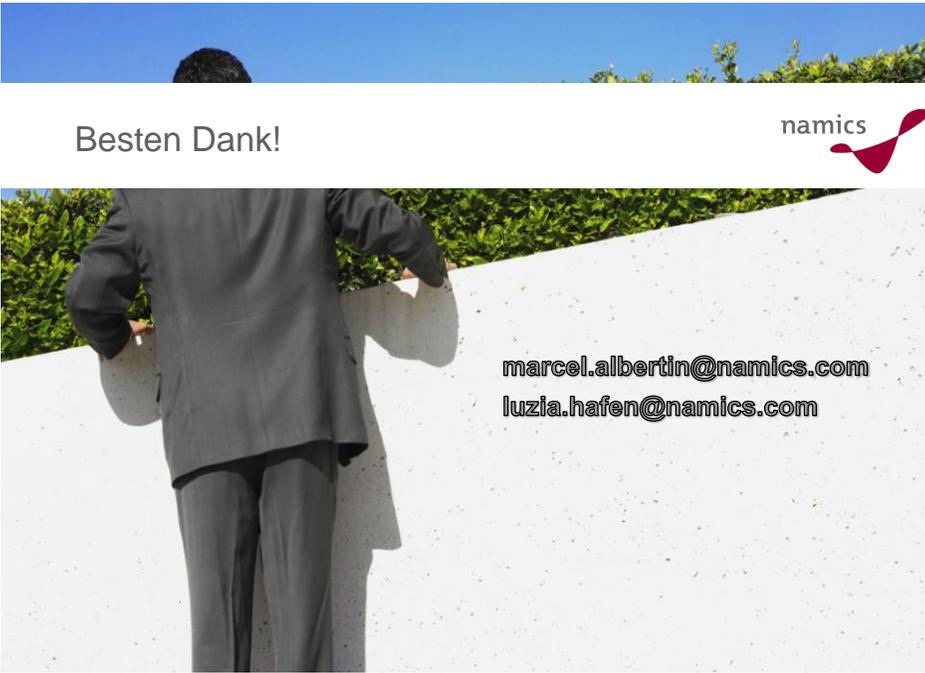
Technische Barrieren abbauen.

Das Erfolgsrezept

- » Know-how-Aufbau
 - Beginn zum Projektstart
 - Einbezug der Informationsträger in alle Projektschritte beginnend bei der Ausschreibung
- » Templating nicht unterschätzen
 - Qualität braucht Zeit
- » Testen
 - Zum Projektende ist zu spät
 - Zielgruppe beim Testen miteinbeziehen
- » Guidelines für Entwickler sichern, damit **technische Barrieren abgebaut werden können!**

26

www.namics.com

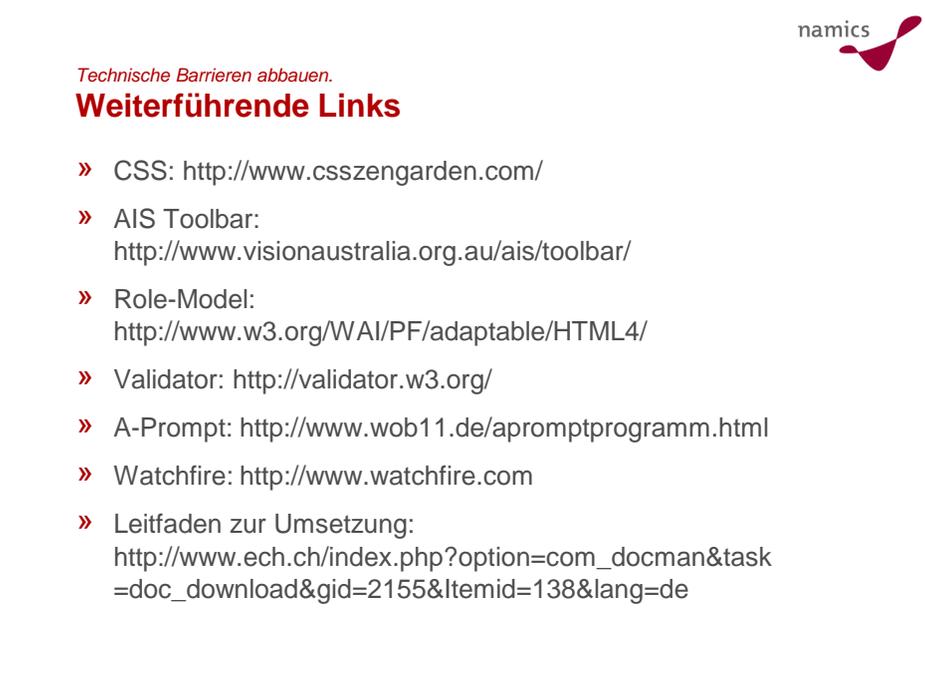


Besten Dank!

namics 

marcel.albertin@namics.com
luzia.hafen@namics.com

27 www.namics.com



namics 

Technische Barrieren abbauen.

Weiterführende Links

- » CSS: <http://www.csszengarden.com/>
- » AIS Toolbar:
<http://www.visionaustralia.org.au/ais/toolbar/>
- » Role-Model:
<http://www.w3.org/WAI/PF/adaptable/HTML4/>
- » Validator: <http://validator.w3.org/>
- » A-Prompt: <http://www.wob11.de/apromptprogramm.html>
- » Watchfire: <http://www.watchfire.com>
- » Leitfaden zur Umsetzung:
http://www.ech.ch/index.php?option=com_docman&task=doc_download&gid=2155&Itemid=138&lang=de

28 www.namics.com