

CSS & HTML
dos and don'ts
when styling multilingual websites

Gunnar Bittersmann
brands4friends

Road ahead

1. pitfalls when styling multilingual websites
2. what's new in CSS 3
3. what's not in CSS (yet?)

problem: word length

Einige Sprachen, wie z.B. Deutsch, Finnisch und Niederländisch, verwenden einzelne lange Wörter dafür, was mehrere kürzere Wörter in anderen Sprachen sind.

Aus „Input processing features“ im Englischen wird „Eingabeverarbeitungsfunktionen“ im Deutschen. Der englische Text kann einfach auf zwei Zeilen verteilt werden, wenn

nur eine geringe Breite zur Verfügung steht, bspw. neben einem Formulareingabefeld, in einer Reihe von Tabs oder Buttons, in schmalen Spalten. Der deutsche Text hingegen wird nicht automatisch umbrochen, was eine Herausforderung für das Layout darstellen kann.

solution: soft hyphens

Einige Sprachen, wie z.B. Deutsch, Finnisch und Niederländisch, verwenden einzelne lange Wörter dafür, was mehrere kürzere Wörter in anderen Sprachen sind.

Aus „Input processing features“ im Englischen wird „Eingabe-verarbeitungs-funktionen“ im Deutschen. Der englische Text kann einfach auf zwei Zeilen

verteilt werden, wenn nur eine geringe Breite zur Verfügung steht, bspw. neben einem Formular-eingabe-feld, in einer Reihe von Tabs oder Buttons, in schmalen Spalten. Der deutsche Text hingegen wird nicht automatisch umbrochen, was eine Herausforderung für das Layout darstellen kann.

Setting break points is only feasible for own content, but often content is provided by other departments of the company (CMS) or is user-generated content .

solution: hyphenation

Einige Sprachen, wie z.B. Deutsch, Finnisch und Niederländisch, verwenden einzelne lange Wörter dafür, was mehrere kürzere Wörter in anderen Sprachen sind.

Aus „Input processing features“ im Englischen wird „Eingabeverarbeitungsfunktionen“ im Deutschen. Der englische Text kann einfach

auf zwei Zeilen verteilt werden, wenn nur eine geringe Breite zur Verfügung steht, bspw. neben einem Formulareingabefeld, in einer Reihe von Tabs oder Buttons, in schmalen Spalten. Der deutsche Text hingegen wird nicht automatisch umbrochen, was eine Herausforderung für das Layout darstellen kann.

Text language must be labeled correctly, also of text fragments in different languages

GERMAN VERSION

dokument-zeichensatz

Der Dokument-Zeichensatz von XML und HTML (ab Version 4.0) ist Unicode (ISO 10646). Das heißt, dass HTML-Browser und XML-Prozessoren sich so verhalten sollten, als ob sie intern Unicode verwenden. Das bedeutet aber nicht, dass die Dokumente als Unicode übertragen werden müssen. Wenn sich Client und Server über die Zeichencodierung verständigen, können sie jede Codierung verwenden, die in Unicode umgewandelt werden kann. Siehe Weiteres zum Dokument-Zeichensatz.

zeichencodierung angeben

englische Original als maßgeblich angenommen werden. Das W3C besitzt das Copyright am Original, wie unten beschrieben.

Übersetzer: Gunnar Bittersmann

118n-Suche

Startseite > Ressourcen > Artikel

auf dieser seite

Dokument-Zeichensatz

The screenshot shows the developer tools interface with the following content:

- HTML Panel:** Shows the document structure with a selected `<h2>` element. The code includes a boilerplate, a section header, and an audience section.
- Styles Panel:** Shows the CSS rules for the selected `h2` element. The rule is: `h2 { color: #005A9C; font-size: 140%; font-weight: normal; text-align: left; text-transform: lowercase; }`. The `text-transform: lowercase;` property is circled in red.
- File Name:** `articl...-v2.css (Zeile 283)`

Nouns are spelled with a capital letter in German language; transformation to lowercase should not be applied to German text.

CORRECTED GERMAN VERSION

Dokument-Zeichensatz

Der Dokument-Zeichensatz von XML und HTML (ab Version 4.0) ist Unicode (ISO 10646). Das heißt, dass HTML-Browser und XML-Prozessoren sich so verhalten sollten, als ob sie intern Unicode verwenden. Das bedeutet aber nicht, dass die Dokumente als Unicode übertragen werden müssen. Wenn sich Client und Server über die Zeichencodierung verständigen, können sie jede Codierung verwenden, die in Unicode umgewandelt werden kann. Siehe Weiteres zum Dokument-Zeichensatz.

Zeichencodierung angeben

englische Original als maßgeblich angenommen werden. Das W3C besitzt das Copyright am Original, wie unten beschrieben.

Übersetzer: Gunnar Bittersmann

118n-Suche

Startseite > Ressourcen > Artikel

Auf dieser Seite

Dokument-Zeichensatz

The screenshot shows the browser's developer tools with the following content:

- HTML panel: `<div id="boilerplate">`, `<h1>Zeichencodierungen</h1>`, `<div class="section">`, ``, `<div id="audience">`, `<h2>`
- Styles panel: `h2:lang(de) { text-transform: none; }` (circled in green), `h2 { color: #005A9C; font-size: 100%; }`

Additional German language-specific rule overwrites transformation.

German Version

Numerische Zeichenreferenzen und **Zeichen-Entity-Referenzen** sind Arten von Zeichen-Escapes, die im Markup verwendet werden können. Das geschützte Leerzeichen (U+00A0 NO-BREAK SPACE) bspw. lässt sich folgendermaßen repräsentieren:

(Das geschützte Leerzeichen sieht wie ein normales Leerzeichen aus, verhindert aber einen Zeilenumbruch zwischen den Zeichen links und rechts davon. Es findet u.a. in der französischen Zeichensetzung Verwendung, wo vor Satzzeichen wie Doppelpunkten und Ausrufezeichen ein Leerzeichen steht, die Satzzeichen aber nicht am Anfang einer neuen Zeile stehen sollen.)

` `

Eine hexadezimale numerische Zeichenreferenz. Alle numerischen Zeichenreferenzen beginnen mit '&#' und enden mit ';'. Das 'x' zeigt an, dass das Nachfolgende eine Hexadezimalzahl ist, die den Zeichencode eines Unicode-Zeichens angibt, d.h. die dem Zeichen in den Unicode-Code-Charts zugewiesene Nummer. Die Angabe ist nicht case-sensitiv (d.h. Groß-/Kleinschreibung spielt keine Rolle).

Verschiedene Spezifikationen verwenden verschiedene Bezeichnungen für diese Konstrukte. Die HTML 5-Spezifikationen bspw. bezeichnet Zeichen-Entity-Referenzen als **named character references** (benannte Zeichenreferenzen). In diesem Artikel haben wir solche Namen gewählt, von denen wir hoffen, dass sie für den Leser wiedererkennbar und verständlich sind, welche Variationen er bisher auch verwendet haben mag.

Wann keine Escapes zu verwenden sind

Wann Escapes zu verwenden sind

Verwendung von Escapes in style-Attributen

Übrigens

Literaturhinweise

Verwandte Links

[Zeichen-Entity-Referenzen in HTML 4](#)

Einige Browser erlauben es, das Semikolon am Ende einer numerischen Zeichenreferenz wegzulassen. Dies ist aber nicht empfehlenswert, denn es kann zu Interoperabilitätsproblemen führen. Wenn ein Semikolon am Ende steht, umgeht das auch das Problem, dass das Ende des Escapes nicht zu erkennen wäre, wenn das Escape im Text eingebettet ist.

Long words may cause short lines or large spaces. Note that there's enough space between the menu and the sidenote on the right...

ukrainian version

NCR (числові посилання) та **посилання на комбінацію символів** є різновидами екранованих символів, які використовуються в розмітці. Наприклад, нижче наведено різні способи представлення символу U+00A0 NO-BREAK SPACE.

(Символ NO-BREAK SPACE виглядає як простір, але і запобігає обтіканню рядків з обох сторін між символами. У Французькій він часто використовується із знаками пунктуації, такими як двокрапки й знаки оклику, які передують простору, але не повинні з'являтися на початку рядка під час обтікання текстом)

** **

Шістнадцяткові NCR. Всі NCR починаються із '&#' і закінчуються ';'. 'x' вказує на те, що далі слідує шістнадцяткове число, яке представляє місце коду Unicode символу. Шістнадцяткове число не чутливе до регістра.

Приклад: `<p>Vive la France !</p>`

** **

Десяткові NCR. Використовується десяткове число для

Різні специфікації дають різні імена для цих конструкцій. Наприклад, специфікація HTML5 називає посилання на комбінацію символів **named character references**. Ми вирішили використовувати назви для цієї статті, що ми сподіваємося, явно зрозумілі для читача, незалежно від змін, які вони використовували до сих пір.

Екрановані символи CSS

Коли не слід використовувати екрановані символи

Коли варто використовувати екрановані символи

Використання екранованих символів в атрибутах стилю

Доречі

Додаткові матеріали

ПОСИЛАННЯ

Посилання на комбінації символів в HTML 4

Деякі браузери дозволяють пропускати крапку з комою в кінці числового посилання, але це не рекомендується, так як це може привести до проблем з сумісністю. Використання крапки з комою також дозволяє уникнути потенційних проблем в кінці екранованого символу стає невидимою, коли екранований символ вкладений в текст.



... as the menu needs more height the space between the menu and the sidenote shrinks...

RUSSIAN VERSION

NCR (числовые ссылки) и ссылки на комбинацию символов являются разновидностями экранированных символов, используемых в разметке. Например, ниже приведены различные способы представления символа U+00A0 NO-BREAK SPACE.

(Символ NO-BREAK SPACE выглядит как пространство, но и предотвращает обтекание строк с обеих сторон между символами. Во Французской он часто используется с такими знаками препинания, как двоеточия и восклицательные знаки, которые предшествуют пространству, но не должны появляться в начале строки при обтекании текстом)

Шестнадцатеричные NCR. Все NCR начинаются с '&#' и заканчиваются ';'. 'x' означает, что далее следует шестнадцатеричное число, представляющее точку кода Unicode символа. Шестнадцатеричное число не чувствительно к регистру.

Пример: `<p>Vive la France !</p>`

Различные спецификации дают разные имена для этих конструкций. Например, спецификация HTML5 называет ссылки на комбинацию символов **named character references**. Мы решили использовать названия для этой статьи, мы надеемся, явно понятные читателю, независимо от изменений, которые они использовали до сих пор.

Какие экранированные символы можно использовать в разметке?

Экранированные символы CSS

Когда не следует использовать

экранированные символы

Когда следует использовать экранированные символы

Использование экранированных символов в атрибутах стиля

Китати говоря

Дополнительные материалы

Связанные ссылки

Некоторые авторы предпочитают пропускать точку с запятой в конце числовой ссылки, но это не рекомендуется, так как это может привести к проблемам совместимости. Использование точки с запятой также позволяет избежать потенциальных проблем в конце экранированного символа становится невидимой, когда экранированный символ вложенный в текст.

... as the menu needs even more height the contents overlap.

ARABIC VERSION

مفتاح الوصول n يتخطى إلى



Polski English
Português-BR
繁體中文 简体汉语

هذه الوثيقة ترجمة للأصل.
في حالة وجود تعارض أو
أخطاء، تكون أحدث نسخة
من المستند الإنجليزي
الأصلي هي التي يُعتد بها.
حقوق النسخ الأصلية
مملوكة بواسطة W3C، كما
هو موضح بأدناه.

إيسية للعولمة

ب الموضوعات الأخبار المجموعات حول

حول العولمة استخدام توى مترجم

XHTML/HTML coders (using editors or scripting developers (PHP, JSP, etc.), Web project managers better understand how to use the select element

The screenshot shows a browser window with a horizontal scrollbar at the bottom, circled in red. Below the browser, the developer tools are open, showing the HTML structure and CSS styles for the #wai-start element. The HTML structure shows a `<div id="wai-start">` element. The CSS styles for #wai-start are:

```
#wai-start {  
  display: block;  
  left: -999px;  
  position: absolute;  
  width: 990px;  
}
```

The `left: -999px;` style is circled in red.

A horizontal scrollbar appears because of content for screenreaders that should be moved to the right out of the viewport on RTL pages.

ARABIC VERSION

الصفحة الرئيسية للعولمة W3C

الصفحة الرئيسية المصادر الأساليب الموضوعات الأخبار المجموعات حول

Polski English
Português-BR
繁體中文 简体汉语

هذه الوثيقة ترجمة للأصل.
في حالة وجود تعارض أو
أخطاء، تكون أحدث نسخة
من المستند الإنجليزي

أسئلة W3C الشائعة حول العولمة استخدام <select> للربط بمحتوى مترجم

Konsole HTML CSS Skript DOM Netzwerk

Bearbeiten div#siteicons < div#boil

```
<div id="disclaimer">  
<div id="search" class="noprin  
<div class="sidebar">  
<div id="boilerplate">  
  <div id="siteicons">  
    <div id="sitelinks" class=  
      <div id="line">&nbsp;</di  
  </div>  
<h1>أسئلة W3C الشائعة حول العولمة  
استخدام <select> للربط بمحتوى  
مترجم</h1>  
<div class="section">
```

div#siteicons {
 border-bottom-left-radius: 30px;
 box-shadow: -4px 4px 7px #888888;
 margin-left: 29%;
 margin-right: 0;
 padding-right: 0;
}

#siteicons {
 background-color: #003163;
 border-bottom-right-radius: 30px;
 box-shadow: -4px 4px 7px #888888;
 margin-right: 29%;

*border-bottom-right-radius should be resetted on RTL pages,
box-shadow should be a mirrored version of that on LTR pages.*

CORRECTED ARABIC VERSION

الصفحة الرئيسية للعولمة W3C

الصفحة الرئيسية المصادر الأساليب الموضوعات الأخبار المجموعات حول

Polski English
Português-BR
繁體中文 简体汉语

هذه الوثيقة ترجمة للأصل.
في حالة وجود تعارض أو
أخطاء، تكون أحدث نسخة
من المستند الإنجليزي
الأصلي هي التي يُعتد بها.

أسئلة W3C الشائعة حول العولمة استخدام `<select>` للربط بمحتوى مترجم

الجمهور المستهدف: XHTML/HTML coders (using editors or scripting), script

Konsole HTML CSS Skript DOM Netzwerk

Bearbeiten div#siteicons < div#boil

Styles Berechnet Layout DOM

```
div#siteicons {  
  border-bottom-left-radius: 30px;  
  box-shadow: 4px 4px 7px #888888;  
  margin-left: 29%;  
  margin-right: 0;  
  padding-right: 0;  
}
```

```
#siteicons {  
  background-color: #003163;  
  border-bottom-right-radius: 0;  
  box-shadow: 4px 4px 7px #888888;  
  margin-right: 29%;
```

article...idi.css (Zeile 168)

article...v2.css (Zeile 561)

No horizontal scrollbar, border-radius and box-shadow properly set.

separate stylesheets?

default.css:

```
#wai-start  
{  
    position: absolute;  
    left: -999px;  
    width: 990px;  
    display: block;  
}
```

rtl.css:

```
#wai-start  
{  
    left: auto;  
    right: -999px;  
}
```



*Don't rely on the order of stylesheets being included in HTML files;
use more specific selectors*

separate stylesheets?

default.css:

```
#wai-start
{
    position: absolute;
    left: -999px;
    width: 990px;
    display: block;
}
```

```
[dir="rtl"] #wai-start
{
    left: auto;
    right: -999px;
}
```

rtl.css:

```
[dir="rtl"] #wai-start
{
    left: auto;
    right: -999px;
}
```

Having all rules in one stylesheet allows to overview corresponding rules at one sight and makes it less likely to forget something on later changes.

dos and don'ts

- Do not assume text to fit into a box, neither of certain width nor of certain height
- Set soft hyphens in long compound words or use hyphenation
- Use text effects that are appropriate for given language
- Make sure you flip everything needed for websites in RTL scripts
- Do not use multiple stylesheets for different languages/scripts/text directions, use one stylesheet (easier to maintain)

new in css 3 text module

- hyphens: auto
- text-emphasis-style: sesame これは日本語の文章です。
- text-justify: inter-ideograph || kashida
- text-align: end
might replace:
#foo { text-align: right }
[dir="rtl"] #foo { text-align: left }

styling by language

```
body
{
    font-family: Palatino, serif;
}
```

:script(Arab)

```
:lang(ar), :lang(fa), :lang(ps), :lang(ur), :lang(az-Arab), :lang(uz-Arab),  
:lang(bal), :lang(bqi), :lang(cja), :lang(glk), :lang(ug)  
{  
    font-family: "Traditional Arabic", serif;  
}
```

When content in another language written in Arabic script would later be added to a multilingual website, the stylesheet also needs to be changed. That's not desired, hence it would be useful if there was a `:script` pseudoclass.

styling by script

```
<html lang="ar" script="Arab" dir="rtl">
```

Type: language

Subtag: ar

Description: Arabic

Added: 2005-10-16

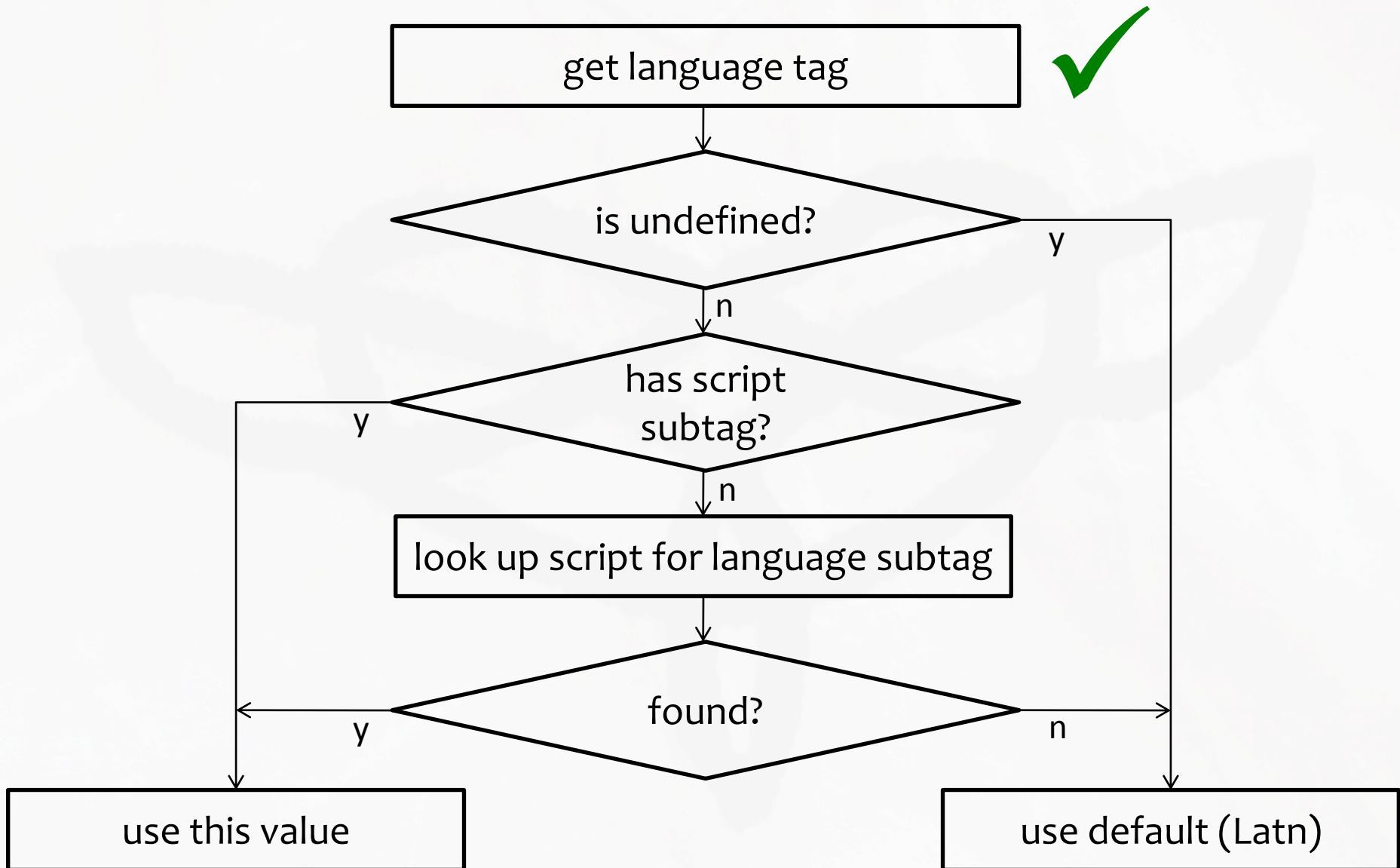
Suppress-Script: Arab

Scope: macrolanguage

```
<html lang="az-Arab" script="Arab " dir="rtl ">
```

The information about the script is already in the language tag: either in the script subtag or in the language subtag.

implementation



so raibh maith agat

gunnar@bittersmann.de

<http://bittersmann.de>