

CSS Print

Jirka Kosek

XML – teorie a praxe značkovacích jazyků (4IZ238)
Poslední modifikace: 2021-04-28 13:53:25 UTC

Papír stále ještě není mrtev

- knížky
- tištěné katalogy
- tiskové výstupy z aplikací (např. faktury)

HTML everywhere

- HTML se stává nejpoužívanějším formátem nejen na webu
- elektronické knihy EPUB/MOBI jsou jen archiv obsahující HTML+CSS
- uživatelské rozhraní dnes většinou běží v prohlížeči a je to opět HTML+CSS

Kacířská myšlenka?

Budeme HTML+CSS používat i pro generování tiskového výstupu (PDF)

- nemusíme se učit další technologii („CSS umí každý“)
- nebude potřeba konvertovat dokumenty/data

Co potřebujeme navíc oproti CSS v prohlížeči

- odlišení stylu pro tisk a obrazovku
- definice rozměrů a layoutu stránky
- záhlaví a zápatí
- generový obsah, číslování, odkazy
- kontrolu zlomu stránek a řádek
- dělení slov
- poznámky pod čarou
- obsah, rejstřík
- záložky

Odlišný styl pro tisk

- `@media print { ... }`
- `<link rel="stylesheet" media="print" href="...">`
- `<style media="print">`
 `...`
 `</style>`

Rozměry a layout stránky

```
@page { size: a5 landscape; }
```

```
/* Rozdílné nastavení pro první, sudou/lichou stránku */
```

```
@page :first { ... }
```

```
@page :left { ... }
```

```
@page :right { ... }
```

Záhlaví a zápatí

```
@page {  
  size: a4;  
  
  @top-left {  
    content: "Ze života hmyzu";  
  }  
  
  @bottom-right {  
    content: counter(page);  
  }  
}
```


Živé záhlaví/zápatí

```
<section class="kapitola">  
  <h3>Nadpis</h3>
```

...

```
section.kapitola h3 {  
  string-set: kapitola contents();  
}
```

...

```
@top-left {  
  content: string(kapitola);  
}
```

Automatické číslování

```
<section class="kapitola">  
  <h3>Nadpis</h3>
```

...

```
section.kapitola h3 {  
  counter-increment: kapitola;  
}
```

```
section.kapitola h3::before {  
  content: "Kapitola " counter(kapitola);  
}
```

Kontrola zlomu

- dokument se sází automaticky bez možnosti ruční korekce
- pomocí podmínek můžeme zajistit, že se stránky zalomí na vhodných místech
- vlastnosti: `page-break-before`, `page-break-after` a `page-break-inside`
- hodnoty: `auto`, `always`, `avoid`, `left` a `right`

```
/* Stránka se nesmí zalomit za nadpisem */  
h2 + p { page-break-before: avoid }
```

Obsah a rejstřík

- CSS neumí samo vygenerovat
- lze vytvořit ručně, pomocí XSLT, JS, ...
- CSS umí vyhodnotit odkazy a nahradit je číslem strany nebo obsahem cílového elementu

Odkazy

```
<h3 id="last">Závěr</h3>
```

...

```
<a href="#last" class="pageref">Poslední kapitola na straně</a>  
ukončuje text.
```

...

```
a.pageref::after {  
  content: " " target-counter(attr(href url), page);  
}
```

Dělení slov

```
<html lang="cs">
```

```
...
```

```
html {  
  hyphens: auto;  
}
```

Záložky

```
h1 { bookmark-level: 1; }  
h2 { bookmark-level: 2; }  
h3 { bookmark-level: 3; }
```

Poznámky pod čarou

```
<p>Text obsahující<span class="poznámka">Poznámku  
pod čarou.</span> něco.</p>
```

...

```
.poznámka { float: footnote; }
```


Výplně

- dynamické vyplnění volného místa

```
<ul class="toc">
  <li><a href="#uvod">Úvod</a></li>
  <li><a href="#stat">Stat'</a></li>
  <li><a href="#zaver">Závěr</a></li>
```

...

```
/* Přidání teček a čísla strany do obsahu */
.toc a::after {
  content: leader(dotted) " " target-counter(attr(href, url), page);
}
```

Jak to souvisí s XML?

- CSS má oproti XSL velice omezené možnosti transformace dokumentu před zobrazením
- CSS lze aplikovat přímo na XML (stejně jako na HTML), ale kromě jednoduchých příkladů to není funkční řešení
- mnoho dat, která se mají zobrazit, v XML chybí (obsah, rejstřík, popisky údajů, ...)
- nejčastěji se tak z XML pomocí XSLT vygeneruje HTML a na něj se pak aplikuje tiskové CSS

Implementace

- [Antenna House Formatter](#)
- [Prince](#)
- [PDFreactor](#)
- [Vivliostyle](#)
- [Weasy Print](#)
- [paged.js](#)

Velcí uživatelé

- Nakladatelství O'Reilly
 - HTMLBook – rozšíření HTML pro psaní knih
 - Atlas – publikační platforma používající HTMLBook
- Nakladatelství Hachette Livre
 - ročně vydají několik tisíc knih vysázených pomocí CSS

Problémy

- pomalá standardizace
- open-source implementace zatím zdaleka nedosahují kvalit těch komerčních
- neumí vše co XSL-FO

Další informace

- [Introduction to CSS for Paged Media – kniha podrobně popisující využití CSS pro generování tiskového výstupu](#)
- [CSS Generated Content for Paged Media Module](#)
- [CSS Paged Media Module Level 3](#)