

KAPITOLA 11

Rozvržení tabulek

Asi jste nyní knihu tuto otočili, abyste se podívali na její titul, protože nevěřičně zíráte. Kapitola pojmenovaná jako "Navrhování tabulek? Tomu jsme se přeci chtěli za každou cenu vyhnout!" Tahle kapitola ovšem není o tom, jak se používají tabulky pro potřeby layoutu. Je o způsobech, jimiž se používají tabulky uvnitř CSS, což je mnohem komplikovanější záležitost, než by se mohlo na první pohled zdát. To je také důvod, proč si této téma zasloužilo samostatnou kapitolu.

Tabulky jsou jedinečné, pokud je porovnáme se zbývajícimi částmi dokumentu. Od CSS 2.1 patří do výbavy samotných tabulek možnost sdružovat velikost prvků s jinými prvky – aby všechny buňky v daném řádku měly stejnou výšku, bez ohledu na to, jak mnoho (nebo naopak málo) obsahu jednotlivé buňky obsahují. Totéž platí pro šířky buněk daného sloupce. V rozvržení dokumentu neexistuje žádná taková jiná situace, kdy by prvky z různých částí stromu dokumentu mohly tímto způsobem přímo ovlivňovat rozměry a rozvržení jiných prvků.

Jak ostatně sami uvidíte – za tuto jedinečnost se platí se dodatečnými náklady, spoustou rozdílného chování a pravidly, která se aplikují pouze na tabulky. V průběhu této kapitoly se podíváme, jak se tabulky vizuálně sestavují, dozvíte se dva způsoby, jimiž se vytváří orámování buněk, a také se seznámíte s mechanismy, které ovlivňují výšku a šířku tabulek a jejich interních prvků.

Formátování tabulky

Než se začneme starat o to, jak se orámují buňky a určí rozměry tabulky, musíme nejprve probrat nezbytné základy, například to, jak se tabulky sestavují a jaké jsou vzájemné vztahy mezi jednotlivými prvky uvnitř tabulky. Tomuto se říká formátování tabulky (table formatting), a je to něco úplně jiného než rozvržení tabulky (table layout) – to druhé je možné až poté, co proběhlo to první.

Vizuální uspořádání tabulky

Nejprve si musíme ujasnit, jak CSS vykresluje orámování buněk a určuje rozměry tabulek. I když mohou někomu tyto vědomosti připadat jako až příliš základní, jsou tím hlavním klíčem k pochopení toho, jak co nejlépe ostylovat tabulky.

CSS rozlišuje mezi prvky tabulky a interními prvky tabulky. V CSS generují interní prvky tabulky pravoúhlé boxy, které mají obsah, výplň a orámování, ale nemají okraje. Proto nejde buňky tabulky od sebe oddělit tak, že jim přidělíte nějaké okraje. Prohlížeč, který pracuje v souladu s CSS, by měl ignorovat všechny pokusy o přidělení okraje buňkám, řádkům nebo jakýmkoliv jiným interním prvkům tabulky (výjimkou jsou titulky, captions, které probereme v této kapitole později.)

Uspořádání tabulek ovlivňuje šest pravidel uvedených dále. Jejich základem je "mřížka buněk", což je oblast mezi čarami mřížky, kde se vykresluje tabulka. Podívejte se na obrázek 11-1, kde jsou dvě tabulky obsahující různě pojmenované buňky. Mřížku buněk vyznačují přerušované čáry.

prison cell	cell phone
cell walls	hard cell

cellery		wide cell
soft cell	tall cell	cellulose
basal cell		end cell

Obrázek 11-1. Základ rozvržení tabulky formuje mřížka buněk.

V prosté tabulce o rozměrech dvakrát dvě buňky, jako je tabulka na obrázku 11-1 vlevo, odpovídá mřížka buněk samotným buňkám. U komplikovanějších tabulek, jako je ta na obrázku 11-1 na-pravo, odpovídají hrany mřížky buněk případnému orámování buněk, přičemž také protínají ty buňky, které se klenou přes několik řádků nebo sloupců.

Mřížky buněk jsou v podstatě pouze teoretickou konstrukcí a nedají se stylovat. Přes objektový model dokumentu se k nim dokonce ani nedá přistupovat. Prostě se jimi pouze popisuje, jak se tabulky sestavují pro ostylování.

Pravidla pro uspořádání tabulky

- Každý řádkový box obsahuje jediný řádek s buňkami. Všechny řádkové boxy v tabulce naplní tabulku zeshora dolů v pořadí, v jakém jsou uvedeny ve zdrojovém HTML dokumentu. Výjimkou jsou řádkové boxy záhlaví (resp. zápatí tabulky), které přijdou na začátek (resp. na konec tabulky). Tabulka tedy obsahuje tolik řádků, kolik je použito prvků pro řádky.
- Box skupiny řádků obsahuje stejné buňky jako řádkové boxy, které tato skupina obsahuje.
- Sloupcový box obsahuje buňky jednoho nebo několika sloupců. Všechny sloupcové boxy se kladou vedle sebe v pořadí, v jakém jsou uvedeny v kódu. První sloupcový box je v jazycích psaných zleva doprava vlevo, v jazycích psaných zprava doleva je vpravo.
- Box skupiny sloupců obsahuje stejné buňky jako sloupcové boxy, které skupina obsahuje.

- Přestože se buňky mohou klenout přes několik řádků nebo sloupců, CSS nedefinuje, jak se to má provést. Toto definuje jazyk samotného dokumentu. Každá buňka, která se klene přes několik řádků (nebo sloupců), vytváří pravoúhlý box mřížky jedné nebo více buněk na šířku a na výšku. Horní řádek tohoto obdélníku je v řádku, který je rodičem buňky. Obdélník buňky musí být v jazycích psaných zleva doprava umístěn co nejvíce nalevo, ale nesmí překrývat žádný box jiné buňky. V těchto jazycích musí být rovněž vpravo od všech buněk téhož řádku, které jsou v dokumentu uvedené před ním. V jazycích psaných zprava doleva musí být buňka klenoucí se přes více řádků (nebo sloupců) co nejvíce napravo (ale nesmí překrýt žádnou jinou buňku), a musí být nalevo od všech buněk téhož řádku, které za ním následují ve zdrojovém dokumentu.
- Box buňky nemůže sahat za box posledního řádku tabulky nebo skupiny řádků. Když struktura tabulky vytvoří takovou situaci, musí se buňka zkrátit tak, aby se vešla do tabulky nebo do skupiny řádků, do které patří.

Specifikace CSS vás sice odrazuje od pozicování buněk tabulky a jiných interních prvků tabulky, nicméně to nezakazuje. Ovšem pozicování nějakého řádku, který obsahuje buňky klenoucí se přes několik řádků, může dramaticky pozměnit rozvržení tabulky např. tím, že tento řádek se z tabulky zcela odstraní. To znamená, že v rozvržení ostatních řádků by se neměly brát v úvahu ty buňky, které do těchto řádků zasahovaly z odstraněného řádku.

Podle definice jsou buňky pravoúhlé, ale nemusejí mít stejnou velikost. Všechny buňky v nějakém sloupci musí mít stejnou šířku, všechny buňky v nějakém řádku musí mít stejnou výšku, ovšem výška buněk může být u jednotlivých řádků různá. Obdobně mohou být buňky různých sloupců různě široké (jeden sloupec tak může být široký 1em, jiný naopak 2em).

Vezmete-li výše uvedená pravidla v úvahu, možná se nyní ptáte, jak přesně se dozvíte (nebo uživatelský agent), které prvky tabulky fungují jako buňky a které ne? To zjistíte následujícím oddílu.

Hodnoty display pro tabulku

V HTML se snadno zjistí, které prvky jsou součástí tabulky, protože zpracování různých tabulkových prvků (jako `tr` a `td`) je již zabudované do samotných prohlížečů. Ovšem v XML není žádný způsob, jak se interně dozvědět, které prvky jsou součástí tabulky (a které by se tudíž měly chovat odpovídajícím způsobem). A tehdy vstupuje do hry kolekce hodnot vlastnosti `display`, které jsou určeny pro práci s tabulkami.

display

Hodnoty:	<code>none inline block inline-block list-item run-in table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption inherit</code>
Výchozí hodnota:	<code>inline</code>
Aplikuje se na:	<i>Všechny prvky</i>
Dědí se:	<i>Ne</i>
Vypočtená hodnota:	<i>Různě pro plovoucí (obtékané), pozicované a kořenové prvky (viz CSS 2.1, sekce 9.7); jinak tak, jak je specifikována.</i>
Poznámka:	<i>V CSS 2 se objevily hodnoty <code>compact</code> a <code>marker</code>, které ovšem byly z CSS 2.1 zase vyhozeny, protože se nedočkaly širší podpory.</i>

V této kapitole se soustředíme pouze na hodnoty vztahující se k tabulkám, protože ostatní (`block`, `inline`, `inline-block`, `run-in` a `list-item`) se probírají v jiných kapitolách. Takže nyní následuje pouze přehled hodnot vztahujících se k tabulce.

■ `table`

Hodnota specifikuje, že prvek definuje tabulku jako blokový prvek. Takže definuje pravoúhlý blok, který generuje blokový box. Odpovídajícím prvkem v HTML, což není žádným překvapením, je `table`.

■ `inline-table`

Hodnota specifikuje, že prvek definuje tabulku jako inline prvek. To znamená, že prvek definuje pravoúhlý blok, který generuje inline box. Nejblížeš analogická hodnota, která se netýká tabulek, je `inline-block`. Nejblížeším prvkem HTML je `table`, i když tabulky v HTML standardně nebývají inline prvky.

■ `table-row`

Specifikuje, že prvek slouží jako řádek s buňkami. Odpovídajícím prvkem v HTML je `tr`.

■ `table-row-group`

Hodnota specifikuje, že prvek tvoří skupina jednoho nebo více řádků. Odpovídajícím prvkem v HTML je `tbody`.

■ `table-header-group`

Skupina řádků záhlaví. Tato hodnota je něco velmi podobného jako `table-row-group`, ovšem z hlediska vizuálního formátování se tato skupina řádků záhlaví vždy zobrazí před všemi ostatními řádky a skupinami řádků, a také za všemi horními titulky. Při tisku – pokud tabulka potřebuje k vytištění více než jednu stránku – může uživatelský agent řádky záhlaví zopakovat na každé stránce nahoře. Specifikace ovšem neřeší, co se má stát, přiřadí-li

`table-header-group` více prvkům. Skupina záhlaví může obsahovat více než jeden řádek. Ekvivalentní prvek v HTML je `thead`.

■ `table-footer-group`

Skupina řádků zápatí. Tato hodnota je něco velmi podobného jako `table-header-group`, ovšem skupina řádků zápatí se vždy zobrazí až za všemi ostatními řádky a skupinami řádků, a před všemi dolními titulky. Při tisku – pokud tabulka potřebuje k vytištění více než jednu stránku – může uživatelský agent zopakovat řádky zápatí na každé stránce dole. Specifikace ovšem neřeší, co se má stát, přiřadíte-li `table-footer-group` více prvkům. Skupina záhlaví může obsahovat více než jeden řádek. Ekvivalentní prvek v HTML je `tfoot`.

■ `table-column`

Hodnota deklaruje, že prvek popisuje sloupec s buňkami. V termínech CSS se prvky s touto hodnotou vlastnosti `display` vizuálně nerealizují (jako kdyby měly hodnotu `none`). Jejich primárním účelem je vám usnadnit práci, když definujete, jak se mají buňky ve sloupci prezentovat. Ekvivalentní prvek v HTML je `col`.

■ `table-column-group`

Hodnota deklaruje, že prvek tvoří skupina jednoho nebo více sloupců. Podobně jako u prvků `table-column`, se ani prvky `table-column-group` vizuálně nerealizují, tato hodnota je však užitečná, když definujete, jak se mají prezentovat prvky ve skupině sloupců. Odpovídajícím prvkem v HTML je `colgroup`.

■ `table-cell`

Hodnota specifikuje, že prvek definuje jedinou buňku tabulky. Odpovídajícími prvky v HTML jsou `td` a `th`.

■ `table-caption`

Tato hodnota definuje titulek tabulky. CSS sice nedefinuje, co se má stát, bude-li mít hodnotu `caption` více prvků, ale explicitně varuje, že "...autoři by neměli dovnitř prvku `table` či `inline-table` klást více než jeden prvek s `display: caption`".

Rychlý přehled o obecných účincích těchto hodnot si uděláte, když se podíváte na výtah z ukázkového stylového předpisu HTML 4.0, který najdete v dodatku C:

```
table      {display: table;}
tr         {display: table-row;}
thead      {display: table-header-group;}
tbody      {display: table-row-group;}
tfoot      {display: table-footer-group;}
col        {display: table-column;}
colgroup   {display: table-column-group;}
td, th     {display: table-cell;}
caption    {display: table-caption;}
```

V XML, kde prvky standardně nemají sémantiku `display`, jsou tyhle hodnoty dost užitečné. Vezměme například značkování:

```
<hodnoceni>
  <zahlaví>
    <popisek>Tým</popisek>
    <popisek>Skóre</popisek>
  </zahlaví>
  <hra sport="Fotbal" uroven="Extraliga">
    <tym>
      <jmeno>Sparta</jmeno>
      <skore>1</skore>
    </tym>
    <tym>
      <jmeno>Slavia</jmeno>
      <skore>3</skore>
    </tym>
  </hra>
</hodnoceni>
```

Do tabulkové podoby se dají tyto prvky naformátovat pomocí následujících stylů:

```
hodnoceni {display: table;}
zahlaví {display: table-header-group;}
hra {display: table-row-group;}
tym {display: table-row;}
popisek, jmeno, skore {display: table-cell;}
```

Jednotlivé buňky je samozřejmě možné stylovat podle potřeby – například dát prvky `popisek` tučně a prvky `skore` jako zarovnané doprava.

I když je teoreticky možné přiřazovat hodnoty vlastnosti `display` vztahující se k tabulkám jakémukoli prvku v HTML, Internet Explorer včetně IE 7 tuto funkcionalitu nepodporuje.

Nadřazenost řádků

CSS definuje model tabulky s "nadřazeností řádků" (row primacy). Jinak řečeno – tento model předpokládá, že budou vytvářeny takové značkovací jazyky, v nichž se explicitně deklarují řádky. Sloupce se pak budou odvozovat z rozvržení řádků buněk. To znamená, že první sloupec bude skládat z prvních buněk všech řádků, druhý sloupec z druhých buněk všech řádků atd.

V HTML nedělá nadřazenost řádků prakticky žádné problémy, protože se jedná o značkovací jazyk orientovaný na řádky. Větší dopad to ovšem má v XML, protože model nadřazenosti řádků všelijak