

PB – Vyšší odborná škola a Střední škola managementu, s.r.o.

# **Absolventská práce**

2005

Jan Chlumský

PB – Vyšší odborná škola a Střední škola managementu, s.r.o.

Nad Rokoskou 111/7, Praha 8

Obor: Aplikace výpočetní techniky

Název absolventské práce:

# **Možnosti a problémy kaskádových stylů při tvorbě webových prezentací**

Školní rok: 2004/2005

Vypracoval: Jan Chlumský

Vedoucí absolventské práce: Milan Randák

Prohlášení:

Prohlašuji, že jsem absolventskou práci na téma „Možnosti a problémy kaskádových stylů při tvorbě webových prezentací“ vypracoval samostatně.

Použitou literaturu a podkladové materiály uvádím v příloženém seznamu literatury.

V Praze dne 13.6.2005

## OBSAH

Úvod .....	6
1 Internet .....	7
1.1 Historie internetu .....	7
1.2 Jak funguje Internet .....	7
1.3 WWW .....	8
1.4 Úspěch WWW .....	8
2 (X)HTML.....	10
2.1 Co je HTML.....	10
2.2 Historie a vývoj (X)HTML.....	10
2.3 Základy (X)HTML .....	12
3 Kaskádové styly (CSS).....	15
3.1 Historie a vývoj CSS.....	15
3.2 Význam a použití CSS .....	16
3.3 Metody připojení CSS k (X)HTML.....	18
3.3.1 Pomocí prvku <link> .....	19
3.3.2 Pomocí prvku <style>.....	20
3.3.3 Pomocí atributu style různých HTML prvků.....	22
3.4 Struktura stylového předpisu .....	22
3.4.1 Deklarace .....	23
3.4.1.1 Hodnoty .....	23
3.4.2 Selektory .....	25
3.4.2.1 Typový selektor (selektor HTML značky).....	25
3.4.2.2 Univerzální selektor .....	26
3.4.2.3 Selektor třídy .....	26
3.4.2.4 Selektor ID .....	27
4 Praktická část .....	28
4.1 Box model.....	28
4.2 CSS hack.....	30
4.2.1 Podtržítkový hack .....	30
4.2.2 Hack s uvozovkami.....	31
4.2.3 Hack se zpětným lomítkem.....	31
4.2.4 Hack s rovnítkem a uvozovkami .....	32
4.2.5 Star html hack .....	32

4.2.6	Çelikův hack .....	32
4.2.7	Validita a používání hacků .....	33
4.3	Vertikální centrování .....	33
4.3.1	Pomocí tabulky .....	33
4.3.2	Yuhůovo řešení .....	34
4.3.3	Victoriino vertikální centrování.....	36
4.3.4	Shrnutí.....	37
4.4	Internet Explorer 7 .....	38
4.4.1	Bezpečnost .....	38
4.4.2	Zobracovací jádro a standardy .....	38
4.4.3	Uživatelské rozhraní .....	39
4.4.4	Rozšíření .....	39
4.4.5	Shrnutí.....	40
4.5	Analýza www.seznam.cz .....	41
4.5.1	Rozbor layoutu.....	42
4.5.2	CSS .....	46
	Závěr.....	54
	Resumé .....	55
	Seznam použité literatury .....	56
	Seznam tabulek a vyobrazení .....	57

# Úvod

V posledních letech zaznamenal internet v České republice, ale i jinde ve světě velký rozvoj. Prakticky každý má možnost přístupu k internetu. Může zde najít potřebné informace (např. počasí, zpravodajství, a další informace), nebo může naopak informace na internetu publikovat pomocí své webové prezentace. Vytvořit si vlastní webovou prezentaci není dnes pro běžného uživatele větší problém a to zejména díky softwaru, který umožňuje jednoduché vytvoření webových stránek. Formátování obsahu webových prezentací se provádí přímo v (X)HTML kódu vytvořeného dokumentu. V poslední době je snaha, vyhnout se formátování v (X)HTML kódu a formátovat dokument pomocí kaskádových stylů (dále CSS).

Důvodem, proč jsem si vybral toto téma je, že se věnuji tvorbě webových prezentací a v budoucnosti se jí chci nadále věnovat.

Má práce má více cílů, jedním z hlavních cílů je zjištění možností a problémů zobrazení CSS v různých webových prohlížečích (kompatibilita webových prohlížečů s CSS). Dále se zaměřím na výhody a nevýhody CSS a pokusím se nastínit jejich budoucnost. Posledním cílem, je shrnutí poznatků o CSS do uceleného dokumentu, který bude obsahovat veškeré důležité základní informace o CSS a který bude možné použít jako průvodce začínajících webdesignerů, a webmasterů.

Hlavním zdrojem při tvorbě mé práce bude internet a to zejména weby týkající se tvorby webových prezentací a CSS. Budu používat české i zahraniční publikace týkající se CSS a vlastní zkušenosti s CSS a tvorbou webových prezentací.

V úvodu teoretické části se budu stručně věnovat historii a vývoji internetu, další kapitolou bude tvorba a formátování webových stránek prostřednictvím jazyka (X)HTML, kde bych se chtěl věnovat hlavně výhodám a nevýhodám této metody. Následovat bude historie a vývoj kaskádových stylů, kde vás seznámím s tím, jak CSS vznikly a jak se vyvíjely. Zmíním zde samozřejmě i význam CSS při tvorbě webových prezentací, k čemu jsou dobré a proč je používat. Dále se budu zabývat propojením CSS a (X)HTML a formátováním webových stránek pomocí CSS.

V poslední části práce se budu věnovat zejména problémům zobrazování webových stránek v různých webových prohlížečích a tipům a trikům jak docílit toho, aby se stránky zobrazovali v prohlížečích stejně. Dále se budu věnovat nové verzi prohlížeče Internet Explorer (IE7). A na závěr předvedu, jak řeší layout webové stránky webdesignerů [www.seznam.cz](http://www.seznam.cz).

# 1 Internet

## 1.1 Historie internetu

V roce 1968 vznikla v USA pod záštitou ministerstva obrany síť ARPANET, která byla zárodkem Internetu. Tato komunikační síť měla odolat i jadernému útoku. V roce 1983 se od APANETu oddělila část, která byla zaměřena jen na vojenství a vznikla tak samostatná síť MILNET. Tato síť měla samozřejmě možnost komunikace s ARPANETem. ARPANET sice nadále financovalo ministerstvo obrany, ale oddělením vojenské části získal mnohem civilnější náplň. ARPANET nebyl jedinou počítačovou sítí, své počítačové sítě si budovaly i jiné resorty a vzhledem ke kvalitám a veřejné dostupnosti protokolů TC/IP byly tyto sítě stále častěji budovány na bázi těchto protokolů. Pro jejich uživatele a provozovatele se ukázalo jako velmi výhodné, když je mohli propojit s ARPANETem. Na ARPANET se tak začali nabalovat další a další sítě, až vznikla jedna velká síť, které se začalo říkat Internet. K ARPANETu se začali připojovat i lokální sítě, které začali vznikat po roce 1983 (v té době hlavně univerzitní sítě). Tyto sítě začali používat přenosovou technologii pro sítě LAN – Ethernet. ARPANET se dostal do role páteřní sítě, přes kterou prochází provoz mezi jednotlivými sítěmi.

## 1.2 Jak funguje Internet

Internet je tvořen desítkami tisíc trvale spuštěných počítačů zapojených do sítě. Počítače pracují buď jako server, nebo jako klient. Klienti požadují služby a servery tyto služby poskytují. Klient má speciální program – WWW klient – pomocí kterého jsou převáděny uživatelské příkazy na dotazy pro jednotlivé servery. Server po obdržení dotazu vyhledá odpověď a odešle ji klientovi.

V první fázi si WWW klient prohlédne URL, z kterého zjistí jaký způsob komunikace má použít, na který server se obrátit a jaký dokument si na něm vyžádat. Poté co naváže spojení s daným serverem, požaduje po něm zaslání dokumentu. Toho se buď dočká, nebo mu přijde odpověď v podobě chybné hlášky. Celá tato komunikace musí být samozřejmě standardizována a řídí se dle protokolu HTTP. Zodpovězením dotazu komunikace končí a v lepším případě se

tedy klient dočká požadovaného dokumentu zapsaného v jazyce HTML, který pomocí speciálních příkazů určuje, jak se dokument zobrazí. Klient tyto příkazy zanalyzuje, dokument podle příkazů zformátuje a zobrazí uživateli.

## 1.3 WWW

O skutečný masový rozvoj a popularizaci Internetu se zasloužila služba WWW (World Wide Web). WWW je jednou z částí Internetu. Je nejmladším médiem historie, ale už překonal jiná média. Na vzniku WWW se podílelo několik lidí.

### **Douglas Engelbert**

V 60. letech přišel na myšlenku provázaných dokumentů (dnes bychom řekli hyperlinků)

### **Ted Nelson**

V roce 1980 přišel s nadějným projektem s názvem Xanadu. Poprvé použil slovo hypertext (propojení více dokumentů odkazy, aktivní odkazy mohou být realizovány částmi textu, obrázky, částmi obrázků).

### **Charles Goldfarb**

V 80. letech definoval jazyk SGML (Standard Generalized Markup Language), což byl předchůdce HTML (HyperText Markup Language).

### **Tim Berners Lee**

V roce 1989 ve švýcarském výzkumném ústavu CERN v Ženevě definoval hypertextový systém (dá se říct, že se jednalo o první intranet na světě). O rok později napsal první program na tvorbu primitivních hypertextových stránek a pro systém běžící na jediném počítači navrhl název World Wide Web.

## 1.4 Úspěch WWW

Proč je dnes WWW tak úspěšný? Jedním z důvodů je, že dnes už tolik nezáleží na tom, jaké máte počítačové znalosti a zkušenosti. Uživatelské prostředí je dnes na takové úrovni, že uživatel nepotřebuje mít skoro žádné znalosti o struktuře počítače, způsobu připojení na síť a nemusí vědět jak je uskutečněna komunikace mezi jeho klientským počítačem a serverem. Uživatel jednoduše zapne počítač a spustí program, který provede vše potřebné.

Další důvodem je, že na Internetu uživatel najde nepřehledné množství informací - zpravodajství, noviny, časopisy, encyklopedie, slovníky, hudbu, video, filmy, hry a mnoho dalších.

Webu dnes patří přívlastek multimediální. Nikoho dnes již nepřekvapí, že WWW spojuje text, grafiku, animace, hudbu, video a kdoví, co se ještě objeví. Porovnáme-li stav webu dnes a před deseti lety, uvědomíme si, jak obrovský skok za tu dobu udělala technika (technologický, cenový). To, co před deseti lety mohla na svých počítačích dělat jen velká filmová, grafická studia nebo NASA, může dnes na svém počítači dělat každý.

Každý uživatel může na Internet umístit svou WWW stránku. Psaní stránek dnes již není výsadou úzké skupiny lidí. Na vytvoření WWW stránky dnes nepotřebujete znát (X)HTML, protože existují programy, které zdrojový kód stránky napíšou za vás a vy se staráte jen o vzhled stránky. (Výraz stránka je trochu zavádějící, protože po vytisknutí může jedna stránka mít i několik listů papíru) Pro WWW stránku je typické, že obsahuje odkazy na další stránky na Internetu. Odkazy vytvářejí velice složitou strukturu, protkávají stránky na celém světě tenkými nitkami odkazů. Z toho také vznikl název World Wide Web.

Internet lidi sblížuje. Každý uživatel má možnost zřídit si na Internetu e-mailovou adresu, pomocí které může komunikovat s přáteli, úřady či obchodními partnery. Posadíte se doma k počítači a můžete si psát, nebo dokonce povídat v reálném čase s kamarádem, známým, nebo úplně cizím člověkem sedícím na druhém konci světa. Na vzdálenosti vůbec nezáleží.

V posledních letech zaznamenala na Internetu velký boom reklama. Stále více firem a podniků si uvědomuje možnosti reklamy na Internetu a začíná ji využívat. Reklamu na Internetu shlédne daleko více lidí, než reklamu v televizi. S reklamou na Internetu jsou spojeny také spamy (nevyžádaná pošta). Mnoho lidí se živí tím, že rozesílají nevyžádané e-maily milionům uživatelů po celém světě.

Začíná se objevovat stále více Internetových obchodů (e-obchod), banky nabízejí stále více on-line služeb, které vám umožní používat své bankovní konto a mít o něm přehled v kteroukoli denní či noční hodinu. A přitom můžete sedět v klidu vašeho domova.

Poslední velkou výhodou WWW je nezávislost na platformě. To znamená, že tutéž stránku si můžeme prohlížet na počítači, s operačním systémem Windows, stejně jako na (Power)Macintoshi s operačním systémem firmy Apple, či na pracovní stanici, s operačním systémem Linux, Unix.

Je ale téměř jisté, že Internet ještě neřekl své poslední slovo a stále má co nabídnout.

## 2 (X)HTML

### 2.1 Co je HTML

Mluvím-li o Internetu a WWW, nemohu opominout HTML (Hypertext Markup Language – hypertextový značkovací jazyk). HTML je totiž základním stavebním kamenem všech webových stránek. Je to jednoduchý, značkový, strukturovaný jazyk. Jeho podstata spočívá v tom, že jeho příkazy neříkají co se má udělat, ale jak se to má udělat. Tím je položen základ nezávislosti na platformě, operačním systému, uživatelském rozhraní, schopnostech grafiky jednotlivých počítačů a možnostech automatizované tvorby WWW dokumentů.

### 2.2 Historie a vývoj (X)HTML

Už v 60. letech 20. století začíná pracovat Theodor Nelson na realizaci nelineárního spojování dokumentů na počítačích. Jako první přišel s pojmem hypertext.

V roce 1986 vznikl SGML (Standard Generalized Markup Language), obecný značkovací jazyk, který umožňuje definici vlastních (podřízených) značkovacích jazyků. HTML je jednou z aplikací SGML.

První definici jazyka HTML vytvořil v roce 1991 Tim Berners-Lee. Tato verze je známá pod označením HTML 0.9. Umožňovala rozčlenit text do několika různých logických úrovní, použít několik druhů zvýraznění textu a zařadit do textu odkazy a obrázky.

Požadavky uživatelů na WWW vzrůstali a tak výrobci internetových prohlížečů obohacovali HTML o nové prvky. Proto Tim Berners-Lee vytvořil návrh standardu HTML 2.0 všechny v té době používané prvky HTML. Tento standard obsahuje 2 úrovně. První z nich trochu rozšiřuje předchozí verzi HTML a druhá definuje práci s formuláři.

Další rozšíření známé jako HTML+ je obohaceno zejména o vytváření tabulek a matematických vzorců, dále obtékání obrázků textem a styly dokumentů. Z této verze HTML vznikl v roce 1995 návrh standardu HTML 3.0, který vytvořil Dave Raggett.

Na počátku roku 1996 bylo zřejmé, že HTML 3.0 bylo tak mohutným skokem vpřed, že se nenašel nikdo, kdo by implementoval prohlížeč s jeho podporou. V té době již bylo založeno konsorcium W3C (1994 World Wide Web Consortium), které se snaží koordinovat vývoj standardů Webu dodnes. Členové W3C se rozhodli rozšířit HTML 2.0 a vytvořili tak HTML 3.2. Tato verze neobsahuje vše z HTML 3.0. Z verze 3.0 zůstali vlastně jen tabulky, zlepšila se podpora formátování písma, podpora Java-pletů. Tato verze je doporučením konsorcia W3C od ledna 1997. Znamená to, že by ji měli používat všichni, aby byla zajištěna kompatibilita. V té době také W3C doporučuje používat CSS (Cascading Style Sheets - kaskádové styly), v té době standard CSS 1.

V létě 1997 zveřejnilo W3C návrh HTML 4.0, které navazuje na HTML 3.2 a je zpětně kompatibilní. Tento návrh se snaží dosáhnout původního záměru, kde je HTML jazyk pro vyznačování významu jednotlivých částí dokumentu. Veškeré příkazy ovlivňující vzhled webu by měli být definovány pomocí kaskádových stylů. Do této specifikace byly přidány rámy a plovoucí rámy, byly vylepšeny formuláře a tabulky, podpora skriptů a nové řádkové elementy.

Aby měli výrobci internetových prohlížečů čas na přechod k této normě, rozdělilo W3C HTML 4.0 na HTML 4.0 Strict, HTML 4.0 Transitional (přechodná norma, zachovává vzhledové atributy HTML 3.2 a zároveň obsahuje všechny výhody a vylepšení HTML 4.0 Strict, především v oblasti přístupnosti a CSS) a HTML 4.0 Frameset (téměř se shoduje s HTML 4.0 Transitional, ale je ještě rozšířena o elementy pro definici rámu).

Na konci roku 1999 W3C zveřejnilo standard HTML 4.01, ve kterém byly revidovány některé chyby HTML 4.0. Tato verze obsahuje kompletní popis jazyka a všechny následující specifikace jazyků pro tvorbu webu se na ni odkazují.

W3C se snažilo prosazovat jazyk XML (eXtensible Markup Language – rozšiřitelný značkovací jazyk) jako jediný a hlavní značkovací jazyk nejen pro web. Logickým vyústěním této snahy je XHTML 1.0 (eXtensible HyperText Markup Language), který W3C vydalo na začátku roku 2000. Tato specifikace je stejná jako HTML 4.01, ale jsou v ní integrována pravidla XML, který vychází z SGML, ale je zbaven některých zbytečných vlastností a některé nové přidává.

Po vydání XHTML 1.0 se ho W3C snaží prosazovat jako jediný jazyk pro definici webových stránek, bez ohledu na typ zařízení, které je zpracovává. Rozvoj a rozšíření webu na různá alternativní zařízení (např. mobilní telefony, PDA) ukázal, že tato zařízení kvůli svým omezeným možnostem podporují jen některé vlastnosti XHTML. Proto bylo potřeba definovat a standardizovat, které vlastnosti jsou tato zařízení schopná podporovat, a proto vznikla specifikace Modularization of XHTML. Ta rozděluje všechny prvky a vlastnosti XHTML 1.0

do modulů, ze kterých se následně skládají nové značkovací jazyky. Vývojáři zabývající se tvorbou webu, pro alternativní prohlížečí zařízení mohou definovat nové moduly s prvky specifickými pro dané zařízení, mohou stávající moduly modifikovat, ale hlavně z nich mohou skládat nové značkovací jazyky, které vyhovují potřebám a možnostem interpretace webu na těchto zařízeních. Z nově vytvořených jazyků zatím W3C uznalo jen XHTML 1.1 a XHTML Basic. XHTML 1.1 se až na několik detailů shoduje s XHTML 1.0 Strict. Hlavní rozdíl je v tom, že XHTML 1.1 je definováno pomocí modulů. XHTML Basic se skládá pouze ze základních modulů a je používán hlavně na mobilních telefonech, PDA a podobných zařízeních.

## 2.3 Základy (X)HTML

Webové stránky neboli HTML dokumenty jsou vlastně obyčejné textové soubory, které obsahují text a speciální značky (tagy). Zdrojový kód každé WWW stránky by měl obsahovat několik základních tagů. A to tagy hlavičky a těla. Tím je zaručeno to, že všechny prohlížeče pochopí, o co v dokumentu jde.

Příklad základní struktury HTML stránky:

```
<!DOCTYPE HTML PUBLIC „-//W3C//DTD HTML 4.01 //EN“>
<html>
<head>
<title>název stránky</title>
</head>
<body>
Obsah HTML dokumentu
</body>
</html>
```

!DOCTYPE specifikace DTD (Document Type Definition). Píše se na začátek souboru před tag <html>. Není nutné ho psát, ale zaručí nám to, že stránka půjde zpracovat různými nástroji vyvinutými pro práci s dokumenty na bázi SGML. Doctype určuje použitou verzi HTML.

**<html>** začíná dokument  
**</html>** končí dokument  
**<head>** začíná hlavičku, která obsahuje důležité informace  
**</head>** končí hlavičku  
**<title> </title>** vymezuje jméno stránky  
**<body></body>** neboli tělo dokumentu, vše co se nachází mezi těmito tagy se zobrazí

Výše vidíte, že se tagy vyskytují ve dvojicích `<tag> </tag>`, kde první z nich něco začíná a druhý něco končí. Lomítko prostě určuje, že jde o uzavírací tag. Jde o takzvané párové tagy. Kromě párových tagů se objevují tagy nepárové, které nemají uzavírací tag. Nejsou tak časté. Například tag `<br>` sloužící k odřádkování. XHTML ale vyžaduje uzavírání tagů a tak by se při použití XHTML zapsal tento tag takto `<br />`.

Příklad formátování HTML dokumentu:

```
<html>
<head>
<title>název stránky</title>
</head>
<body bgcolor="yellow" text="blue">
<h1>Nadpis první úrovně</h1>
<p>odstavec s <b>tučným textem</b>, <i>kurzivou</i>, <font color="blue">modrým
textem</font> a <font color="blue"><b><i>kombinací.</i></b></font></p>
<h2>Nadpis druhé úrovně</h2>
<p>text v odstavci, <br>
který je zalomen. </p>
Odkaz na stránku vyhledávače <a href=http://www.google.com>google</a>
</body>
</html>
```

**bgcolor** je atribut tagu body, který určuje barvu pozadí pro celý dokument

**text** je atribut tagu body, který určuje základní barvu písma pro celý dokument

**<h1></h1>** tagy používané pro nadpisy, html nám dává k dispozici 6 úrovní nadpisů, kdy h1 je nejdůležitější a h6 je nejméně důležitý.

**<p></p>** vymezuje odstavec, je jedním z nejpoužívanějších tagů. Po jeho uzavření prohlížeč automaticky zalomí řádek a udělá vertikální mezeru.

**<b></b>** formátuje text na tučný.

*<i></i>* formátuje text nakurzívu.

**<font color="blue"></font>** formátuje barvu textu. Tag font má atribut color, který má hodnotu blue (modrá). Tag font může mít mnoho dalších atributů.

**<br>** slouží k zalomení řádku. Následující text bude začínat na novém řádku. Tento tag je v HTML nepárový, takže nemá uzavírací tag `</br>`, ale jak jsem psal výše při použití XHTML se zapisuje `<br />`

**<a href="url adresa">odkaz</a>** text (nebo obrázek) mezi těmito dvěma tagy se zobrazí jako odkaz. Po kliknutí na odkaz se prohlížeč přepne na url (uniform resource locator) adresu, která je zadaná pomocí atributu href.

Několika málo zásad pro HTML (převzato a upraveno z [www.jakpsatweb.cz](http://www.jakpsatweb.cz)):

- nezáleží na velikosti písmen, `<body>` je totéž jako `<BODY>`
- v adresách a jménech souborů záleží na velikosti písmen, nesmějí tam být mezery a české znaky (háčky,čárky)
- tagy, které prohlížeč nezná, jako by nebyly
- na začátku tagu nesmí být mezera, například `< br>` je špatně
- dvě mezery po sobě (nebo víc) mají stejný význam jako jedna mezera
- konec řádku ve zdroji se chápe jako mezera
- jména atributů je dobré dávat vždy do uvozovek, ale není to zcela nutné, pokud uvnitř nejsou mezery
- speciální znaky typu © se do zdroje zadávají jako posloupnost znaků `&něco;`, například `&copy;`, pevná mezera je `&nbsp;`;
- poznámka se do zdroje vkládá mezi značky `<!-- a -->` `<!-- Toto je poznámka, která se nezobrazí -->`

Samozřejmě, že tagů a jejich atributů je mnohem víc, ale cílem této práce není jejich popis a pro naše účely tyto základy postačují. Již víme, jak by měl vypadat zdrojový kód stránky a co by měl, nebo neměl obsahovat. Víme, jakým způsobem lze formátovat vzhled HTML dokumentu pomocí různých tagů a atributů. Ovšem HTML není určeno primárně k formátování vzhledu HTML dokumentů, pro tento úkol byly vytvořeny kaskádové styly (CSS).

## 3 Kaskádové styly (CSS)

### 3.1 Historie a vývoj CSS

Již od roku 1990 byla snaha o oddělení struktury od layoutu (formátování) dokumentu. Tim Berners-Lee navrhl svůj prohlížeč NeXT tak, že formátování dokumentu bylo řízeno jednoduchým zabudovaným stylovým předpisem. Bohužel však tento předpis nikdy nepublikoval a spoléhal na autory dalších prohlížečů, že tuto koncepci dále rozvinou. To se však nestalo. Prohlížeč NSCA Mozaic z roku 1993 dokonce omezil možnosti uživatelského či autorského stylování HTML dokumentů na minimum.

Postupem času, jak popularita Internetu stoupala, si stále více autorů webových stránek stěžovalo, že nemohou vzhled HTML dokumentů nijak ovlivnit. To co v té době bylo samozřejmostí nejen u DTP programů, ale i u textových editorů (změna velikosti, barvy písma, barvy pozadí a další) v HTML dokumentech nebylo možné. Na tyto stížnosti přišly skoro současně dvě reakce.

Hakon Wium Lie v listopadu 1994, těsně před konferencí o WWW v Chicagu, publikoval první koncept Cascading HTML Style Sheets. I když nebyl návrh zdaleka ještě dokonalý, stal se základem, ze kterého vycházejí kaskádové styly dodnes.

Jen o tři dny později se objevila první verze prohlížeče Netscape, která na všeobecnou potřebu formátovat HTML dokumenty reagovala po svém. Místo implementace stylového jazyka totiž Netscape zabudoval základní formátovací prostředky přímo do jazyka HTML. Jak se později ukázalo, pro vývoj webu to znamenalo velký problém, neboť střet těchto dvou koncepcí zásadně poznamenal kompatibilitu a použitelnost webových dokumentů na mnoho let dopředu a jeho negativní důsledky neseme dodnes.

O kaskádových stylech se ještě dlouho diskutovalo, přičemž hlavním tématem diskuse nebyl ani tak vlastní jazyk, jako spíš koncepce, která poslední slovo o formátování HTML dokumentu přenechá uživateli, nikoli autorovi. Zastánci této myšlenky nakonec zvítězili a v prosinci 1996 byla vydána Specifikace kaskádových stylů 1 (CSS 1), již jako doporučení W3C, které vzalo kaskádové styly pod svá křídla jako nedílnou součást rozvoje HTML.

W3C poté kaskádové styly dále rozvíjelo a rozvíjí dodnes. V květnu 1998 byla vydána Specifikace CSS 2. A v únoru 2004 byla vydána Specifikace CSS 2.1, která opravuje některé

chyby v CSS 2 a přidává některé nové vlastnosti. V současné době se pracuje na Specifikaci CSS3.

Svůj vývoj prodělali i implementace kaskádových stylů v prohlížečích. První komerční prohlížeč, který částečně podporoval CSS 1 byl Internet Explorer 3 (IE3) vydaný v srpnu 1996 společností Microsoft. V té době však Specifikace CSS 1 nedospěla do stadia oficiálního doporučení W3C, a tak se implementace v IE3 se specifikací zcela neshoduje.

Dalším prohlížečem podporující CSS 1 byl Netscape Navigator 4.0 (NN4). Protože však byla společnost Netscape od počátku ke kaskádovým stylům značně skeptická, je implementace v NN4 spíše důsledkem konkurenčního boje než pečlivé přípravy. Podpora je tak sice velmi rozsáhlá, ale také velmi chybová a některé z podporovaných vlastností jsou tak nepoužitelné. Netscape navíc implementuje kaskádové styly prostřednictvím JavaScriptu, takže pokud je vypnut v NN4 JavaScript, nejsou aktivní ani kaskádové styly.

Poté se vývoj Netscapu na dlouhá léta zastavil a nové implementace CSS byly záležitostí Microsoftu a jeho prohlížeče IE a od roku 1998 také prohlížeče Opera. Netscape byl oživen až poté, co byly otevřeny zdrojové kódy a převzetím vývoje komunitou nezávislých vývojářů. Vzniklo tak vykreslovací jádro Gecko rozvíjené do současnosti jako otevřený kód (open source), které převzala společnost Netscape a zabudovala do svých nejnovějších prohlížečů verzí NN6 a NN7.

Přes dlouhou dobu od vydání Specifikace CSS 2 se bohužel zatím žádnému výrobcu nepodařilo dotáhnout implementaci CSS 2 do zdárného konce. Posledních pár let platí, že vždy ta nejaktuálnější verze z trojice hlavních prohlížečů (Microsoft Internet Explorer, otevřený kód Gecko/Mozilla, Opera) je s implementací nejdál.

## **3.2 Význam a použití CSS**

CSS neboli Cascading Style Sheets, česky kaskádové styly, vznikly kolem roku 1997. Tvoří je soubor metod pro grafickou úpravu webových stránek. Kaskádové se jim říká proto, že se na sebe mohou vrstvit.

Poté, co se Internet začal rozvíjet, začaly vznikat požadavky na lepší grafický vzhled webových stránek. Tyto požadavky byly částečně uspokojeny přidáním nových tagů a atributů (například tag font, atribut align). Kromě toho, začaly prohlížeče nabízet nové proprietární rozšíření HTML, která umožnila lepší kontrolu nad výsledným vzhledem stránky.

Bohužel použití těchto nových možností formátování, mělo neblahé účinky. V HTML dokumentech se místo struktury začal vyznačovat spíše grafický vzhled. Další nevýhoda spočívala ve velké pracnosti, kdy způsob formátování se musel nastavit pro každý element dokumentu zvlášť. Tyto nevýhody byly odstraněny díky kaskádovým stylům, pomocí nichž lze jednoduše definovat druh, barvu, velikost písma, způsob zarovnání a mnoho dalších vlastností. V HTML dokumentu se pak můžeme zaměřit už jen na strukturu. V jednom dokumentu může být použito více stylů, které se vzájemně doplňují. A jeden styl může být použit pro více dokumentů, které tak získají jednotný vzhled.

CSS představuje alternativu k formátovacím prostředkům v HTML. Co CSS přináší:

- **širší možnosti formátování dokumentu** – CSS přináší daleko širší formátovací možnosti než HTML
- **snadná tvorba a údržba konzistentního stylu** – všechny stránky jednoho webu by měly mít jednotný styl (nadpisy, odstavce, odkazy, barvy, atd.). V HTML se toho dosahuje velice těžko. Ještě těžší je provést celkovou stylovou změnu webu, který nevyužívá CSS. Znamená to změnit velké množství atributů (barva, rozměry, atd.). Ale s dobře navrženým stylovým předpisem jsou tyto změny otázkou několika minut
- **oddělení struktury obsahu a stylu** – je to jeden z nejvýznamnějších přínosů CSS. HTML slouží hlavně k popisu struktury hypertextového obsahu. Lze jím označit strukturální význam jednotlivých částí obsahu (např. nadpisy, odstavce textu, seznamy, atd.). CSS slouží k nastýlování obsahu dokumentu
- **lepší přístupnost dokumentů** – pokud správně uplatníme oddělení struktury obsahu a stylu, získáme tím i lepší přístupnost webových dokumentů. Dochází k tomu v důsledku zjednodušení zdrojového kódu dokumentu. Ten je poté přístupnější hendikepovaným osobám a méně běžným zařízením. CSS počítá s různými typy zobrazovacích médií a výrazně tak usnadňuje tvorbu alternativních verzí dokumentu
- **dynamická práce se styly** – CSS jsou výborným prostředkem k tvorbě dynamických efektů na webových stránkách (nejznámější je změna stylu odkazu)
- **formátování XML dokumentů** – XML dokumenty se v moderních prohlížečích zobrazují bez jakéhokoli formátování, a proto je CSS jednoduchou cestou jak XML dokumenty zobrazit srozumitelným a přehledným způsobem

Kaskádové styly můžeme použít různým způsobem a v různém rozsahu. V praxi se nejčastěji uplatňují tyto 3 možnosti:

- **obohacení formátovacích možností** – zřejmě nejrozšířenější možnost. Layout stránky je většinou tvořen tabulkami, jejichž vlastnosti (barva písma, pozadí, ohraničení, atd.) jsou určeny jejich atributy. Často se používají zastaralé značky a atributy a také proprietární prvky zavedené výrobcí jednotlivých prohlížečů, které nejsou ve standardech nijak definovány. CSS zde slouží jen jako doplněk k dosažení efektů, které pomocí HTML nelze vytvořit.
- **veškeré formátování s výjimkou rozmístění (layout) prvků** – layout prvků na stránce je proveden pomocí tabulky, ale veškeré formátování je provedeno pomocí CSS.
- **formátování jen pomocí CSS, včetně rozmístění prvků** – jedná se o nejpokročilejší možnost při využití CSS. Rozvržení stránky, včetně polohování jednotlivých prvků a formátování stránky provádí CSS. Jedná se o takzvaný beztabulkový layout, protože se obejde bez tabulek (použitých pro layout).

Vždy záleží na konkrétním projektu a úsudku webdesignéra, jakou strategii zvolit. Každý může zvolit různé kombinace podle vlastních potřeb.

### 3.3 Metody připojení CSS k (X)HTML

Aby byl (X)HTML dokument zformátován prostřednictvím CSS, musí se k němu CSS připojit. Existuje několik metod připojení CSS k (X)HTML dokumentu s různými výhodami a nevýhodami a různou podporou v klientech. To umožňuje definovat stylové předpisy pro různé prohlížeče.

Tři základní metody připojení stylových předpisů k (X)HTML:

- pomocí prvku <link>
- pomocí prvku <style>
- pomocí atributu style různých html prvků

### 3.3.1 Pomocí prvku <link>

Myšlenka CSS je taková, že se definice stylů umisťují do externího souboru, na který se pak v dokumentu odkazují. Docílíme toho pomocí prvku <link>. Jedná se o nejpreferovanější způsob.

Externí stylový soubor mohou vytvořit v poznámkovém ve speciálním programu na tvorbu CSS, poznámkovém bloku nebo jiném textovém editoru. Vytvořenému souboru uložíme s příponou css a pak už se stačí na soubor v dokumentu odkázat.

Prvek <link > se vkládá do hlavičky dokumentu a má tyto atributy:

**href** – určuje cestu k externímu souboru se stylovým předpisem

**type** – určuje typ obsahu odkazovaného souboru, v případě CSS je tato hodnota text/css

**rel** – určuje typ vazby dokumentu k externímu zdroji. Hodnota stylesheet pro základní stylové předpisy, alternate stylesheet pro alternativní stylové předpisy

**media** – určuje média, pro které je stylový předpis určen. Není povinný, jeho hodnotou je řetězec obsahující jeden nebo více deskriptorů média

**title** – přiřazuje textový titulek danému stylovému předpisu. Praktický význam má jen u alternativních stylových předpisů.

V hlavičce dokumentu může být definováno více prvků <link> a připojeno tak více stylových předpisů. Příklady použití prvku <link>:

```
<link rel="stylesheet" type="text/css" href="styl1.css">
```

K dokumentu se připojí stylový předpis ze souboru styl1.css. Použije se pro libovolné médium.

```
<link rel="stylesheet" type="text/css" href="styl2.css" media="all">
```

K dokumentu se připojí stylový předpis ze souboru styl2.css. Použije se pro všechny typy médií.

```
<link rel="stylesheet" type="text/css" href="styl3.css" media="screen">
```

K dokumentu se připojí stylový předpis ze souboru styl3.css. Použije se pro médium screen.

```
<link rel="stylesheet" type="text/css" href="styl4.css" media="screen, print">
```

K dokumentu se připojí stylový předpis ze souboru styl4.css. Použije se pro média screen a print.

**<link rel="alternate stylesheet" type="text/css" href="styl5.css" title="Alternativní styl">**

K dokumentu se připojí alternativní stylový předpis ze souboru styl5.css. Klientovi by měl být nabídnut například prostřednictvím menu, ve kterém se zobrazí obsah atributu title. Tento styl se spustí jen v případě, že si ho uživatel zvolí.

### **Výhody:**

Pro všechny webové stránky jednoho webového místa stačí jeden externí soubor se stylovým předpisem, což usnadní správu webu a dosažení jednotného grafického vzhledu celého webu.

Při prvním načtení se stylový předpis cachuje na straně klienta a načítání dalších dokumentů ze stejného webového místa se příště urychlí.

Touto metodou je dosaženo idey oddělení formátování a struktury dokumentu, což má vliv na přenositelnost a správu webového obsahu.

### **Nevýhody:**

Metoda připojení externího souboru se styly je nevhodná v případě, že chceme dokument zachovat jako celek (např. pro odeslání elektronickou poštou – musíme s html souborem odeslat i soubor se styly, ale pozor na nastavení cesty k tomuto souboru), nebo pokud se dokument včetně některých částí stylového předpisu generuje dynamicky na straně serveru.

## **3.3.2 Pomocí prvku <style>**

Stylový předpis se může zapsat přímo do hlavičky dokumentu, pomocí prvku <style>. Tato metoda připojení stylového předpisu je považována za méně praktickou. Prvek <style> má tyto atributy“

**type** – určuje typ obsahu odkazovaného souboru, v případě CSS je tato hodnota text/css

**media** – určuje média, pro které je stylový předpis určen. Není povinný, jeho hodnotou je řetězec obsahující jeden nebo více deskriptorů média

**title** – přiřazuje textový titulek danému stylovému předpisu. Praktický význam má jen u alternativních stylových předpisů.

Příklad použití prvku <style>:

```
<style type="text/css" media="all">
body{
  background-color: #ffffff;
  font-family: Verdana, Geneva, Arial, Helvetica, sans-serif;
  font-size: 12px;
  color: #000000;
}
table{
  width: 400px;
  border-color: red;
  border-style: solid;
  border-width: 2px;
}
</style>
```

### **Výhody:**

Prvek <style> umožňuje ovšem načítání stylových předpisů z externích souborů pomocí pravidla @import a tím se tak zbavit jeho nevýhod.

Příklad použití:

```
<style type="text/css">
  @import "styl.css";
</style>
```

Tento způsob se používá k ukrytí stylových předpisů před prohlížeči, které pravidlo @import, nebo jednu z jeho syntaxí nepodporují.

### **Nevýhody:**

Stylový předpis zvětšuje objem dokumentu a necacheje se, takže se s každou novou stránkou načítá znovu.

Obtížnější spravování a údržba dokumentů. Pokud chci změnit vzhled dokumentů, musím provést úpravy ve všech dokumentech.

### 3.3.3 Pomocí atributu style různých HTML prvků

Styl lze definovat ve zdroji přímo u stylovaného elementu prostřednictvím atributu style. Tato metoda se jmenuje vřazené stylové předpisy (inline style sheets). Považuje se za nejméně výhodnou.

Příklad použití:

```
<body style="background-color: #000000; font-family: Verdana, Geneva, Arial, Helvetica,
sans-serif; color: #ffffff >
<p style="font-size: 12px; font-style: italic>
Tato stránka bude mít černé pozadí, tento odstavec bude napsán bílým písmem o velikosti
12px, kurzívou.</p>
</body>
```

Zapisovat styl přímo do zdrojového kódu dokumentu není vhodné a v novějších verzích XHTML (verze 1.1) nepřípustné. Jedinou výhodou je snadné testování vzhledu jednotlivých prvků během tvorby dokumentu.

## 3.4 Struktura stylového předpisu

Práce s kaskádovými styly není moc složitá, stylový předpis se skládá z posloupnosti jedné nebo více definic. Definice je buď sadou pravidel nebo at-pravidlem.

Styly v sadě pravidel se skládají ze dvou částí, selektoru a bloku deklarací. Nejdříve se zapíše selektor a poté do složených závorek blok deklarací.

At-pravidlo začíná znakem @, za kterým následuje klíčové slovo. At-pravidlo zahrnuje vše, co následuje až po další středník, nebo začátek dalšího bloku.

Do stylového předpisu se mohou vkládat i komentáře, které se uzavírají zleva /\* a zprava \*/.

### 3.4.1 Deklarace

Deklarace se skládá z vlastnosti a hodnoty, které jsou odděleny dvojtečkou. Jednotlivé deklarace se od sebe oddělují středníkem.

Příklad:

```
p{font-family: verdana}
p{font-size: 12px}
p{line-height: 1.6em}
```

Deklarace pro stejný selektor mohou být sdružena, takže zápis lze zjednodušit takto:

```
p{ font-family: verdana
   font-size: 12px;
   line-height: 1.6em}
```

#### 3.4.1.1 Hodnoty

Existuje několik typů hodnot:

- číslo
- velikost
- procento
- URI
- barva
- úhel
- čas
- frekvence

##### Číslo

Používají se celá nebo reálná čísla. Pro oddělování desetinného čísla se používá tečka, je možné začít zápis i tečkou (0.5=.5). Čísla mohou obsahovat znaménko plus nebo mínus. Mezi znaménkem a číslem nesmí být žádná bezvýznamná mezera. Mnoho vlastností má definovaný obor hodnot (rozsah použitelných čísel) pouze na několik určitých čísel, uvedených v referenčním popisu dané vlastnosti.

## **Velikost**

Velikosti se skládají ze znaménka (plus nebo mínus), čísla (celého nebo reálného) a jednotky. Nesmí mezi nimi být žádný bezvýznamný znak. Jestliže je hodnota 0, není třeba psát jednotku. Používají se jednotky relativní nebo absolutní.

Relativní jednotky určují velikost relativně k nějaké jiné velikosti

- em – velikost aktuálního písma
- ex – střední výška aktuálního písma ( $1ex=0.5em$ )
- px – pixely

Absolutní jednotky se používají, když známe fyzikální vlastnosti výstupního média

- in – palce ( $1in=2.54cm$ )
- cm - centimetry
- mm - milimetry
- pt – typografické body ( $1pt=1/72in$ ), používají se ve stylových předpisech určených pro tiskárnu
- pc – pica ( $1pc=12pt$ )

## **Procento**

Procenta se skládají ze znaménka (plus nebo mínus), čísla (celé nebo reálné) a znaku %. Nesmí mezi nimi být žádný bezvýznamný znak. Procenta se vztahují k nějaké jiné hodnotě. Mohou se vztahovat k hodnotě jiné vlastnosti stejného elementu, k vlastnosti rodičovského elementu, nebo šířce a výšce okna.

## **URI (Uniform Resource Identifier)**

URI identifikuje externí zdroj. Formát zápisu je řetězec `url(uri/soubor.pripona)`. Obsah zápornky může být uzavřen do uvozovek, nebo apostrofů.

## **Barva**

Barvy mohou být vyjádřeny buď jménem nebo pomocí RGB. Je definováno 16 základních barev, které se mohou vyjádřit jménem. RGB lze vyjádřit čtyřmi způsoby. Hexadecimálním zápisem, kdy zápis začíná znakem #, za kterým následuje trojčiferné nebo šesticiferné hexadecimální číslo. Hodnoty RGB lze zapsat také procentuálně nebo dekadicky.

## Úhel

Úhly se používají pro aurální média (zvukový výstup). Skládají se ze znaménka (plus nebo mínus), čísla a úhlové jednotky. Nesmí mezi nimi být žádný bezvýznamný znak. Používané úhlové jednotky:

- deg – stupně
- grad – grady
- rad – radiány

## Čas

Používá se pro aurální média (zvukový výstup). Skládá se z čísla a časové jednotky. Nesmí mezi nimi být žádný bezvýznamný znak. Používané časové jednotky:

- ms – milisekundy
- s – sekundy

## Frekvence

Používá se pro aurální média (zvukový výstup). Skládá se z čísla a jednotky frekvence. Nesmí mezi nimi být žádný bezvýznamný znak. Používané jednotky:

- Hz – Hertz
- kHz – KiloHertz

### 3.4.2 Selektory

Selektor je skupina jednoho nebo více jednoduchých selektorů oddělených kombinátory. Kombinátory jsou bezvýznamné znaky (např. mezera), znak „>“ nebo znak „+“.

#### 3.4.2.1 Typový selektor (selektor HTML značky)

Typový selektor se váže k HTML značce (tagu), jsou podporovány všemi prohlížeči, které podporují CSS1.

Příklad: `p{color: blue}` – barva textu u všech prvků `<p>` bude modrá.

### 3.4.2.2 Univerzální selektor

Zapíše se znakem \* a páruje se se všemi prvky. Takovýto styl se bude aplikovat na všechny prvky, u nichž je to možné.

Příklad:

`*{color:blue}` znamená, že všechny texty na stránce budou mít modrou barvu. Nezáleží, na tom, kde se nacházejí, jestli jsou uvnitř `<p>`, `<div>`, nebo uvnitř tabulky `<table>`.

### 3.4.2.3 Selektor třídy

Třída (class) se tvoří ve stylovém předpisu pomocí tečky, která je následována jménem třídy. V (X)HTML se takovýto styl zapisuje `class="jméno_třídy"`. Element se stejnou třídou se může na stránce vyskytovat vícekrát.

Příklad:

Ve stylovém předpisu - `.modra{color:blue}`

V (X)HTML – `<p class="modra">text v odstavci</p>`

Výsledek – v odstavci s atributem `class="modra"` bude text napsán modře. Tato třída se může použít pro více odstavců, ale i pro další tagy (např. `div`, `table`, atd.).

Mohu ovšem tento styl definovat jen pro `<p>` - `p.modra{color:blue}` styl se použije pro všechny odstavce, které mají uveden atribut `class="modra"`.

Je možné použít i tzv. vícenásobné třídy, kdy se každá jednotlivá třída v selektoru oděluje tečkou. Nezáleží na pořadí tříd.

Příklad:

`p.prvni.druha{color:blue}` – tento styl se použije u všech prvků `<p>`, které budou obsahovat atribut `class="prvni druha"`.

`<p class="prvni druha">Tento odstavec je napsán modře.</p>`

`<p class="prvni treti">Tento odstavec není napsán modře</p>`

### 3.4.2.4 Selektor ID

Selektor id je velice podobný selektoru třídy, ale na rozdíl od ní nezačíná tečkou, ale znakem #, který je následován identifikátorem. Používá se pro jednoznačný popis nějakého elementu. V těle dokumentu by se element s jedním identifikátorem měl vyskytovat pouze jednou.

Příklad:

Ve stylovém předpisu - #telo{ width: 770px }

V (X)HTML - <div id="telo"> obsah </div>

Výsledek – blok s id="telo" bude mít šířku 770px.

## 4 Praktická část

### 4.1 Box model

Box model je velice důležitá součást webových stránek. většina nově vytvářených stránek je tvořena pomocí box modelu. Každý box se skládá z několika částí. Obsah > vnitřní okraj > rámeček > vnější okraj. Okraje se dále dělí na horní, pravý, dolní a levý. Rozměry boxu jsou závislé na několika faktorech, například, jestli jsou boxu nastaveny vlastnosti width a height, box obsahuje další boxy nebo text atd.

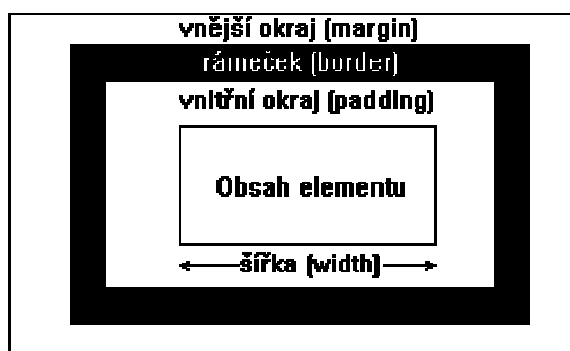
Pokud boxu nastavíme vlastnost width, specifikace CSS považuje za šířku boxu šířku jeho obsahu. Rozměry vnitřního okraje, rámečku a vnějšího okraje se k ní dodatečně přičítají. Pokud například prvku DIV určíme, že má být široký 100px, má mít levý i pravý vnitřní okraj 10px a rámeček 5px:

```
div{ width: 100px;
padding: 10px;
border: 1px solid red }
```

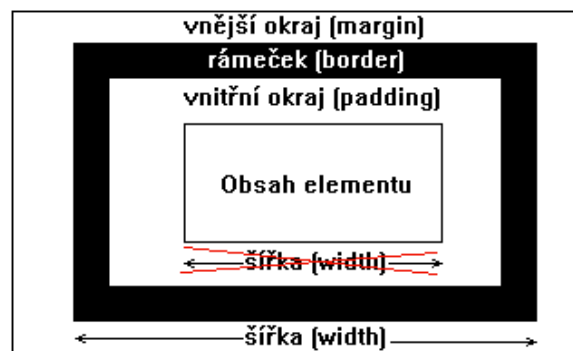
Jeho celková šířka bude  $100 + 10 + 10 + 5 + 5 = 130\text{px}$

Bohužel ne všechny prohlížeče se řídí specifikací CSS. V Microsoftu specifikaci buď špatně pochopili, nebo se rozhodli jít svou vlastní cestou a tak prohlížeč Internet Explorer počítá šířku jinak. IE počítá jinak i výšku, ale její vliv na vzhled webové stránky není tak vidět. IE chápe hodnotu width jako rozměr celého boxu, tzn., že při stejném zadání bude šířka 100px včetně šířky obsahu, okrajů i rámečku.

Šířka obsahu pak bude  $100 - 10 - 10 - 5 - 5 = 70\text{px}$



Obrázek 1 - šířka boxu podle standardů



Obrázek 2 - šířka boxu podle IE

Názor na to, který z postupů je lepší se různí, ale ve specifikaci CSS je jasně dáno, že hodnota width udává jen šířku obsahu a okraje a rámeček se k ní přičítají. Jiný přístup je chybný.

I laikovi musí být jasné, že tyto dva rozdílné výpočty šířky boxu značně znesnadňují webdesignérům tvorbu webových stránek, obzvláště v dnešní době, kdy je koncepce boxů nejpoužívanější způsob pro rozvržení layoutu webových stránek.

Chybný postup používají starší prohlížeče IE verze 5 a 5.5. Oba dva jsou sice již starší, ale používá je stále velké množství uživatelů. Správný postup výpočtu používají všechny nové prohlížeče vyrobené v posledních letech. Výrobci prohlížečů v posledních letech zapracovali do nejnovějších verzí svých prohlížečů 3 režimy zpracování:

- Standardní režim (STD) – prohlížeč se snaží co nejvíc dodržovat specifikace CSS a (X)HTML. Netolerují se zde chyby a výpočet šířky boxu se provádí podle specifikace CSS.
- Quirk režim - nestandardní režim, který simuluje chybné chování a postupy starých verzí IE (verze 5 a 5.5) a Netscape. Formátování stránky je méně předvídatelné. IE 6 v tomto režimu zobrazí stránku přibližně stejně jako IE 5.
- Převážně standardní režim – nedodržuje zcela standardy a přejímá některé vlastnosti ze starých prohlížečů. IE tento režim nemá.

Z důvodů zpětné kompatibility jsou prohlížeče schopny zpracovat tyto 3 modely (kromě IE, který nezná převážně standardní režim). Jak prohlížeč pozná, který režim má aplikovat? Prohlížeč se řídí použitým DTD (Dokument Type Declaration) v prvku DOCTYPE.

Problém s různým výpočtem šířky boxu se dá řešit několika způsoby. Používají se:

- různé verze stránek pro různé prohlížeče
- vytváření stylových předpisů pro jednotlivé prohlížeče
- hacky (různé konstrukce v CSS, které některé prohlížeče nepochopí a nepoužijí a jiné ano)
- matrjoška hack – obsah se zapouzdří do prvku DIV, kterému se nastaví jen šířka a nulové okraje a rámeček. Obsahu necháme automatickou šířku a nastavíme potřebné okraje a rámeček. V tomto případě se zachovají všechny prohlížeče stejně, protože když jsou okraje a rámeček nulové, je jedno, jestli se přičítají nebo odečítají. Protože obsah má nastavenou automatickou šířku, roztáhne se na šířku vymezenou DIVem.

Vytváření různých verzí stránek pro různé prohlížeče je dle mého názoru zbytečně pracné, časově více náročné a výsledné stránky zabírají na serveru zbytečně moc prostoru. A webdesigner nemá jistotu, že stránky budou vypadat všechny stejně, většinou se trochu liší. Dnes se prakticky nepoužívá.

Vytváření stylových předpisů pro jednotlivé prohlížeče se používá častěji, ale také není zcela ideální, protože je velice obtížné docílit stejného vzhledu v různých prohlížečích.

Hacky jsou v dnešní době velice používané. V jednom stylovém předpisu jsou styly pro všechny prohlížeče a výsledek se zobrazuje ve všech prohlížečích prakticky stejně.

Matrjoška hack je dnes také velmi rozšířen. Ve všech prohlížečích se zobrazuje stejně.

Osobně se přikláním k hackům a matrjoška hacku, protože jsou pro webdesignéra pohodlnější, zobrazení stránek v různých prohlížečích stejné.

## 4.2 CSS hack

Jak již bylo řečeno, různé prohlížeče pochopí některé CSS vlastnosti jinak než ostatní a zobrazují webové stránky různým způsobem. CSS hacky jsou různé triky, pomocí nichž můžeme docílit toho, že webové stránky se zobrazují v různých webových prohlížečích skoro stejně. Nebo naopak, můžeme docílit i toho, že se v každém prohlížeči zobrazí naprosto rozdílně.

### 4.2.1 Podtržítkový hack

Jde o jeden z nejznámějších hacků, který se u nás v současné době používá. Podtržítkový hack funguje pro WinIE (5, 5.5, 6). Zapisuje se tak, že před vlastnost prvku se napíše ještě podtržítko. Všechny prohlížeče kromě IE tuto vlastnost ignorují a IE ignoruje podtržítko na začátku vlastnosti a zpracuje vlastnost bez podtržítka.

U tohoto hacku dochází k zajímavé situaci, protože podtržítko je ve specifikaci CSS2.1 v identifikátorech povoleno, takže se zdá, že se jedná o validní zápis, který ignorují všechny prohlížeče kromě IE.

Tato „chyba“ umožňuje nastavit v CSS vlastnosti jen pro IE (MacIE tuto vlastnost nemá). Dá se tím například vyřešit problém box modelu.

```
.test{ width: 500px; padding: 10px; border-width: 1px; border-style: solid; _width: 522px }
```

Kdy prohlížeče kromě IE nastaví šířku obsahu boxu 500px, vnitřní okraje 10px, rámeček 1px a ignoruje vlastnost `_width`, takže celý box bude mít šířku 522px. IE nastaví šířku boxu 500px, vnitřní okraje 10px, rámeček 1px a znovu nastaví šířku boxu, tentokrát 522px, takže celý box bude mít šířku 522px. Takže výsledná stránka bude mít stejnou šířku boxu jak v IE tak v ostatních prohlížečích.

Bohužel IE 6, který se již chová jako ostatní moderní prohlížeče, ve standardním režimu vlastnost s podtržítkem vidí, a proto je tento hack použitelný jen v případě, že je stránka zobrazována v quirk režimu.

Jak jsem již napsal, jedná se o velice rozšířený a používaný hack, ale pokud by nebylo prohlížečů IE 5 a 5.5,

#### **4.2.2 Hack s uvozovkami**

Jedná se o nevalidní hack, který spočívá v tom, že prohlížeč IE (5, 5.5 a 6 v quirk režimu) ignoruje uvozovky u hodnot vlastností.

```
.test{ width: 500px; padding: 10px; border-width: 1px; border-style: solid; width: „522px“ }
```

Výsledek bude stejný jako u podtržítkového hacku, rozdíl je ten, že IE 6 ve standardním režimu vlastnost s uvozovkami ignoruje, takže je tento hack použitelný i pro standardní režim.

#### **4.2.3 Hack se zpětným lomítkem**

V CSS se zpětné lomítko používá pro zápis znaků sady ISO 10646 (speciální znaky) v dekadickém nebo hexadecimálním tvaru. Zpětná lomítka se můžou také předřazovat písmenům od „g“ až po „z“, aniž by se význam znaku změnil. Předřazovat lomítko nelze písmenům „a“ až „f“, protože mají význam v hexadecimálním tvaru. Pokud vložím zpětné lomítko do vlastnosti v CSS, některé prohlížeče tuto vlastnost budou ignorovat. Příklad:

```
.test{ wi\idth: 500px; }
```

Moderní prohlížeče (Mozilla, IE 6, Opera 6,7, NN 6,7) ve standardním režimu vlastnost přečtou jako width: 500px a zpracují. Starší prohlížeče (IE 5, 5.5, NN 4, Opera 5) a IE 6 v quirk režimu tuto vlastnost ignorují.

## 4.2.4 Hack s rovnítkem a uvozovkami

Specifikace CSS neumožňuje zadávat deklarace s rovnítky a uvozovkami, ale IE 4, 5, 5.5 a IE 6 v quirk režimu to umožňuje. Příklad:

```
.test{width=500px; color: „red“; border-style=“solid“}
```

Mozilla, NN 4, 6, 7, Opera 6, 7 a IE 6 ve standardním režimu rovnítko a uvozovky ignorují.

## 4.2.5 Star html hack

V CSS je znak \* používán jako univerzální selektor. Trik pracuje s elementem div, který je potomkem elementu html.

```
* html div {width: 500px}
```

Žádný prohlížeč mimo IE 5, 5.5. a 6 (jak v režimu quirk tak ve standardním režimu) tento zápis nepoužije, protože prvek html je u běžných dokumentů vždy rodičovský.

## 4.2.6 Čelikův hack

Jedná se zřejmě o jeden z nejznámějších hacků na celém světě. Tento hack využívá toho, že starší prohlížeče (IE 5, 5.5 a IE 6 v quirk režimu) neznají definici `voice-family: “{}”`; a když se při zpracovávání vlastností dostanou k této vlastnosti, zastaví zpracování, přestože se za touto definicí nacházejí další. Příklad:

```
.test{ width: 522px;
padding: 10px;
border-style: solid;
border-width: 1px;
border-color: red;
voice-family: “{}”;
voice-family: inherit;
width: 500px}
```

Moderní prohlížeče (Mozilla, Opera, NN 6,7, IE 6 ve standardním režimu) končí zpracování předpisu u `width: 500px`. IE 5, 5.5 a 6 v quirk režimu skončí u `voice-family: “{}”`. Tento příklad řeší chybné zpracování box modelu u IE.

## 4.2.7 Validita a používání hacků

O validitě a používání CSS hacků se vedly, vedou a budou vést diskuze. CSS hacky nejsou z větší části validní, ale a to je důležité fungují a plní svou funkci. Problém není v hackech, ale v prohlížečích, které ne zcela dodržují specifikace CSS a nutí tak webdesignéry, aby si pomohli a obešli nedostatky v prohlížečích hacky. Až přestanou uživatelé používat staré prohlížeče, kvůli kterým se CSS hacky používají, budeme moci na hacky zapomenout a psát čisté CSS. Ale vymýcení starých prohlížečů je zatím v nedohlednu. Uvidíme co přinese rok 2005, protože Microsoft se rozhodl rozmrazit vývoj IE a v létě vydá verzi 7.

CSS hacky se používají zejména kvůli prohlížečům od Microsoftu, IE 5 a 5.5. Tyto prohlížeče stále používá okolo 10% uživatelů. Na konci roku 2004, měl IE podíl 85%, Gecko (NN, Mozilla, Firefox) 12%, Opera 2,5%, ostatní 0,5%.

Obecně platí, že použití CSS hacku by mělo být až posledním řešením a webdesigner by se měl pokusit pomoci si jinak. Na závěr bych použil citát Pavla Kouta:

*„Validita není dogma, ale vodítko. Mohu jistá pravidla porušit, ale jen tehdy, uvědomuji-li si sakra dobře, co a proč dělám. Dělal-li bych to jen proto, že to dělají ostatní, pak bych byl nebetyčný vůl.“*

## 4.3 Vertikální centrování

Dalším problémem je vertikální centrování některých prvků. Existuje několik možností, jak to řešit.

### 4.3.1 Pomocí tabulky

Vertikální centrování pomocí tabulky pochází z dob, kdy se layout stránek tvořil pomocí tabulek. Je to řešení, kterému se dnes snažíme vyhnout, protože tabulky nejsou primárně určeny pro tvorbu layoutu.

### 4.3.2 Yuhůovo řešení

Řešení, vhodné pro vertikální vycentrování prvků uvnitř nějakého bloku, které vymyslel Dušan Janovský, mezi webdisagnery známý jako Yuhů. Toto řešení je rozděleno na 2 části. První část pro IE (z důvodu špatné interpretace vlastnosti height) a druhá část pro ostatní prohlížeče. Nejlepší bude toto řešení vysvětlit na příkladu.

Příklad:

Chceme vertikálně vycentrovat text. Tento text obalíme do 3 prvků div. Z těchto prvků vlastně vytvoříme jednoduchou tabulku o jedné buňce.

#### Pro prohlížeče mimo IE

Vnější div se označí jako tabulka (display: table), prostřední div se označí jako buňka tabulky (display: table-cell) a pro tento div poté bude fungovat vlastnost vertical-align, pomocí které již obsah buňky zarovnáme na střed. Pro vnitřní div nejsou potřeba žádné vlastnosti.

#### Pro prohlížeč IE

Vnějšímu divu se nastaví relativní pozicování, prostřední div se napozicuje absolutně do poloviny výšky vnějšího divu a pak se posune o polovinu své výšky, což se provede pro vnitřní div.

Na závěr se oba zápisy spojí. Pro IE použijí podtržítkového hacku. Toto řešení funguje pro IE 5, 5.5, 6 (a to jak ve standardním režimu tak i v quirk režimu), v prohlížečích Mozilla, Firefox, NN, Opera 7.

Zdrojový kód:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<title>Yuhůovo řešení</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1250">
<style type="text/css">
.kontik{
display: table;
height: 500px;
width: 770px;
_position: relative;
```

```

overflow: hidden;
border-color: blue;border-style: solid;
border-width: 1px
}
.obal{
  _position: absolute;
  _top: 50%;
display: table-cell;
vertical-align: middle;
}
.cent{
  _position: relative;
  _top: -50%;
border-color: red;
border-style: solid;
border-width: 1px}
</style>
</head>
<body>
<div class="kontik">
  <div class="obal">
    <div class="cent">
      Tento text bude vertikálně vycentrovaný v div class="kontik"
    </div>
  </div>
</div>
</body>
</html>

```

Barevná legenda:

styly pro všechny prohlížeče

styly pro Internet Explorer

styly pro ostatní prohlížeče

Ale pozor, pokud by byl text větší (delší), než vnější div, vertikální vycentrování nebude vidět a text bude oříznut a nebude vidět celý. Je také potřeba dávat pozor na chybný výpočet box modelu u IE při nastavení vnitřních okrajů boxu. Na chybný výpočet mají vliv také rámečky, ale při šířce rámečku 1px je tento vliv minimální.

### 4.3.3 Victoriino vertikální centrování

Toto řešení vymyslel Vilém Málek. Používá se pro vertikální vycentrování bloku na stránce. Nejlépe vše pochopíte z příkladu.

Chci, aby byl obsah stránky vycentrován na střed. Obsah stránky obalím do prvku div. Celý trik spočívá v tom, že prvkům html a body nastavím vlastnost height: 100% a tím tak donutím prohlížeče, aby použili výšku okna pro výpočet procentně zadaných výšek v něm umístěných prvků. Pro prvek div stačí nastavit rozměry (height, width), absolutní pozicování a potřebné odsazení shora a zleva.

Zdrojový kód:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Victoriino vertikální centrování</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1250">
<style type="text/css">
html, body {
  height: 100%
}
.center{
  height: 50%;
  width: 50%;
  position: absolute;
  left: 25%;
  top: 25%;
  border-color: red;
  border-style: solid;
```

```
border-width: 1px;
overflow: scroll;
}
</style>
</head>
<body>
<div class="center">
Tento div by bude vertikálně i horizontálně vycentrovaný.
</div>
</body>
</html>
```

Důležité je dávat si pozor na velikosti okrajů, pokud velikosti okrajů budou nulové, je to v pořádku, pokud však nastavím vnitřní okraj prvku div, objeví se již výše řešený problém s box modelem. Pokud bude obsah boxu větší, je třeba nastavit boxu vlastnost overflow:scroll, což umožní posouvání v boxu. Procentuální nastavení výšky a šířky má tu nevýhodu, že pokud zmenším okno prohlížeče, zmenší se i velikost boxu. Toto řešení vertikálního centrování lze použít i pro box s pevnými rozměry, ale je potřeba vypočítat přesné hodnoty odsazení zleva a shora.

#### 4.3.4 Shrnutí

Vertikální centrování není věc, kterou by webdesignéři často používali, ale někdy se hodí. Když jsem poprvé řešil vertikální vycentrování textu v bloku, dlouho jsem si s tím lámal hlavu a nakonec mi pomohlo Yuhůovo řešení. Jak řešení Yuhůovo, tak Victoriino řešení fungují v prohlížečích IE 5, 5.5, 6, Mozilla, NN, Opera 7. Takže se dá říci, že jsou to vhodná pro vertikální centrování. Naopak řešení pomocí tabulek je již zastaralé a nevyhovující standardům a proto se doporučuje ho nepoužívat.

## 4.4 Internet Explorer 7

Společnost Microsoft před časem zmrazila svůj program vývoje webového prohlížeče IE, ale narůstající konkurence a vývoj nového operačního systému přiměl Microsoft tento program rozmrazit a v létě letošního roku vydá beta verzi IE 7. O připravovaném prohlížeči se toho ví velice málo, protože Microsoft vydává jen obecná prohlášení, z kterých není úplně jasné, co mají uživatelé očekávat. Nejvíce se hovoří o bezpečnosti, zobrazovacím jádru a standardech, uživatelském rozhraní a rozšířeních.

### 4.4.1 Bezpečnost

Slovo bezpečnost se v posledních měsících v souvislosti s webovými prohlížeči skloňuje ve všech pádech. Z prohlášení Microsoftu se zdá, že bezpečnost prohlížeče je na prvním místě. Přece jen, produkty od Microsoftu jsou známé tím, že v okamžiku jejich vydání nejsou zcela bezpečné a Microsoft poté musí vydávat různé opravné a bezpečnostní balíčky, které je potřeba doinstalovat. Zkrátka IE 7 by měl, podle prohlášení Microsoftu, lépe odolávat spyware, adaware, malware, phishing, spamu, virům, červům, trojským koním.

### 4.4.2 Zobracovací jádro a standardy

Otázky týkající se zobrazovacího jádra a podpory CSS standardů jsou asi nejdiskutovanější téma mezi webdesignéry. Je několik možností, jak to se zobrazovacím jádrem a standardy dopadne.

IE 7 převezme jádro i podporu standardů od IE 6 a vše zůstane při starém. Při tvorbě webových stránek by se používaly stále stejné postupy, nic by se neulehčilo, ale ani neztížilo. Tato možnost je dost nepravděpodobná, protože by Microsoft zaostal ještě více za svými konkurenty.

Zcela nové jádro a podpora všech standardů. Nové jádro by jistě zkomplikovalo Microsoftu zpětnou kompatibilitu (na které Microsoft velice lpí) se staršími IE, a proto je i tato možnost nepravděpodobná.

Podle prohlášení, že Microsoft zkoumá potřeby webdesignérů a podle potřeby doplní, či opraví chyby v IE 7 se dá očekávat, že se jádro moc nezmění. V quirk režimu se stránky budou zobrazovat stejně jako v IE 5.5, ve standardním režimu snad bude IE 7 podporovat většinu CSS standardů a doufejme, že správně. Tato možnost se jeví jako nejpravděpodobnější.

### **4.4.3 Uživatelské rozhraní**

Pro uživatele je uživatelské rozhraní, tedy vzhled prohlížeče, jak se sním pracuje tou nejdůležitější věcí. Když se podíváme na změny vzhledu IE v průběhu let, dá se s jistotou říci, že se nedají očekávat žádné převratné změny, spíše naopak. Uživatelé jsou na vzhled IE zvyklí a pokud by se do něj udělal větší zásah, nemuselo by to být ku prospěchu, uživatelé by neinstalovali nový IE 7, ale zůstali by u svých starých IE, na které jsou zvyklí. Informace o vzhledu IE 7 zatím Microsoft nezveřejnil, ale dá se očekávat možnost tabbed browsing (záložky) a integrace RSS čtečky.

K uživatelskému rozhraní bych ještě zařadil informaci, že IE 7 bude k dispozici pouze pro Windows XP. Nedá se předpokládat, že by Microsoft uvolnil IE 7 i pro jiné operační systémy než jsou Windows, ale že bude IE 7 použitelný jen ve Windows XP je špatná zpráva pro uživatele starších verzí Windows.

### **4.4.4 Rozšíření**

Ani zde se nedají očekávat převratné změny oproti IE 6. Podpora Javascriptu je u IE asi jedna z nejlepších, přesto se i v ní najdou nějaké chyby. Na jejich odstranění se zřejmě v Microsoftu zaměří. Podpora Flash v IE funguje bez větších problémů, takže ani zde se nedají změny očekávat.

Existují dvě technologie, na které by se mohl Microsoft zaměřit. První je podpora formátu PNG. A druhá je podpora SVG, což je vektorový grafický formát, jehož integrací by se podstatně rozšířili možnosti webdesignerů.

#### 4.4.5 Shrnutí

Vzhledem k tomu, že podíl IE je okolo 85%, je vydání beta verze IE 7 spojeno s velkými nadějemi a očekáváním. Ostré verze IE 7 se nejspíše dočkáme až s novým operačním systémem Longhorn, ale už beta verze může mnohé změnit a naznačit vývoj prohlížeče. Můžeme doufat, že uživatelé starších verzí IE (zejména verzí 5 a 5.5) přestanou tyto staré a dnes již nevyhovující prohlížeče používat a zvolí si nějaký moderní prohlížeč. Doufejme také, že vývojáři IE se poučí ze svých předešlých chyb a vytvoří prohlížeč, který bude správně podporovat CSS standardy a ulehčí tak webdesignérům práci. Nevěřím moc tomu, že by beta verze byla bez chyb a bezpečnostních problémů. Určitě se můžeme těšit na spoustu opravných a bezpečnostních balíčků, jak už to tak u softwaru od Microsoftu bývá. V každém případě se máme na co těšit.

## 4.5 Analýza www.seznam.cz

Na závěr jsem se rozhodl analyzovat zřejmě nejznámější český internetový rozcestník www.seznam.cz. Rozhodl jsem se zaměřit zejména na řešení layoutu stránky.

**SEZNAM**

Vítejte! [Nastavit oslovení >](#)  
Středa 13.4.2005, svátek má Aleš.  
[Nastavit Seznam jako domovskou stránku!](#)

[Napiš si poznámku](#) [Seznam se](#)

Internet **Firmy** Mapy Slovník

Hledej v ČR [Poločas hledání](#)

[Auto](#) [Finance](#) [Počasí](#) [SMS, logo](#)  
[Bazar](#) [Horoskopy](#) [Práce](#) [Sport](#)  
[Bezpečný Inet](#) [Hry](#) [Připojení](#) [Super nově!](#)  
[Denní tisk](#) [Chat LIVE](#) [Reality](#) [TV program](#)  
[Dovolená](#) [Kina](#) [Seznamka](#) [Zprávy](#) [Novinky](#)  
[E-mail](#) [Mapy](#) [Slovník](#) [Více služeb >](#)

Náš tip: Překvapivě výhodná CITIBANK osobní půjčka.

**Katalog** A B C Ď E F G H CH I J K L M N O  
P Q R Ř S Š T U Ú V W X Y Z Z

Auto Moto (11.480)	Hokej (1.020)	Počítače (10.820)	Technika (3.340)
Autobazary (990)	Hry (5.060)	Pohostinství (5.530)	Ubytování (7.120)
Banky (9.770)	Hudba (9.440)	Práce (130)	Úřady (1.730)
Bydlení (7.380)	Informace (6.000)	První pomoc (32.170)	Věda (9.050)
Cestování (7.480)	Instituce (25.760)	Půjčovny (940)	Velkoobchody (6.530)
Časopisy (2.500)	Jízdní řády (30)	Rádio (330)	Vše pro firmy (15.000)
Deníky (50)	Knihy (240)	Reality (2.320)	Výrobci (2.650)
Elektro (10.350)	Koníčky (8.670)	Restaurace (3.690)	Zábava (9.900)
E-shopy (5.950)	Kultura (3.490)	Služby (49.070)	Zdraví (1.000)
Film (850)	Mobily (540)	Sport (11.120)	Zpravodajství (4.250)
Fotbal (1.800)	Nábytek (100)	Stavebnictví (11.160)	
Fotografie (1.240)	Obchody (20.650)	Školy (13.740)	

[Více kategorií >](#)

**Sbazar** [Nejpřehlednější inzertní server v ČR](#) **www.SBAZAR.CZ**

- FUJIFILM Finepix 2300, 2.1MP, usb kabel, 64MB. Cena: **1 500,- Kč** >>
- PC 486 DX2 66Mhz, RAM 8MB, HDD 220 MB. Cena: **za odvoz** >>
- Jawa ČZ 250, r.v. 1959, po rekonstrukci, nový lak. Cena: **4 000,- Kč** >>
- Výkonný univerzální vysavač VAX 4100, 450-1500 W. Cena: **4 500,- Kč** >>

[Podělte si soukromý inzerát zdarma!](#) [Více inzerátů >](#)

**E-mail** [Jméno:](#)  @seznam.cz  
[Heslo:](#)  [Přihlásit](#)  
[Založit novou e-mailovou schránku](#)

**Zprávy** [dnes 13 až 17°C](#) [zítra 16°C](#)

**Topolánek pro duhovou koalici**

- [Nehodu hasičské cisterny zřejmě způsobil řidič](#)
- [Soud opět osvobodil ruského skinheada](#)
- [České stamilióny Asii pomohly dětem a rybářům](#)
- [Třináct let za vraždu šestnáctileté dívky](#)
- [Popová zpěvačka Britney Spearsová je těhotná](#)
- [LM: UEFA oznámí výsledek derby v Miláně v pátek](#)
- [Venera má za úkol dostat Karlov Vary do play-off](#)

**11:52** Složení nové irácké vlády by mělo být oznámeno v neděli 17. dubna, na kdy se plánuje další parlamentní zasedání. Sdílel ...

[Více sportů >](#) [Více zpráv >](#)

**Horoskop** [Nastavit znamení](#) [Více horoskopů >](#)

**Lev** Hlavní aspekty dneška se odehrávají v protějším znamení. Možná se dnes budete cítit poněkud odstrčeni... >>

**Společnost** [Nastavit znamení](#) [Více horoskopů >](#)

**Ivana Christová: "Jsem tlustá a nešťastná"**

- [Co se to děje s Bohoušem Josefem?](#)
- [V redakci je horko: Sex se dostal i mimo natáčení](#)
- [Maxa dostal podmínku!](#)

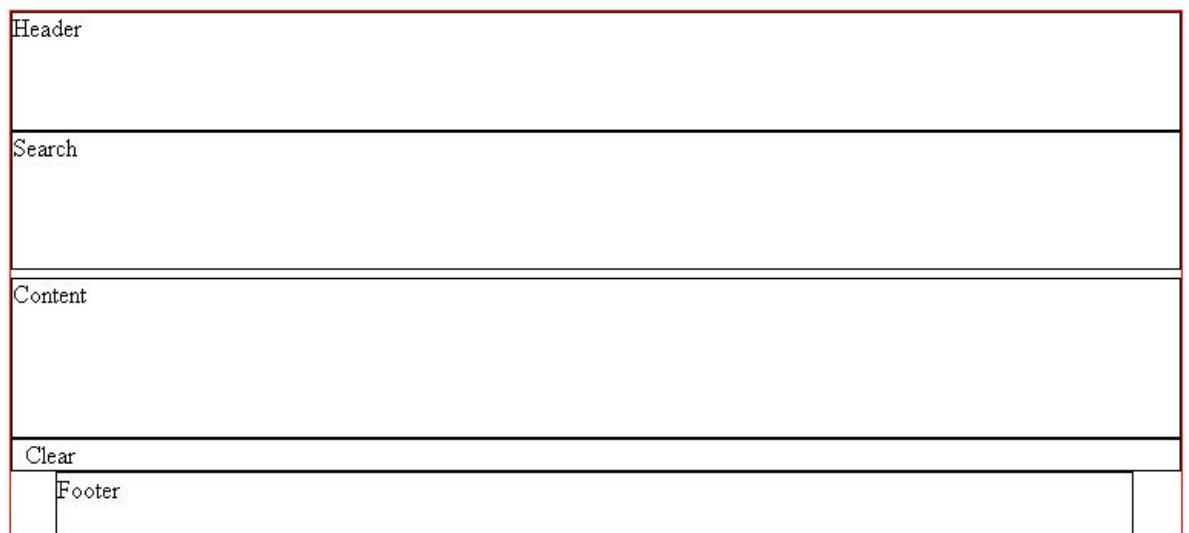
[Více Super zpráv >](#)

Jak přidat do Seznamu [firmu](#) / [odkaz](#) - [O Seznamu](#) - [Volná místa](#) - [Reklama](#) - [Banka](#) - [Návoděda](#) - [Technická podpora](#)  
Copyright (c) 1996 - 2005, Seznam.cz, a.s. - Připojeno k: [net4net](#)

Obrázek 3 - www.seznam.cz

## 4.5.1 Rozbor layoutu

Celý obsah stránky je uzavřen v prvku <div id="page"> (na obrázku červený rámeček). Box má nastavenou šířku 760px. Tato šířka je nastavena z toho důvodu, že mnoho uživatelů dnes ještě používá monitory s rozlišením 800x600 a kdyby byla šířka větší, museli by tito uživatelé posouvat stránku do stran, aby ji viděli celou. Tento box je pozicován na střed (vycentrován) a má nastavené nulové vnitřní okraje. Uvnitř tohoto boxu se nachází dalších 5 boxů.



Obrázek 4 - Page

Box Header obsahuje logo, uvítání, možnost nastavení oslovení, datum, informaci o tom, kdo má dnes svátek a možnost nastavit stránku jako domovskou. Ikony v pravé části boxu nejsou umístěny v boxu, jsou zde umístěny pomocí absolutního pozicování.



Obrázek 5 - Header

Box Search obsahuje možnosti vyhledávání v internetu, firmách, mapách a slovníku.



Obrázek 6 - Search

Box Content se skládá boxů Main, Right, Bottom. Tyto boxy obsahují odkazy na další podstránky, reklamy, přístup k e-mailové schránce, možnost založení nové e-mailové schránky, předpověď počasí, zpravodajství, horoskop, informace ze společnosti a nabídky bazaru.

**Auto**  
**Bazar**  
**Bezpečný Inet**  
**Denní tisk**  
**Dovolená**  
**E-mail**

**Finance**  
**Horoskopy**  
**Hry**  
**Chat**  
**Kina**  
**Mapy**

**Počasí**  
**Práce**  
**Připojení**  
**Reality**  
**Seznamka**  
**Slovník**

**SMS, logo**  
**Sport**  
**Super nově!**  
**TV program**  
**Zprávy**  
**Více služeb >**

Náš tip: Nejširší nabídku biopotravin najdete na [www.multimarket.cz](http://www.multimarket.cz)

**Katalog** A B C Č D E F G H CH I J K L M N O P Q R Ř S Š T U Ú V W X Y Z Ž

Auto Moto (11.480)	Hokej (1.020)	Počítače (10.820)	Technika (3.340)
Autobazary (990)	Hry (5.060)	Pohostinství (5.530)	Ubytování (7.120)
Banky (9.770)	Hudba (9.440)	Práce (130)	Úřady (1.730)
Bydlení (7.380)	Informace (6.000)	První pomoc (32.170)	Věda (9.050)
Cestování (7.480)	Instituce (25.760)	Půjčovny (940)	Velkoobchody (8.530)
Časopisy (2.500)	Jízdní řády (30)	Rádio (330)	Vše pro firmy (15.000)
Deníky (60)	Knihy (240)	Reality (2.320)	Výrobci (2.650)
Elektro (10.350)	Koničky (8.870)	Restaurace (3.890)	Zábava (8.900)
E-shopy (5.950)	Kultura (3.490)	Služby (49.070)	Zdraví (1.000)
Film (850)	Mobily (640)	Sport (11.120)	Zpravodajství (4.250)
Fotbal (1.800)	Nábytek (100)	Stavebnictví (11.150)	
Fotografie (1.240)	Obchody (20.650)	Školy (13.740)	<b>Více kategorií &gt;</b>

**Sbazar** Nejlepší inzerční server v ČR [www.SBAZAR.CZ](http://www.SBAZAR.CZ)

- Notebook Asus, Intel Celeron 2.6GHz, 256MB SDRAM. Cena: **15 800,- Kč** >>
- Krémový kabát s umělou kožesinou. Téměř nenošený. Cena: **za odvoz** >>
- Motocykl Kawasaki GPZ 750 stříbrné barvy, r.v. 1985. Cena: **25 000,- Kč** >>
- Digitální fotoaparát Canon Powershot A 95, 256Mb karta. Cena: **8 000,- Kč** >>

[Podejte si soukromý inzerát zdarma!](#) **Více inzerátů >**

**E-mail** Jméno:  @seznam.cz  
Heslo:  **Přihlásit**  
**Založit novou e-mailovou schránku**

**Zprávy** dnes 13 až 17°C zítřka 16°C

**Evroparlament schválil členství Rumunska a Bulharska v EU**

- Gross: Dnes se musí rozhodnout
- Vondráčková vysoudila na časopisu čtvrt miliónu
- Únosce dětí v Německu není psychicky zdrav
- Třináct let za vraždu šestnáctileté dívky
- Dvě stě cvičných veverek v novém Burtonově filmu
- LM: UEFA oznámí výsledek derby v Miláně v pátek
- ČECH: Tři góly nepotěší, ale nejdůležitější je postup

**13:57** Armáda České republiky převzala první tři bitevní vrtulníky typu Mi-24 z Ruské federace v rámci deblokace ruského dluhu. ...

**Horoskop** **Více sportu >** **Více zpráv >**

**Váhy** V podnikání nebo milostném životě se dnes naskytne možnost volby. Někteří se seznámí s velmi různou... >>

**Společnost** [Nastavit znamení](#) **Více horoskopů >**

**Ivana Christová: "Jsem tlustá a nešťastná"**

- Benešová z Prásku: Jsem ve čtvrtém měsíci
- Heidi nechce slavné děti
- Smrt Stoklasvy zavinil řidič

**Více Super zpráv >**

Obrázek 7 - Content

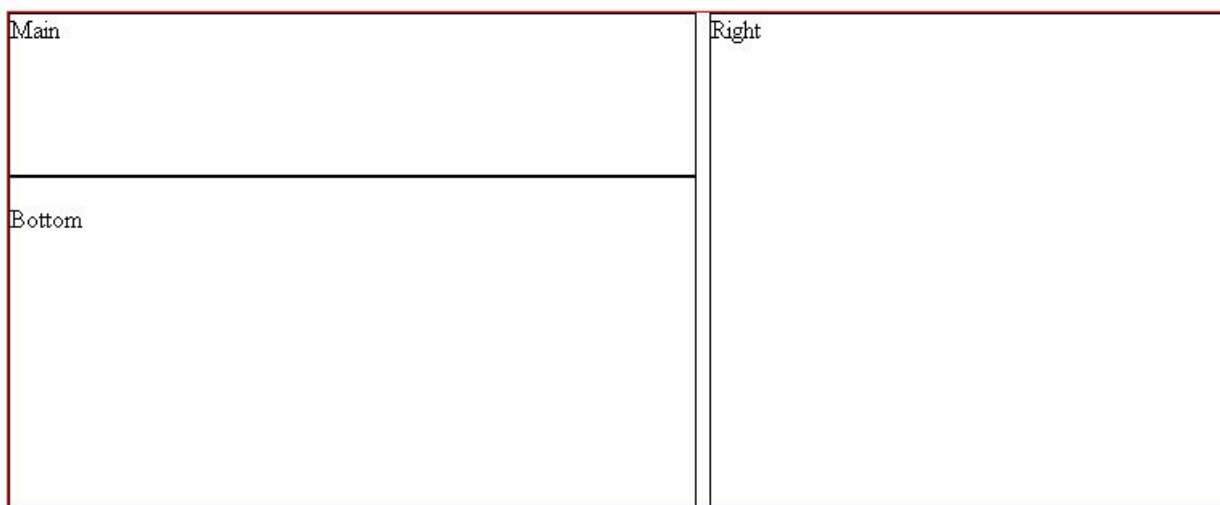
Box Clear slouží jen k oddělení boxu Content od boxu Footer. Vytváří mezi nimi mezeru.

Box Footer obsahuje informace a pomoc pro uživatele.

Jak přidat do Seznamu [firmu](#) / [odkaz](#) - [O Seznamu](#) - [Volná místa](#) - [Reklama](#) - [Banka](#) - [Nápověda](#) - [Technická podpora](#)  
Copyright (c) 1996 - 2005, Seznam.cz, a.s. - Připojeno k: [net4net](#)

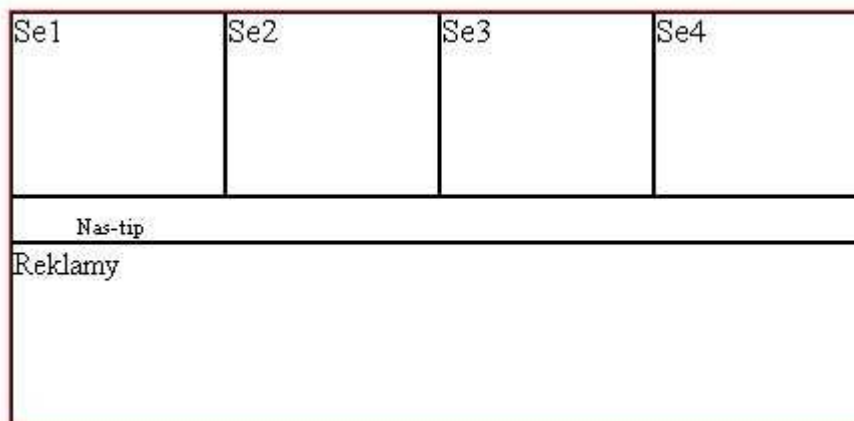
Obrázek 8 - Footer

Jak jsem již výše napsal, box Content se skládá z dalších boxů (Main, Right, Bottom).



Obrázek 9 - Content

Box Main obsahuje dalších 6 boxů. Uvnitř boxů Se1, Se2, Se3 a Se4 jsou odkazy na různé servery (např. auto, hry, kina, atd.). Box Nas-tip obsahuje odkaz na webovou stránku se zajímavou nabídkou, a který Seznam propaguje jako svůj tip pro tento den. Box Reklamy obsahuje placenou reklamu.

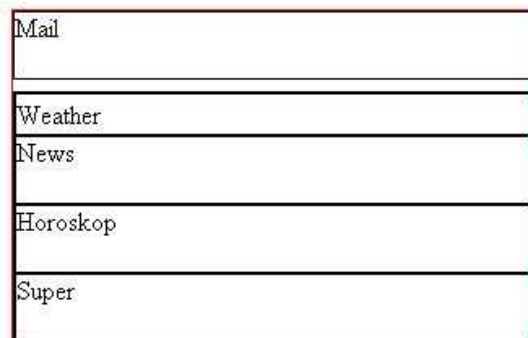


Obrázek 10 – Main

Box Right se skládá ze 2 boxů (Mail, Headlines). Box Mail obsahuje možnost přístupu k e-mailové schránce, možnost založení nové e-mailové schránky. Box Headlines se skládá z dalších 4 boxů. V boxu Weather se nachází aktuální zpravodajství o počasí, v boxu News aktuální zpravodajství, v boxu Horoskop si uživatel může přečíst horoskop pro tento den a v boxu Super jsou odkazy na zpravodajství ze společnosti.



Obrázek 11 - Right

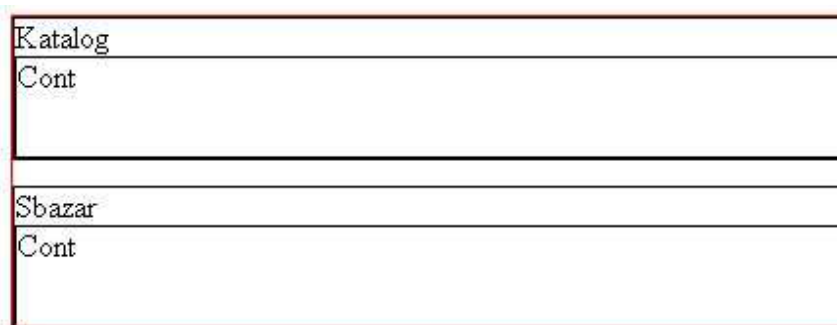


Obrázek 12 - Right - Headlines

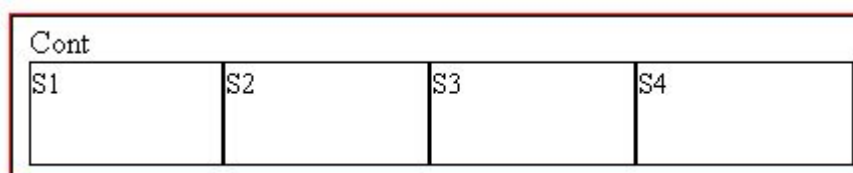
Box Bottom se skládá z boxů Katalog a Sbazar. Box Katalog obsahuje box Cont, který se skládá z boxů S1, S2, S3 a S4. Box Sbazar obsahuje box Cont.

V boxu Katalog jsou odkazy A – Z, které vedou na stránky katalogu řazené podle abecedy. V boxech S1, S2, S3 a S4 se nacházejí odkazy na jednotlivé kategorie katalogu (např. Banky, Bydlení, Hudba, Knihy a další).

V boxu Sbazar se nachází odkaz na webové stránky Sbazaru a logo Sbazaru. V boxu Cont se nachází výběr z nabídek Sbazaru.



Obrázek 13 - Bottom



Obrázek 14 - Bottom – Katalog

## 4.5.2 CSS

Tělo dokumentu má nastavené nulové vnitřní a vnější okraje, bílou barvu pozadí (#fff), rodinu písma, která bude použita v celém dokumentu ("Arial CE",Arial,"Helvetica CE",Helvetica,sans-serif), velikost písma 75% (relativní velikost vzhledem k velikosti písma rodičovského prvku) a celý obsah dokumentu je vycentrován na střed pomocí vlastnosti text-align: center (vycentrování pro IE).

```
body {
    padding-right: 0px;
    padding-left: 0px;
    font-size: 75%;
    padding-bottom: 0px;
    margin: 0px;
    padding-top: 0px;
    font-family: "Arial CE",Arial,"Helvetica CE",Helvetica,sans-serif;
    background-color: #fff;
    text-align: center}
```

Tělo dokumentu obsahuje box page, kterému jsou nastaveny tyto vlastnosti – šířka 760px, nulové vnitřní okraje (řeší tak chybný box model v IE, šířka boxu se zobrazí ve všech prohlížečích stejně), vnější levý a pravý okraj je nastaven na hodnotu auto (vycentrování pro všechny prohlížeče mimo IE), nulový vnější horní okraj, vnější dolní okraj nastaven na 0,5em, box je relativně pozicován, nastaveno zarovnání doleva (text-align: left) z toho důvodu, že vycentrování, které bylo provedeno v těle dokumentu (text-align: center) nastavilo vycentrování pro všechny prvky dokumentu.

```
#page {
    padding-right: 0px;
    padding-left: 0px;
    padding-bottom: 0px;
    margin: 0px auto 0.5em;
    width: 760px;
    padding-top: 0px;
    position: relative;
    text-align: left}
```

Jak jsem výše napsal, box Page se skládá z dalších boxů. Box Header má nastavenou jen výšku 74px. Šířka je 760px. Není sice ve stylovém předpisu zadána, ale vlastnost width má výchozí hodnotu auto, což je u blokových prvků 100%. V tomto případě tedy 100% ze 760px. Vlastnosti vnitřních a vnějších okrajů také nejsou nastaveny a tak prohlížeč použije výchozí hodnotu pro tyto vlastnosti, která je 0px.

```
#header {  
    height: 74px}
```

Box Search je relativně pozicován, má nastavenou výšku 5.4em, zarovnáno doleva a hodnota z-index nastavena 1 (kromě horizontální a vertikální osy je možné pozicovat prvky ještě v ose z pomocí vlastnosti z-index).

```
#search {  
    z-index: 1;  
    position: relative;  
    height: 5.4em;  
    text-align: left}
```

Box Content se skládá z dalších boxů, má nastavenou šířku 760px a vnější horní okraj 5px, ostatní vnější okraje jsou nulové.

```
#content {  
    margin: 5px 0px 0px;  
    width: 760px}
```

Box Clear slouží jen k oddělení boxů Content a Footer. Opět používá výchozí hodnoty pro šířku a vnitřní a vnější okraje. Má nastavenou vlastnost clear: both, to znamená, že je zařazen za všechny plovoucí boxy uvedené dříve ve zdrojovém kódu a po obou stranách bude prázdné místo, nebudou se zde nacházet žádné prvky.

```
.clear {  
    clear: both}
```

Box Footer má nastavenou šířku 700px, vnitřní horní a dolní okraj 0px a levý a pravý okraj auto (vycentrování). Má nastavenou vlastnost clear: both, box bude zařazen pod všechny plovoucí prvky uvedené dříve ve zdrojovém kódu. Horní orámování o šířce 1px, plnou čarou a barvou #bbbbbb (šedivá). Velikost písma 100% a výšku řádku 1.25em.

```
#footer {
  clear: both;
  border-top: #bbb 1px solid;
  font-size: 100%;
  margin: 0px auto;
  width: 700px;
  line-height: 1.25em}
```

Box Main je plovoucí box v levé části boxu Content, má nastavenou šířku 427px.

```
#main {
  float: left;
  width: 427px}
```

Box Right je plovoucí box v pravé části boxu Content, má nastavenou šířku 321px.

```
#right {
  float: right;
  width: 321px}
```

Box Bottom je pozicován pod box Main vlevo od Boxu Right, má nastavenou šířku 427px, levý pravý a dolní okraj 0px a horní okraj 1em. Vlastnost clear:left určuje, že se tento box zařadí pod plovoucí box Main.

```
#bottom {
  clear: left;
  padding-right: 0px;
  padding-left: 0px;
  padding-bottom: 0px;
  width: 427px;
  padding-top: 1em}
```

Box Se1 umístěný v boxu Main nemá nastavenou žádnou vlastnost. Boxy Se2, Se3 a Se4 umístěné v boxu Main mají nastavenou jen šířku.

```
#se1 {}  
#se2 {  
  width: 107px}  
#se3 {  
  width: 112px}  
#se4 {  
  width: 97px}
```

Box Nas-tip v boxu Main má nastaven pravý a dolní okraj 0, levý okraj 32px a horní okraj 0.6em. Velikost písma 11px, vlastnost clear:both určuje, že se box bude nacházet pod plovoucími boxy uvedené dříve ve zdrojovém kódu. Všechny odkazy v tomto boxu budou mít šedivou barvu (#a6a6a6).

```
#nas-tip{  
  clear: both;  
  padding-right: 0px;  
  padding-left: 32px;  
  font-size: 11px;  
  padding-bottom: 0px;  
  padding-top: 0.6em}  
#nas-tip A {  
  color: #a6a6a6}
```

Box Reklamy v boxu Main je pozicován relativně, je opět umístěn pod všechny plovoucí boxy dříve uvedené ve zdrojovém kódu a má výšku 90px.

```
#reklamy {  
  clear: both;  
  position: relative;  
  height: 90px}
```

Box Mail v boxu Right je relativně pozicován, má nastavenou šířku 319px, vnější spodní okraj 7px, ostatní vnější okraje jsou nulové, orámování plnou čarou o šířce 1px má barvu #e8a883, barva pozadí #ffe5d4, barvu písma #ff6911.

```
#mail {
border-right: #e8a883 1px solid;
border-top: #e8a883 1px solid;
background: #ffe5d4;
margin: 0px 0px 7px;
border-left: #e8a883 1px solid;
width: 319px! important;
color: #ff6911;
border-bottom: #e8a883 1px solid;
position: relative}
```

Box Headlines se skládá z dalších 4 boxů, má nastaveno jen orámování plnou čarou, šířky 1px, barvy #9ed2a3.

```
#headlines {
border-right: #9ed2a3 1px solid;
border-top: #9ed2a3 1px solid;
border-left: #9ed2a3 1px solid;
border-bottom: #9ed2a3 1px solid}
```

Box Weather v boxu Headlines má nastavenou výšku 24px, levý vnitřní okraj 70px, ostatní okraje jsou nulové, barvu pozadí #e7ffe9 a výšku řádku 24px.

```
#weather {
padding-right: 0px;
padding-left: 70px;
background: #e7ffe9;
padding-bottom: 0px;
line-height: 24px;
padding-top: 0px;
height: 24px}
```

Boxy News, Horoskop a Super v boxu Headlines mají vlastnosti určené třídou item. Jejich šířka je 319px, jsou relativně pozicovány a horní orámování je plnou čarou, šířky 1px a barvou #9ed2a3. Box News má navíc nastaven styl orámování none.

```

#news {
border-top-style: none;
border-right-style: none;
border-left-style: none;
border-bottom-style: none}
#headlines .item {
border-top: #9ed2a3 1px solid;
width: 319px! important;
position: relative}

```

Box Katalog v boxu Bottom má nastavenou šířku 425px, orámování plnou čarou, šířky 1px, barva #eede71. Je relativně pozicován.

```

#katalog {
clear: left;
border-right: #eede71 1px solid;
border-top: #eede71 1px solid;
border-left: #eede71 1px solid;
width: 425px;
border-bottom: #eede71 1px solid;
position: relative}

```

Box Sbazar v boxu Bottom má nastavenou šířku 425px, orámování plnou čarou, šířky 1px, barva #d6bdef, má nastaven horní vnější okraj 0.8em, je relativně pozicován.

```

#sbazar {
clear: left;
border-right: #d6bdef 1px solid;
border-top: #d6bdef 1px solid;
margin-top: 0.8em;
border-left: #d6bdef 1px solid;
width: 425px;
border-bottom: #d6bdef 1px solid;
position: relative}

```

Box Cont v boxu Katalog je plovoucí box vlevo, má nastavenou šířku 416px, vnitřní horní okraj 3px, pravý 0px, dolní 5px a levý 8px.

```
#katalog .cont {  
    padding-right: 0px;  
    padding-left: 8px;  
    float: left;  
    padding-bottom: 5px;  
    width: 416px! important;  
    padding-top: 3px }
```

Box Cont v boxu Sbazar má nastaven vnitřní horní okraj 5px, pravý 0px, dolní 0px a levý 10px, velikost písma 92%.

```
#sbazar .cont {  
    padding-right: 0px;  
    padding-left: 10px;  
    font-size: 92%;  
    padding-bottom: 0px;  
    padding-top: 5px }
```

Box Cont v boxu Katalog se skládá z dalších 4 boxů S1, S2, S3 a S4. Všechny tyto boxy jsou plovoucí vlevo, mají nastaveny šířku 101px, vnější a vnitřní okraje 0px, velikost písma 92% a výšku řádku 1.4em. pro box S1 je zvlášť nastavena šířka 95px a pro box S4 107px.

```
#katalog .cont div {  
    padding-right: 0px;  
    padding-left: 0px;  
    font-size: 92%;  
    float: left;  
    padding-bottom: 0px;  
    margin: 0px;  
    width: 101px;  
    line-height: 1.4em;  
    padding-top: 0px }  
#katalog .cont .s1 {  
    width: 95px }
```

```
#katalog .cont .s4 {  
  width: 107px}
```

Box Cont v boxu Sbazar má nastaven vnitřní levý okraj 10px, horní 5px, pravý a dolní 0px. Velikost písma je 92%.

```
#sbazar .cont {  
  padding-right: 0px;  
  padding-left: 10px;  
  font-size: 92%;  
  padding-bottom: 0px;  
  padding-top: 5px }
```

Jak je vidět, webdesigneri Seznamu se při tvorbě layoutu webové stránky obešli bez hacků. Pomohli si zanořováním prvků div. Zdrojový kód webové stránky je delší a složitější, než při použití hacků, ale je zcela validní.

Osobně dávám přednost hackům před zanořovanými prvky div. Zápis zdrojového kódu je kratší a řekl bych i jasnější a přehlednější, než při zanořování prvků div. Některé hacky sice nejsou validní, ale je to pohodlnější, rychlejší a plní stejnou funkci.

## Závěr

V mé absolventské práci jsem se snažil shrnout základní poznatky o kaskádových stylech, jejich použití a různé problémy, s kterými se setkáváme při tvorbě webových stránek. V úvodu práce jsem stručně vysvětlil historii Internetu a webových stránek. Poté jsem se věnoval jazyku (X)HTML, jeho funkci a možnostem a nakonec jsem se věnoval kaskádovým stylům, které spolu s XHTML představují nejmodernější způsob formátování webových stránek. V závěrečné části práce jsem se věnoval aktuálním problémům kaskádových stylů, zejména co se týká interpretace CSS v různých webových prohlížečích a různým tipům a trikům, jak se těmto problémům vyhnout.

Z mé práce vyplývá, že formátování webových stránek za pomoci HTML je zastaralé a dnes již prakticky nepoužitelné. Jsou pro to dva hlavní důvody. První důvod je, že by se od sebe měla oddělovat struktura a layout dokumentu, což samotné HTML neumožňuje. Druhý důvod se objevil s nástupem jazyka XHTML, který některé formátovací HTML tagy nepodporuje a tudíž jsou pro něj nepoužitelné. Nejúčinnější a nejmodernější prostředek pro formátování webových stránek jsou kaskádové styly.

S kaskádovými styly jsou spojena různá úskalí, zejména podpora a implementace CSS v různých webových prohlížečích. Největší problémy způsobuje IE, který špatně interpretuje některé standardy. IE je nejrozšířenější prohlížeč, takže je mnoho webových stránek optimalizováno přímo pro něj, ale v poslední době se začínají prosazovat nové prohlížeče, které standardy interpretují správně. To nutí tvůrce webových stránek, aby své stránky optimalizovali pro všechny prohlížeče. Používají k tomu různé triky, nejznámější z nich jsem popsal ve své práci.

Poslední částí práce jsem rozebral layout [www.seznam.cz](http://www.seznam.cz). Ukázal jsem, jakým způsobem je tvořen a jak se webdesignéři vypořádali s problémy, které CSS a jejich implementaci přináší.

Naučit se používat kaskádové styly není nic obtížného, pokud má člověk čas a trpělivost. Čím hlouběji pronikám do problematiky kaskádových stylů zjišťuji, jak je rozsáhlá, ale také jaké netušené možnosti mi kaskádové styly jako webdesignerovy poskytují.

# Resumé

The name of my thesis is „Possibilities and problems of cascading style sheets at the creation of web presentations“. I have chosen this subject because I am interested in the creation of web pages and I want to attend to it in the future. At the beginning of my work I am dealing with the history and evolution of the Internet and creation and formatting of web pages through (X)HTML (Extended Hypertext Markup Language). In the next part there follows the history and development of CSS (Cascading Style Sheets), connection of CSS and (X)HTML and formatting web pages through CSS. In the last part I am going into the problems with viewed and implementation of CSS in different web browsers, tips and tricks how to reach the point where the web pages are viewed in all browsers in the same way. I am engaged in expected presentation of the new version of web browser IE7. In entire conclusion I am attempting to take into parts the source code of [www.seznam.cz](http://www.seznam.cz) and to show how webdesigners solve the layout of this web page. Cascading style sheets together with (X)HTML are the most modern and the most effectual instruments for creation of web pages. There are plenty of difficulties, but it is rather a problem of web browsers than cascading style sheets. It is not difficult to learn CSS if man has got enough time and patience. The deeper I am passing through the problems of CSS I am gathering how extensive it is, but also what unsuspected potentialities CSS offer to webdesigners.

## Seznam použité literatury

1. Kosek Jiří: *HTML – tvorba dokonalých WWW stránek – dokonalý průvodce*. Grada Publishing, spol. s r.o. 1998. ISBN 80-7169-608-0
2. Prokop Marek: *CSS – kaskádové styly pro webdesignéry*. 1.vydání. Mobil Media a.s. 2003. ISBN 80-86593-35-5

### Webové stránky

1. <http://www.jakpsatweb.cz>
2. <http://www.kosek.cz>
3. <http://www.pixy.cz>
4. <http://www.wellstyled.com>
5. <http://weblog.plavacek.net/>
6. <http://www.sovavsiti.cz/>
7. <http://www.vitdlouhy.cz/weblog/index.php>
8. <http://www.mraveniste.org/>
9. <http://www.w3.org/>
10. <http://www.zpravy.net/>
11. <http://www.webtvorba.cz/>
12. <http://www.maciweb.net/blog/web/>
13. <http://interval.cz/>
14. <http://www.moderniweb.cz/>
15. <http://www.dobryweb.cz/>
16. <http://css.e-sign.cz/>
17. <http://www.xhtml.euweb.cz/uvodni.php>
18. <http://cssvault.com/>
19. <http://www.builder.cz/>
20. <http://www.webtip.cz/>
21. <http://pristupnost.nawebu.cz/>
22. [http://www.dithered.com/css\\_filters/css\\_only/index.php](http://www.dithered.com/css_filters/css_only/index.php)
23. <http://css.maxdesign.com.au/>

## Seznam tabulek a vyobrazení

Obrázek 1 - šířka boxu podle standardů .....	28
Obrázek 2 - šířka boxu podle IE.....	28
Obrázek 3 - www.seznam.cz .....	41
Obrázek 4 - Page .....	42
Obrázek 5 - Header.....	42
Obrázek 6 - Search .....	42
Obrázek 7 - Content .....	43
Obrázek 8 - Footer.....	43
Obrázek 9 - Content .....	44
Obrázek 10 – Main .....	44
Obrázek 11 - Right .....	45
Obrázek 12 - Right - Headlines.....	45
Obrázek 13 - Bottom .....	45
Obrázek 14 - Bottom – Katalog .....	45