**Продвинутая CSS-верстка: шаг за шагом**

Rogelio Vizcaino Lizaola, Andy King

Задача: воспроизвести с помощью CSS основную страницу WebReference. Инструменты: CSS и много свободного времени.

Kwon Ekstrom предложил свою CSS версию главной страницы WebReference. Rogelio Lizaolo улучшил ее. Через несколько месяцев напряженной совместной работы финальный вариант на CSS полностью повторял прежний, табличный. В процессе мы обнаружили массу ошибок в NN и IE и нашли несколько элегантных уловок для их обхода. Ниже следует наша пошаговая инструкция, с помощью которой вы можете повторить наш подвиг.

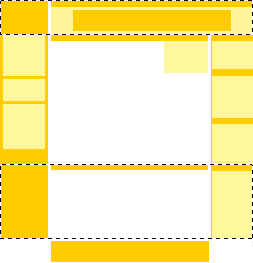
**Вступление**

В статье Эволюция домашней странички Andy King бросил вызов читателям, предложив им воспроизвести табличную верстку WebRef с помощью CSS. Kwon Ekstrom предложил решение, которое работало, но только в ограниченном наборе броузеров. После исследования его проекта я нашел в нем некоторые несообразности, которые попытался решить. Несколько месяцев Andy, Kwon и я не покладая рук трудились над дизайном WebRef.

Основной нашей целью было создать CSS разметку, которая бы полностью повторяла табличный дизайн WebReference.com, и прилично бы выглядела при любых размерах окна и шрифта.

Работа велась с прицелом на пятое поколение броузеров, в качестве платформы предполагались Windows и Macintosh. Мы намеревались использовать одни и те же стилевые таблицы для всех броузеров. Это ограничение усложнило нашу задачу, и сделало финальный код куда сложнее и запутаннее, чем если бы мы ориентировались только на броузеры с корректной поддержкой CSS-2 (например, Netscape 6). Понятно, что из-за проблем с совместимостью мы не могли использовать весь спектр возможностей CSS.

**Условия задачи**



Весной 2001 года основная страница WebReference состояла из шести блоков (в основе верстки лежали три таблицы):

Верхний навигационный блок с пятью текстовыми ссылками и формой для поиска

Рекламный блок, содержащий стандартный баннер

Левый навигационный блок с логотипом WebReference и текстовыми ссылками

Правый навигационный блок, содержащий текстовые ссылки, отображающие структуру сайта

Центральная область, содержащая основной контент

Цветной блок внутри центральной области, содержащий  "Совет дня"

Возможно ли воспроизвести эту разметку с помощью CSS? Разумеется. И мы расскажем, как это сделать.

Если не принимать в расчет верхний навигационный блок, рекламный блок и "Совет дня", то мы имеем дело с версткой в три колонки. Так как центральная колонка будет содержать контент, то от ее высоты будет зависеть высота всего документа. Понятно, что высота всех трех колонок должна быть одинакова. Обратите также внимание на отступы шириной 9px слева и справа от центральной колонки.

**Верстка в три колонки – как это делается**

Для начала, создадим простую трехколоночную верстку с промежутками между колонками, где высоту всего документа будет задавать центральная колонка. Начнем с четырех слоев и соответствующих им стилевых таблиц.

1. Сперва зададим нужные нам поля и отступы для тела документа. Обратите внимание, что мы задаем для него нулевой отступ, потому что в Opera он по умолчанию не равен нулю. Также мы задаем #FFF в качестве цвета фона, так как в Netscape 6 для Macintosh фон по умолчанию серый.

body {

margin:9px 9px 0 9px;

padding:0;

background:#FFF;

}

2. Теперь мы определим наш первый DIV (назовем его "level0"). Для него мы только лишь укажем цвет фона – это будет цвет нашей левой колонки. Взгляните

#level0 {

background:#FC0;

}

3. Вложим второй слой внутрь нашего level0 и назовем его "level1".

#level1 {

margin-left:143px;

padding-left:9px;

background:#FFF;

}

В этом новом слое мы указали место для левой колонки и отступ от нее. Цвет отступа задается цветом фона для слоя. Взгляните.

Вложим в "level1" слой под названием "level2"

#level2 {

background:#FFF3AC;

}

С помощью этого слоя мы задаем цвет фона для правой колонки. Теперь левая колонка и отступ отчетливо видны. Взгляните.

Для правой колонки мы по уже описанной методике добавляем в слой "level2" слой "level3"

#level3 {

margin-right:143px;

padding-right:9px;

background:#FFF;

}

#main {

background:#CCC;

}

Слой "main" мы добавили исключительно для того, чтобы сделать цвет фона для центральной колонки отличным от цвета других блоков и отступов между ними. Взгляните.

Добавляем шапку и рекламный блок

Мы пока еще не углублялись в тонкости разметки для боковых колонок, лишь визуально их отделили, но уже сейчас понятна методика по отображению трех колонок одной высоты. Перед тем как двигаться дальше, добавим  в разметку верхний навигационный блок и место для рекламы. Мы вложим слой "topBar" в "level1" и перед слоем "level2", а в "topBar" поместим слой "advBar". Взгляните.

...

#topBar {

background:#FC0;

}

#advBar {

background:#FFF3AC;

}

</style>

...

<body>

<div id="level0">

<div id="level1">

<div id="topBar">

This is the top navigation bar.

<div id="advBar">

This is the advertisement bar.

</div>

</div>

<div id="level2">

<div id="level3">

<div id="main">

The code you are reading.

</div>

</div>

</div>

</div>

</div>

</body>

...

"Совет дня"

Теперь у нас есть три блока, куда мы можем поместить любое содержимое, основная область, верхний навигационный блок и рекламный блок. Каждый из этих элементов принимает высоту содержимого, но ширина у них фиксированная. Давайте теперь создадим блок, содержащий "Совет дня". Сделаем его плавающим, чтобы содержимое центральной области его огибало. Вложим в "main" слой "tipDay", и зададим для него перемещение вправо. Также укажем для него некоторую ширину (для плавающих элементов ширина обязательно должна быть задана). Взгляните.

#tipDay {

float:right;

width:175px;

background:#FFF3AC;

}

Что с контентом?

Теперь давайте решим, как именно мы будем заполнять блоки содержимым. Самым естественным решением было бы использовать свойство float, но из-за проблем с отображением плавающих блоков в Opera и IE нам придется использовать абсолютное позиционирование. Ошибки в IE небольшие, и ими можно было бы пренебречь, но, к сожалению, с проблемами в Opera нам справиться не удалось. По этим ссылкам расположены описания возникших ошибок: одна ошибка в Opera, другая ошибка в Opera, ошибка в IE.

Заполняем боковые блоки

Начнем с левого навигационного блока – добавим к нему слой "lftBar". Так как мы решили использовать абсолютное позиционирование, то у нас теперь есть относительная свобода в выборе места для кода, определяющего слой. Его можно поставить внутрь "level0", до или после "level1" или даже на один уровень вложенности со слоем "level0" – это не повлияет на его отображение – если, конечно, мы выберем правильные координаты.

Мы попытались вложить "lftBar" в "level0", указав для "lftBar" нулевые координаты top и left, а для "level0" – относительное позиционирование (position:relative), но в IE5.5 слои почему-то перекрывались. Поэтому мы решили установить для "lftBar" абсолютные координаты относительно тела документа. Взгляните. А также – одна ошибка в IE, другая ошибка в IE.

#lftBar {

position:absolute;

width:143px;

top:9px;

left:9px;

}

Для правого навигационного блока мы использовали слой "rgtBar". Нам было важно, что бы его верхний край совпал с верхним краем слоя "main", что содержит основной контент. Поэтому мы решили вложить "rgtBar" в "level2", до или после "level3", установив для "level2" относительное позиционирование, а для "rgtBar" – нулевые координаты top и left и ширину, равную правому полю слоя "level3". Взгляните.

#level2 {

background:#FFF3AC;

position:relative;

}

...

#rgtBar {

position:absolute;

width:143px;

top:0;

right:0;

}

**Расставляем по местам**

С нашей версией было две проблемы в IE для Windows. Первая проблема состояла в том, что блок с «Советом дня» исчезал, хотя содержимое блока main его исправно огибало. Эту проблему мы разрешили, установив для слоя "tipDay" относительное позиционирование (такая установка привела к повышению z-index и слой автоматически разместился поверх остальных).

**Странности наследования: "Auto" против "Inherit"**

Второй проблемой в IE/Windows было то, что правый навигационный блок отодвигался максимально вправо. Для решения проблемы нам пришлось явно указать ширину слоя "level2" (в который вложен "rgtBar"). IE не понимает декларацию ‘auto’, но зато понимает ‘inherit’ (хотя и довольно-таки странным образом), и это можно использовать для наших целей. Мы установили  для "level0" ширину в 100%, а в качестве ширины для "level2" указали ‘inherit’. Фокус в том, что в IE "level2" наследует значение ширины от "level0" (100%), в то время как в прочих броузерах – происходит наследование от "level1" (‘auto’) (а укажи мы для "level2" ширину в 100%, то у нас возникли бы проблемы с отображением в IE 5.5+, который понимает 100%, как ширину всего окна броузера). Эту ошибку CSS в IE5 вы, кстати, вполне можете использовать для решения сходных проблем – подробности смотрите в разделе использованных приемов. Взгляните

#level0 {

background:#FC0;

width:100%; # - New, outermost DIV

}

#level2 {

background:#FFF3AC;

position:relative;

width:inherit; # - New, inherits 100% from level0 in IE5

# inherits auto from level1 in all others

}

#tipDay{

float:right;

width:175px;

background:#FFF3AC;

position:relative; # - New, forces z-index so this div displays in IE

}

Очищаем стиль

Теперь самое время провести генеральную уборку. Для начала избавимся от слоя "level3" и передадим все его стилевые декларации слою "main". Добавим правому блоку границу с помощью свойства border-right слоя "main". Также сделаем два заголовка для основной области и поставим в правый блок несколько тестовых подзаголовков. Внизу правого блока добавим подзаголовок "A\_Long\_String", чтобы продемонстрировать проблемы, возникающие при использовании больших шрифтов. Этот подзаголовок нельзя разбить на несколько линий, и он может раздвинуть блок, если превысит его по ширине. Взгляните.

#main { # Where the main content goes

margin-right:143px;

padding-right:9px;

background:#FFF;

}

#main h1 { # Where our main section headers go

margin:0;

padding-left:.3em;

font:1.25em Verdana,Helvetica,Arial;

color:#609;

background:#FC0;

}

#main dt { # Bump up the main content links

font-weight:bold;

font-size:120%;

margin-top:.8em

}

#rgtBar h3 { # Subheads for right nav bar, (newsletter, experts...)

margin:0;

padding:3px;

background:#FC0;

font-weight:bold;

font-size:1em;

text-align:center;

}

Размер фиксированный против размера относительного

Чтобы предотвратить превышение длинным подзаголовком ширины блока, мы изменим некоторые декларации для "rgtBar" и "main", переписав их в относительных единицах ("em"). Прежде мы задавали ширину в 143 px. Это составляет примерно 9em, если размер шрифта в броузере равен стандартным двенадцати пунктам (двенадцать пунктов – это16px). Поэтому мы задаем для "rgtBar" и "main" поля и ширину не в 143px, а в 9em. Теперь, благодаря относительным единицам, правый блок будет менять свою ширину пропорционально установленному пользователем размеру шрифта. Взгляните.

#main {

margin-right:9em; # - Use variable widths rather than fixed, they are

more user-friendly. This also applies to font sizes.

padding-right:9px;

background:#FFF;}

...

#rgtBar {

position:absolute;

width:9em; # - likewise

top:0;

right:0;

}

Описанная методика решает большинство проблем с превышением размера содержимого над размером содержащего блока. Для блока "Совет дня" мы тоже задали ширину в относительных единицах, но для левого блока все же решили остановиться на единицах абсолютных, так как его содержимое не могло разрушить верстку.

**Плавающая форма поиска**

Теперь разместим форму поиска и текстовые ссылки в верхнем навигационном блоке. Форму мы определим как первого потомка "topBar" и укажем ей перемещение вправо. Мы уже выяснили, что для плавающих элементов следует явно задавать ширину, и наша форма не является исключением. Определим ей ширину 10.5em – благодаря относительным единицам элементы формы будут пропорционально изменять размер в зависимости от размера шрифта в броузере пользователя. Также мы зададим для формы выравнивание к правому краю (text-align:right), так как сама форма чуть шире, чем ее элементы. Но мы, очевидно, еще не закончили – взгляните, и вы поймете, какую проблему нам предстоит решить.

#topBar form {

float:right;

width:10.5em;

text-align:right;

}

Нам нужно установить в ноль все поля формы поиска, чтобы максимально уменьшить ее размер по вертикали. Для Netscape мы определим высоту строки в 0.7 em, тогда кнопка "Search" по высоте сравняется с полем ввода (на отображение в IE и Opera эта директива никак не повлияет). Двигаемся дальше. Форма частично накладывается на рекламный блок. Этого мы позволить не можем, так как в нем прямо под формой будет располагаться текстовая ссылка, поэтому определим для "advBar" свойство clear. Напоследок, установим для "topBar" ширину в 100%, дабы избежать ошибки в IE и укажем для поля ввода размер шрифта в .8 em, чтобы его размер изменялся соответственно размеру шрифта в броузере. Взгляните.

#advBar {

background:#FFF3AC;

clear:right; # - New, clear the search form

}

...

#topBar form { # The search form

float:right;

width:10.5em;

text-align:right;

margin:0;

line-height:.7em; # - Crucial!

}

#topBar input {

font-size:.8em;

}

**Добавляем текстовые ссылки**

Поместим текстовые ссылки на полагающееся им место. Эти пять ссылок разделены буллитом и заключены в тэг <B>, поэтому именно этот тэг должен стать вторым потомком блока "topBar". В IE 5.0 для Windows мы обнаружили странный эффект, из-за которого ссылки выравнивались к правому краю. Решить эту проблему можно было либо заключив ссылки в слой с выравниванием налево, либо сделав <B> блочным элементом, указав для него необходимое выравнивание. Мы предпочли второй способ. Также мы указали стиль для шрифта и необходимую высоту строки, чтобы ссылки были выровнены по вертикали. Взгляните.

#topBar b { # Top nav bar text links (sitemap, experts...)

display:block;

text-align:left;

font:bold .8em/1.7em Arial,Geneva,sans-serif;

}

**Добавляем содержимое и заголовок для "Совета дня"**

На этом этапе мы заменим тестовые заголовки типа "Это заголовок 1" и "Это заголовок 2" на реальные заголовки сайта WebReference – "WebReference.com: Dev the Web™" и "internet.com" соответственно. Еще мы добавим заголовок для блока "Совет дня", причем располагаться он будет в том же блоке, что и заголовок "WebReference.com: Dev the Web™". Для этого нам понадобится новый слой, назовем его "tipTit", который будет первым потомком слоя "main". Укажем для него перемещение и выравнивание к правому краю, ширину, стиль шрифта и высоту строки для вертикального выравнивания. Ширину определим в 7em (такую же, как и у блока "tipDay"). Чтобы избежать проблем в IE, которые описаны выше, позиционирование его сделаем относительным. Взгляните.

#tipTit {

float:right;

width:7em;

text-align:right;

position:relative;

font:.8em/1.8em Arial,Geneva,sans-serif;

background:red;

}

.note {

color:red

}

Дело сделано! Мы успешно воспроизвели основную страницу WebReference.com без единой таблицы. Нам осталось только добавить баннер на его законное место. Взгляните. Примечание: финальный вариант разметки еще не тестировался в IE6, но мы обязательно это сделаем в самом ближайшем будущем. (прим. переводчика: в IE6 страница отображается вполне корректно)

#advBar p{text-align:right;margin:0 0.5em;font-weight:bold;}

#advBar img{margin-bottom:1em}

**Дальнейшие планы**

Надеемся, что из этой статьи вы почерпнули некоторые идеи по использованию CSS для создания многоколоночной верстки. Для дальнейшего улучшения дизайна мы можем добавить в левую колонку дополнительные информационные или рекламные блоки. Для верхнего блока ([sitemap | experts .... search]) и центрального заголовка (WebReference: Dev the Web .... Tip of the Day]) мы можем применить методику, описанную в A List Apart (хотя наверняка возникнут проблемы с позиционированием формы поиска). Разумеется, любые изменения мы будем тестировать во всех основных броузерах.

Ниже кратко сказано о том, чему мы научились в процессе разработки нового дизайна.

**Основные приемы CSS**

Сохраняйте гибкость

По возможности используйте относительные величины (em или % для шрифтов, em для слоев)

Вкладывайте слои друг в друга

Эта техника помогает решить проблему с вариациями реализации блочной модели в разных броузерах. В IE5 поля, границы и отступы в одном блоке не очень-то хорошо уживаются друг с другом. Обязательно отделяйте декларации полей и границ от деклараций ширины. В неверной блочной модели, реализованной в IE5/Windows, поля и границы не прибавляются к общей ширине блока, тогда как W3C с полной определенностью говорит о том, что ширина блока складывается из ширины его содержимого плюс ширина полей и границ. Заметим, что IE5/Mac и броузеры Gecko корректно отображают блочную модель. Поэтому если вам необходимо задать ширину слоя, который содержит поля или границы, разделите его на два, один из которых будет вложен в другой.

Не используйте "nowrap" для слоев

В Netscape и Opera могут возникнуть проблемы при больших размерах шрифта

В IE 5.x есть ошибка CSS, которой можно воспользоваться

IE 5.x интерпретирует комментарии, хотя должен их игнорировать. Это можно использовать для создания некоего подобия "if-then", чтобы обойти в нем некоторые ошибки CSS. Мы использовали этот прием, но потом перешли на методику с использованием свойства inherit. В IE6 этот прием не работает.

#fontChanger {font-size:.8em;//font-size:1em;}

Следите за шириной слоев и таблиц

Установка ширины в 100% может привести к тому, что ширина такого слоя в IE не совпадет с шириной слоя, для которого ширина задана как auto (по умолчанию). Кроме того, для таблицы с width=100% IE5.5 установит ширину равную ширине всего экрана, а вовсе не блока, в который эта таблица вложена. Во избежание вы можете использовать //width:100% для внешнего блока и width:inherit для самой таблицы.

#topBar{background:#FC0;//width:100%;}

#topBar form{display:inline;}

table.c{font:.8em/1em Arial,Geneva,sans-serif;width:100%;//width:inherit;}

В IE некорректно работает механизм наследования, поэтому таблица наследует ширину от ближайшего родителя, для которого ширина указана (в абсолютных единицах, относительных единицах или значением inherit).

У плавающих блоков могут возникать проблемы с директивой clear

Особенно у форм. Во избежание можно заключать блоки в дополнительные контейнеры и указывать clear для них. Хороша также методика, описанная в A List Apart.

Не используйте фиксированную высоту

Броузеры пятых и шестых версий ее просто не поддерживают, равно как и min-height. А IE 5.5 к тому же не понимает min-width, max-width и max-height.

В Opera и Netscape задавайте для форм нулевой отступ (margin:0)

Особенно в плавающих блоках.

Если вы хотите задать ненулевые отступы сверху (по бокам), обязательно указывайте высоту (ширину) для родительского блока.

Opera расходится с W3C в толковании спецификации CSS, поэтому для корректного отображения отступов вам придется задать высоту и ширину не только для родителя, но и для потомка. Отметим, что в Opera и IE 5.5 систематически возникают ошибки, если у родителя тоже заданы отступы.