

# Семантическая верстка

Семантика (фр. *sémantique* от др.-греч. *σημαντικός* — обозначающий) — наука о понимании определенных знаков, последовательностей символов и других условных обозначений

# Цели создания семантического кода

- Для разработчика - Упрощение работы по созданию и ведению ресурса.
- Для программ-обработчиков – извлечение информации из HTML-страницы

# Читаемый HTML-код составляет

- Семантически правильное использование элементов (согласно спецификации).
- Последовательное и логичное именованение значений атрибутов.
- Наличие отступов, формирующих дерево документа.

# Самодокументируемый код

- При именовании элементов в соответствии с их содержимым или назначением код начинает звучать – его можно легко читать по внятно именованным ключевым элементам
- это упрощает разработку и поддержку сайтов,
- является отличной основой для единого стиля верстки для команды.

# Программы-обработчики

- Создатели адресных книг
- Поисковые роботы
- Программы для чтения RSS лент
- Голосовые браузеры
- Google Maps API
- и др.

# Уровни семантики

- Применение HTML-элементов в точном соответствии со спецификацией
- Именованние элементов – классов, ID и др.
- Создание смысловых связей между семантическими единицами в документе. Микроформаты.

# СМЫСЛОВЫЕ СВЯЗИ

Семантика в верстке — совокупность смысловых отношений, возникающих в документе

```
<dl>
```

```
  <dt>Семантика</dt>
```

```
    <dd>совокупность смысловых отношений.
```

```
  </dd>
```

```
</dl>
```

Семантические связи в документе существуют только между HTML-элементами, а не между отдельными словами или частями текста

# Первый уровень семантики

## POSH

- POSH (Plain Old Semantic HTML) — простой старый семантический HTML — использование HTML-элементов по их назначению
- Все особенности применения элемента и смысловые связи, которые образует с остальными элементами, расписаны в спецификации.



# **Использование языка в соответствии с его**

## **предназначением - POSH**

- Концепция POSH основывается на использовании встроенных возможностей языка.
- Прежде чем работать с HTML, неплохо было бы изучить его - использование вещей по их прямому назначению — это всегда наиболее оптимально.

# Определения заголовков в HTML DTD

```
<!ENTITY % heading "H1|H2|H3|H4|H5|H6">
```

```
<!--There are six levels of headings from H1 (the  
most important) to H6 (the least important).-->
```

```
<!ELEMENT (%heading;) - - (%inline;)* -- heading  
-->
```

```
<!ATTLIST (%heading;) %attrs; -- %coreattrs,  
%i18n, %events -->
```

# Пример – имя, адрес

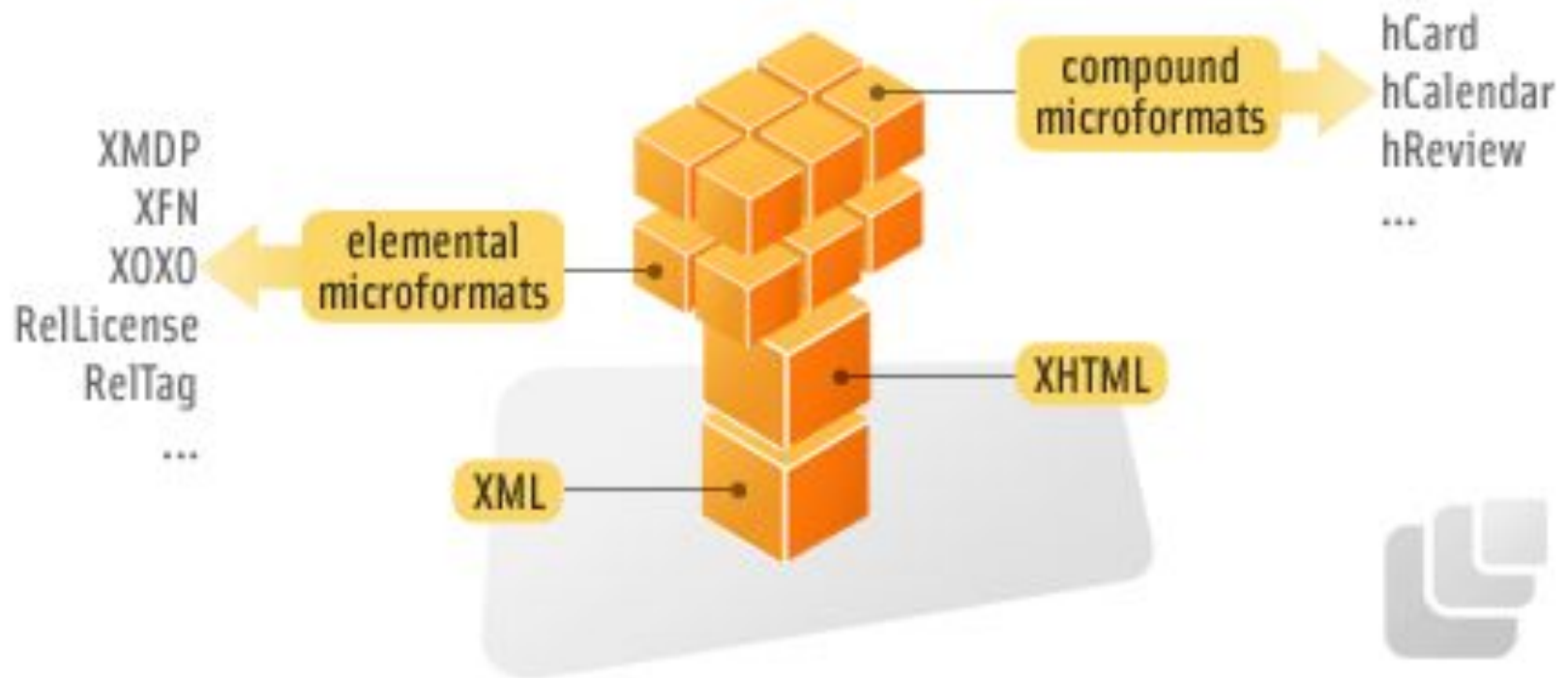
Алексей Рыбаков, alex@example.com

1. `<h3>Алексей Рыбаков</h3>`  
`<address>alex@example.com</address>`
2. `<h3 class="name">Алексей Рыбаков</h3>`  
`<address class="email">`  
[alex@example.com](mailto:alex@example.com)  
`</address>`

# Применение hCard

```
<div class="vcard">  
  <h3 class="fn n">  
    <span class="given-name">Алексей</span>  
    <span class="family-name">Рыбаков</span>  
  </h3>  
  <address class="email">alex@example.com  
</address>  
</div>
```

# μF - базовая идея Semantic Web



# Микроразметка

Микроразметка — разметка данных на странице и на сайте, которая используется для того, чтобы программы-обработчики

- распознавали контент сайта;
- правильно отображали страницы;
- собирали информацию для
  - вывода на другие устройства ;
  - голосового воспроизведения ;
  - формирования анонсов, RSS ;
  - создания карт и других изображений;
  - формирования записных книжек («Контакты»)и др.

# Микроформат

- **Микроформат** ( $\mu$ F или uF) — способ семантической разметки сведений о сущностях (событиях, организациях, людях, товарах и пр.) на веб-страницах с использованием стандартных элементов (X)HTML.
- Программы-обработчики могут извлечь из такой страницы структурированную информацию, следуя определенным соглашениям.
- Человек может воспринимать страницу с микроформатом как обычную веб-страницу.

# Цели и преимущества $\mu F$

- Сделать контент более заметным и понятным для поисковых машин
- извлекать и интерпретировать информацию из документа на основе формализованного синтаксиса.
- использовать известные стандарты для передачи большего количества семантической информации. Чтобы создать блоки кода с определенной семантикой, используются существующие элементы XHTML: address, cite, blockquote... и атрибуты: rel, rev, title...
- браузеры, плагины, веб-сервисы смогут ассоциировать семантически размеченные данные с различными приложениями.



# Актуальные синтаксисы

- Microdata
- RDFa
- JSON-LD

# Microdata

- впервые появился в HTML-5;
- поисковые системы используют его для вывода контента сайта в результатах поиска;
- синтаксис Microdata Google использует для создания словаря Schema.org.

# Результат работы словаря Shema.org и грамотной разметки

[Apple iPhone 6 32GB Space Gray купить, iPhone 6 32GB ... - Алло](#)

[allo.ua](#) > ... > [Apple iPhone 6 32GB Space Gray](#) ▼

★★★★★ Рейтинг: 5 - 1 отзыв

ALLO.ua ►►► Купить ☆ Apple iPhone 6 32GB Space Gray ☆ по лучшей цене ✓ Качество ✓

Отзывы ✓ ➔ Быстрая доставка по всей Украине ☎ Звони ...

[Apple iPhone 6 16Gb. Цены в г. Киев. Купить - Price.ua](#)

[price.ua](#) > Мобильная связь > Мобильные телефоны, смартфоны ▼

★★★★★ Рейтинг: 4,2 - 136 голосов

Цены на iPhone 6 16Gb в г. Киев. Описание iPhone 6 16Gb, отзывы и динамика цен. Фото. Купить

Apple iPhone 6 16Gb в г. Киев. Рейтинг на Price.ua.

[Купить iphone 6 в Одессе - iphone 6 цена - "Mobile Planet"](#)

[www.mobileplanet.ua](#) > Apple > Apple iPhone > Apple ▼

Ищите где выгодно купить iphone 6? Интернет-магазин "Mobile Planet" предлагает цену на айфон 6 ниже чем где-нибудь.

[Apple iPhone 6 16GB \(Gold\) купить в интернет-магазине – цены ...](#)

[hotline.ua/mobile-mobilnye-telefony-i-smartfony/apple-iphone-6-16gb-gold/](#) ▼

★★★★★ Рейтинг: 4,9 - 2 отзыва - От 6 999,00 грн. до 16 440,00 грн.

10 сент. 2014 г. - Apple iPhone 6 16GB (Gold): сравнить цены в интернет-магазинах Украины, характеристики, отзывы покупателей, купить в кредит.

# Это информационная статья

- `<article itemscope itemtype="http://schema.org/Article">`
- `<h1 itemprop="name">Микроразметка</h1>`
- `<meta itemprop="inLanguage" content="ru" />`
- `<section itemprop="articleBody">Давайте поговорим о микроразметке</section>`
- `<section itemprop="author" itemscope itemtype="http://schema.org/Person">`
- `<span itemprop="additionalName">Представители SEO</span>`
- `</section>`
- `</article>`

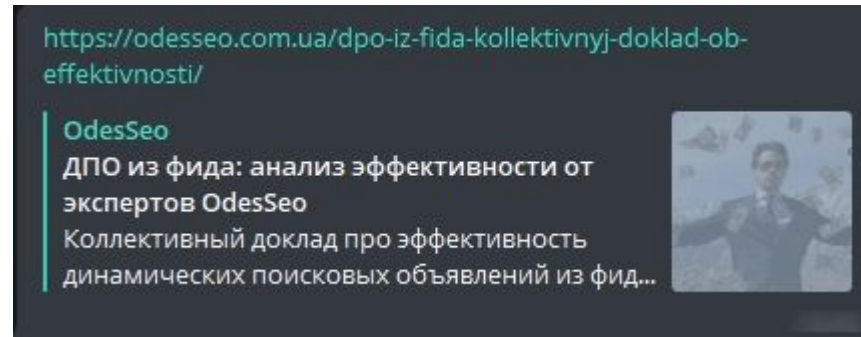
# RDFa

- синтаксис разработали специалисты из W3C в 2004;
- реже используется из-за сложности реализации;
- чаще всего используется в словаре Open Graph.

# Основные атрибуты словаря Open Graph

- «og:locale» — указывает на язык контента и локацию сайта
- «og:type» — указывает тип контента на странице (статья, новость, обзор и т.д.)
- «og:title» — указывает метатег title
- «og:description» — указывает метатег description
- «og:url» — URL страницы
- «og:image» — указываем изображение статьи, обзора и т.д.
- «og:site\_name» — имя сайта.

# Ссылка в социальной сети



Выполнена средствами словаря **OpenGraph**

# Разметки статьи в RDFa

- `<div vocab="http://schema.org/" typeof="Article">`
- `<span property="name">RDFa</span>`
- `by <span property="author">SEO-специлист</span>`
- Эту статью твитнули 20 раз и откомментировало 10 человек
- `<div property="interactionStatistic" typeof="InteractionCounter">`
- `<div property="interactionService" itemscope itemid="http://www.twitter.com" itemtype="http://schema.org/Website">`
- `<meta property="name" content="Twitter" />`
- `</div>`
- `<meta property="interactionType" content="http://schema.org/ShareAction"/>`
- `<meta property="userInteractionCount" content="20" />`
- `</div>`
- `<div property="interactionStatistic" typeof="InteractionCounter">`
- `<meta property="interactionType" content="http://schema.org/CommentAction"/>`
- `</div>`
- `</div>`



# JSON-LD

- самый молодой и актуальный формат;
- внедряется не в код страницы, а в тег `<script>`;
- представляет из себя набор пар “ключ-значение”;
- предназначен для понимания не только поисковыми ботам, но и человеком.

# Статья размещена в JSON-LD

- <script type="application/ld+json">
- {
- "@context": "http://schema.org",
- "@type": "Article",
- "author": "SEO-специалист",
- "interactionStatistic": [
- {
- "@type": "InteractionCounter",
- "interactionService": {
- "@type": "Website",
- "name": "Twitter",
- "url": "http://www.twitter.com"
- },
- "interactionType": "http://schema.org/ShareAction",
- "userInteractionCount": "20"
- },
- {
- "@type": "InteractionCounter",
- "interactionType": "http://schema.org/CommentAction",
- "userInteractionCount": "10"
- }
- ],
- "name": "JSON-LD"
- }
- </script>



# Лучше

```
<ul id="menu">
```

```
<li><a href="#">Колбаса</a></li>
```

```
<li><a href="#">Макароны</a></li>
```

```
<li><a href="#">Тушенка</a></li>
```

```
</ul>
```

- Колбаса 100 гр. 250 руб.
- Макароны 100 гр. 18 руб.
- Тушенка 100 гр. 75 руб.

<table>

<tr>

<td>Колбаса</td>

<td>100 гр.</td>

<td>250 руб.</td>

</tr>

<tr>

<td>Макароны</td>

<td>100 гр.</td>

<td>18 руб.</td>

</tr>

<tr>

<td>Тушенка</td>

<td>100 гр.</td>

<td>75 руб.</td>

</tr>

</table>

## Вариант развития задачи

# Второй уровень семантики - именование элементов

Второй смысловой уровень создается двумя способами именования атрибутов элементов:

- `id` — это имя уникального элемента, подходит для *структурной* разметки;
- `class` — общее имя множества элементов и их общая семантическая роль.

```
<ol id="comments">  
  <li>Первый</li>  
  <li>Второй</li>  
</ol>
```

## Пример с id

```
<body>  
  <div id="header"></div>  
  <div id="content"></div>  
  <div id="footer"></div>  
</body>
```

# Пример с for вместо id

```
<label for="full-name">
```

Полное имя

```
</label>
```

```
<input type="text" id="full-name"/>
```

id применяется для создания связи ярлык-поле при помощи атрибута for элемента

```
<label>
```



# Примеры с class

- class — это общее имя множества элементов и их общая семантическая роль.
- Множественные классы для одного элемента позволяют удобно комбинировать не только стили, но и смысловые значения нескольких имен.
- `<a href=http://yandex.ru/ class="fn org url">`  
Яндекс  
`</a>`

# Оформление ссылок на RSS-ПОТОК

```
<div id="header">  
  <a href="#" class="rss">  
    RSS  
  </a>  
</div>  
  
<div id="footer">  
  <a href="#" class="rss">  
    RSS  
  </a>  
</div>
```

```
.rss {  
    padding-left:20px;  
    background:url (rss.png)  
    no-repeat;  
}  
  
#header .rss {  
    color:#C00;    }  
  
#footer .rss {  
    color:#069;  
}
```

ссылки шапки и подвала  
отличаются цветом

# Третий уровень семантики

Третий уровень семантики составляет - создание смысловых связей между семантическими единицами.

Единственным примером этого механизма на данный момент (HTML4) являются микроформаты ( $\mu$ f).

# Разметка имени пользователя и иконки рядом

```
<span class='ljuser' lj:user='Kuteev' style='white-space:
nowrap;'>
  <a href='#'>
    <img src='#' alt='[info]' width='17' height='17'
      style='vertical-align: bottom;
      border: 0; padding-right: 1px;' />
  </a>
  <a href='#'>
    <b>Kuteev</b>
  </a>
</span>
```

Плохо

# Имя пользователя и иконка рядом с помощью

## hCard

```
<span class="ljuser vcard">  
  <a href="#" target="_blank"  
    class="ico" title="Профиль">  
      
  </a>  
  <a href="#" target="_blank"  
    class="fn nickname url" title="Журнал">  
    kuteev  
  </a>  
</span>
```

# Разметка информации о компании с hCard и Google Maps API

```
<div class="vcard">
  <h1 class="fn org">Яндекс</h1>
  <dl>
    <dt>Адрес:</dt>
    <dd class="adr">
      <span class="postal-code">111033</span>,
      <span class="locality">Москва</span>,
      <span class="street-address"> ул. Самокатная, дом 1</span>
    </dd>
  </dl>
  <div id="map" class="geo">
    <abbr class="latitude" title="55.7582">N 55° 75.82</abbr>
    <abbr class="longitude" title="37.6786">W 37° 67.86 </abbr>
  </div>
</div>
```

# Преимущества семантической верстки

- Для разработчика и для конечного пользователя - повышение доступности сайта для альтернативных устройств просмотра:
  - мобильные
  - и голосовые браузеры,
  - принтеры,
  - перемещения по заголовкам,
  - поисковые роботы;

# Для мобильных браузеров

- Код, не переполненный презентационной разметкой, компактнее:

Меньше кода —> легче страницы —> быстрее грузятся, меньше требуется оперативной памяти на стороне пользователя, меньше трафика, меньший объем баз данных. **Сайт становится быстрее и менее затратным.**

- не в полной мере поддерживается CSS и поэтому ориентируются в основном на HTML код, отображая его на экране согласно используемым тегам.



# Для голосовых браузеров

- важны теги и их атрибуты, чтобы произнести правильно и с нужной интонацией содержимое, или наоборот не произнести лишнего.
  - Голосовой браузер правильно акцентирует каждый уровень заголовков,
  - последовательно перечисляет общие элементы списка и пр.

# Для печатающего устройства

- даже без дополнительного CSS напечатают информацию качественней (ближе к дизайну),
- создание версии для печати представляет собой либо
  - простое отключение стилей,
  - минимальное оформление документа, который уже и так готов для удобного чтения.

# Для специальных устройств и плагинов

- Существуют встроенные функции, расширения и плагины, которые позволяют быстро перемещаться по документу, например:
  - по заголовкам у Opera;
  - расширения (vimperator на firefox);
  - плагины ([Key Binder](#) - удобная работа с Chrome без мыши).
- устройства для быстрого перемещения.

# Для поисковых машин

- код поддается гораздо лучшему анализу — чище, логичнее (четко видно где заголовки, где навигация, содержимое и пр);
- увеличение веса ключевых слов в заголовке `<h1>` за счет уменьшения «мусорного» кода на странице;
- поддержка микроформатов браузерами и специальными плагинами для более точного и подробного анализа документов

# Недостатки

- Код не проходит валидацию, поскольку микроформаты не включены в спецификацию или другие нормативные документы консорциума
- **ОДНАКО**, каждая поисковая система предоставляет веб-мастерам свой валидатор микроразметки для проверки правильности оформления данных
  - Сервис от ПС Google — Проверка структурированных данных
  - Сервис от ПС Яндекс — Валидатор микроразметки

# Семантическая паутина

- **Семантическая паутина** ( *semantic web*) — это общедоступная глобальная семантическая сеть, формируемая на базе Всемирной паутины путем стандартизации представления информации в виде, пригодном для машинной обработки.

# Полезные Ссылки

- <http://www.xiper.net/learn/tegofenshuj/about-semantic.html> - Евгений Рыжков. Семантический HTML — рекомендация с большими выгодами
- [http://microformats.org/wiki/Main\\_Page](http://microformats.org/wiki/Main_Page) - Microformats Wiki  
или [http://microformats.org/wiki/Main\\_Page-ru](http://microformats.org/wiki/Main_Page-ru) - Main Page-ru
- <http://web.archive.org/web/20080730163823/http://ay4.getalime.ru/blog/2008/04/24/semantics-shmemantics/> - Sans стыда, sans совести. Семантика-шмемантика
- <https://freshit.net/chto-takoe-snippet/> - Что такое сниппет? Как сделать сниппет для Гугла?
- <https://zametkinapolyah.ru/onlajn-servisy/rss-lenta/chto-takoe-rss-lenta-i-potok-programmy-dlya-chteniya-rss-lent-rss-reader-ikonki-i-knopki-rss-dlya-sajta-kak-ustroen-format-rss.html> - Что такое RSS лента и поток. Программы для чтения RSS лент — RSS reader. Иконки и кнопки RSS для сайта. Как устроен формат RSS
- <https://zaurmag.ru/priemy-verstki-html-css/osnovy-semanticheskoy-verstki-na-html5.html> - Основы семантической верстки на HTML5

# Еще статьи на тему

- <https://odesseo.com.ua/chto-takoe-mikrorazmetka/> - Что такое микроразметка?
- <https://medium.com/@stasonmars/%D1%81%D0%B5%D0%BA%D1%80%D0%B5%D1%82%D1%8B-%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F-%D1%81%D0%B5%D0%BC%D0%B0%D0%BD%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B8%CC%86-%D0%B2%D0%B5%D1%80%D1%81%D1%82%D0%BA%D0%B8-%D0%B2-html5-c7cd5e6f1ebb> - Что такое семантический HTML5?
- <https://habrahabr.ru/hub/microformats/> - Микроформаты - Способ семантической разметки на веб-страницах
- [http://professorweb.ru/my/html/html5/level1/1\\_9.php](http://professorweb.ru/my/html/html5/level1/1_9.php) - Семантическая разметка и микроформаты
- <https://www.w3.org/TR/html401/intro/sgmltut.html#h-3.3%20On%20SGML%20and%20HTML/3.3> - How to read the HTML DTD



# RDF - среда описания ресурса

- **Resource Description Framework** — это разработанная консорциумом Всемирной паутины модель для представления данных, в особенности — метаданных.
- RDF представляет *утверждения* о ресурсах в виде, пригодном для машинной обработки. RDF является частью концепции семантической паутины.

# Что есть ресурс в RDF?

- Ресурсом в RDF может быть любая сущность — как информационная (например, веб-сайт или изображение), так и неинформационная (например, человек, город или некое абстрактное понятие).
- С 10 февраля 2004 года RDF имеет статус стандарта W3C.

# Еще об RDF

- Утверждение, высказываемое о ресурсе, имеет вид «субъект — предикат — объект» и называется ***триплетом***.
- Для обозначения субъектов, отношений и объектов в RDF используются URI.
- Множество RDF-утверждений образует ориентированный граф, в котором вершинами являются субъекты и объекты, а ребра отображают отношения.
- RDF является не форматом файла, а только лишь абстрактной моделью данных, то есть описывает предлагаемую структуру, способы обработки и интерпретации данных. Для хранения и передачи информации, уложенной в модель RDF, существует целый ряд форматов записи.
- Для обработки RDF-данных предлагается реализовать языки запросов: SPARQL (стандарт W3C), RQL, RDQL.

# Форматы для RDF

- Для записи и передачи RDF используется несколько форматов, в том числе:
- RDF/XML — запись в виде XML-документа;
- RDF/JSON (англ.) — запись в виде JSON-данных;
- [RDFa](#) (*RDF in attributes*) — запись внутри атрибутов произвольного HTML- или XHTML-документа;
- N-Triples, [Turtle](#), N3 — компактные формы записи утверждений.