Семантические особенности семейства языков разметки

Р.С. Алиев, А.А. Копий

***Аннотация*— Языки разметки позволяют создавать гибкую форму представления информации. С их помощью стало возможным демонстрировать информацию в удобном человеку виде, что активно используется во всех сферах жизни. Хотя основные задачи этих языков заключаются в структурировании представления информации и облегчении ее восприятия, сами же языки разметки обладают рядом проблем, которые затрудняют восприятие их самих как человеком, так и программными средствами. В настоящей статье рассматриваются проблемы, характерные для современного представления семейства языков разметки. Так же исследуются существующие методы решения этих проблем. На основе исследования предлагаются улучшенные методы, составляющие базу нового оригинального подхода представления информации.**

***Ключевые слова*— язык разметки, семантика, семантическая верстка, интернет, представление информации.**

I. Введение

Современный мир сложно представить без сети Интернет, которая открыла доступ к непрерывно увеличивающимся объемам информации, таким образом, повлияв на все сферы жизни. Языки разметки используются для структурирования и корректного представления любого вида информации. Являясь инструментарием для выражения информации в удобной человеку форме, языки разметки отличаются строгой формализованностью, как и любые другие языки, отвечающие за взаимодействие компьютера и человека. И хотя наполнение семантикой тезауруса семейства языков разметки, произошедших от SGML, упрощает понимание исходного кода для человека, их синтаксис заставляет мысленно представлять структуру документов в виде дерева визуализации. Человеку же более удобно воспринимать естественные языки. Отсюда возникает необходимость пересмотра представления разметки и создание нового подхода, который был бы ориентирован в первую очередь на восприятие языка человеком.

Manuscript received Apr 30, 2017.

Aliev Rustam Suleyman ogly – candidate Sc. of Engineering, Associate Professor of the sub-department «Management and Informatics in Technical Systems» of Moscow State Technological University «STANKIN» (e-mail: r.s.o.aliev@gmail.com)

Copiy Anna Aleksandrovna – student of the sub-department «Management and Informatics in Technical Systems» of Moscow State Technological University «STANKIN» (e-mail: n-copiy@ya.ru)

II. Проблематика восприятия языка разметки

На сегодняшний день семейство языков, произошедших от SGML (Standart Generalized Markup Language), занимает доминирующие позиции в задачах представления информации. Наиболее часто используемыми языками этого семейства являются XML, HTML и XHTML, семантика и синтаксис которых были разработаны в конце прошлого столетия [1]. В числе ключевых особенностей данных языков можно выделить:

1) относительно компактный базовый тезаурус;
2) возможность добавления в тезаурус пользовательских литеральных последовательностей;
3) строго формализованная модель синтаксического анализа исходных кодов.

Несмотря на синтаксические различия, языки данного семейства основаны на базовом семантическом принципе: блочная структура кода использует модель контейнеров данных, каждый из которых помечается тегом, определяющим способ интерпретации данных.

Фундаментальным недостатком такой модели является низкая человекочитаемость (мера способности оператора-человека анализировать содержимое документа), обусловленная:

1) сложной структурой кода;
2) большим количеством специальных символов.

III. Современные методы улучшения восприятия кода

Мы оцениваем повышение человекочитаемости языков разметки как актуальную практическую задачу, поскольку отказ от учёта данной проблемы фактически нивелирует разницу между двоичным и текстовым представлениями документов.

В качестве очевидного пути решения данной задачи выделим:

1) компактификация исходного кода (уменьшение размера эквисемантических текстов в разных синтаксисах)
2) максимально возможное сближение семантики и её синтаксического выражения;
3) снижение количества специфических элементов тезауруса.

Рассмотрим подробнее решения, которыми ограничивается современное представление языков

разметки.

Для компактификации используются принципы декомпозиции кода. Наиболее распространенным инструментарием решения этой проблемы являются таблицы стилей CSS (Cascading Style Sheets) и использование языка преобразования XML-документов – XSLT (eXtensible Stylesheet Language Transformations) (листинг 1-2). Без подобной декомпозиции код перестает быть гибким, фрагменты кода часто дублируются (листинг 3).

A. ***Листинг 1.*** *Пример структуры кода на HTML с применением декомпозиции путем подключения файла «some.css».*

```
<HTML>
  <HEAD>
        <TITLE>Title</Title>
    <link rel="stylesheet"
href="some.css">
  </HEAD>
  <BODY>
    <H1>
      <img src="url.png" alt="something"
width="90%">
      <br>
      Hello, world!
      <br>
    </H1>
    <H2>
      We have been waiting for you!
      <br>
    </H2>
    <H1>
      <br>
      Another title.
      <br>
    </H1>
    <H2>
      First paragraph.
      <br>
    </H2>
    <H2>
      Second paragraph.
      <br>
    </H2>
    <H2>
      Third paragraph.
      <br>
    </H2>
  </BODY>
  </HTML>
```

B. ***Листинг 2.*** *Описание файла «some.css».*

```
H1 {
  text-align: center;
  size: 23px;
  color: black;
}

H2 {
  text-align: center;
  size: 5px;
  color: black;
}
```

C. ***Листинг 3.*** *Предст авление исходного кода (Лист инг 1) без элемент ов декомпозиции.*

```
<HTML>
  <HEAD>
        <TITLE>Title</Title>
  </HEAD>
  <BODY>
    <H1 align="center" size=23
color="black">
      <img src="url.png" alt="something"
width="90%">
      <br>
      Hello, world!
      <br>
    </H1>
    <H2 align="center" size=5
color="black">
      We have been waiting for you!
      <br>
    </H2>
    <H1 align="center" size=23
color="black">
      <br>
      Another title!
      <br>
    </H1>
    <H2 align="center" size=5
color="black">
      First paragraph.
      <br>
    </H2>
    <H2 align="center" size=5
color="black">
      Second paragraph.
      <br>
    </H2>
    <H2 align="center" size=5
color="black">
      Third paragraph.
      <br>
    </H2>
  </BODY>
</HTML>
```

Очевидная проблема данной концепции – увеличение порога вхождения. Для разработки и поддержки кода написанного с использованием композиции языков требуется больше усилий. Такое решение противоречит начальной цели языков XML, HTML и XHTML, которая заключалась в создании синтаксиса с низким порогом вхождения [2].

Для реализации не менее важной концепции уменьшения количества символов и литеральных последовательностей интуитивно не понятных человеку так же используются таблицы стилей CSS и язык преобразования XSLT.

В данном решении ответственность за адекватный семантики синтаксис перекладывается на разработчика. Разработчик в свою очередь не всегда заинтересован в изящности исходного кода, а тем более в использовании инструментария семантических разметок кода, которые требуют больших усилий и временных затрат [3]. Таким образом, разработчик может только увеличивать сложность восприятия исходного кода.

Второй метод уменьшения количества не очевидных единиц языка заключается в корректировки словаря, заменой тегов новыми, но с большим семантическим весом (рисунок 1-2) [4].

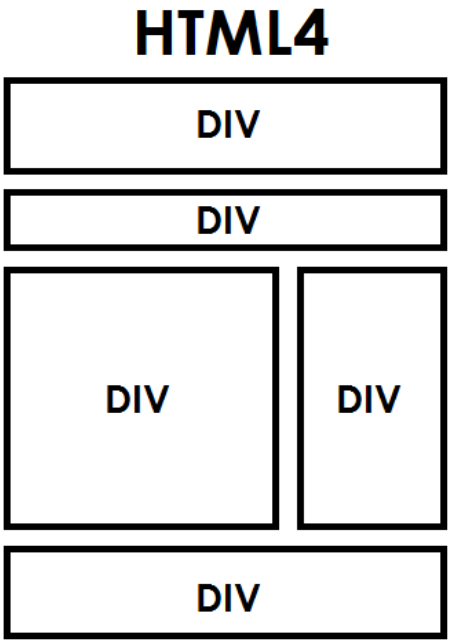

Рисунок 1. Структура HTML документа до семантической верстки.

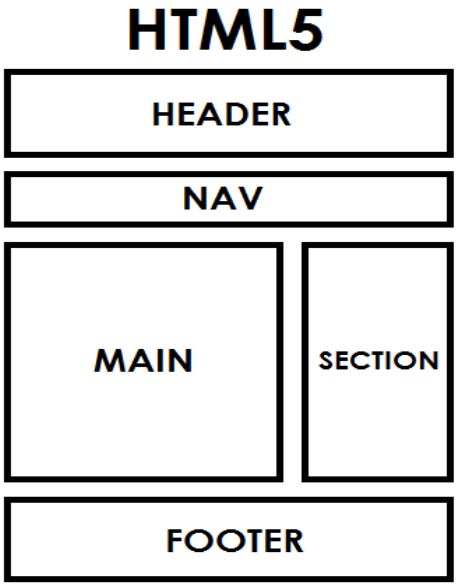

Рисунок 2. Структура HTML документа после семантической верстки.

Для совместимости со старыми стандартами кодирования сохраняется частичная поддержка и устаревших стандартов, что только усложняет словарь. Новые теги становятся эвфемизмами старых, а сложность восприятия словаря увеличивается вместе с его размером.

IV. Альтернативное представление разметки

Языки разметки предоставляют людям возможность работать с информацией в удобном и структурированном виде, однако сами не имеют удобного представления.

Авторы статьи полагают, что композиция структурных кодов носит деструктивный характер. Нами предлагается решение данных проблем в ином подходе – в смене парадигмы, которая подразумевает изменение структуры страницы и принципов ее обработки.

В первую очередь существующий способ представления исходного кода разметки отличается большой загруженностью (листинг 3). Формат кода, предлагаемый нами, отличается же легкостью восприятия (листинг 4), поскольку в нем отсутствует фундаментальная особенность языков разметки - лишняя загруженность семантикой, выражаемая закрывающими тегами, которые только повторяют название используемого тега, но не несут практического смысла.

Современные языки взаимодействия человека и компьютера демонстрируют стремление к максимальному упрощению восприятия семантики. Это происходит путем избавления от загруженности кода и добавления «синтаксического сахара» (конструкций языка, не влияющих на производительность, но упрощающих восприятие кода человеком) [5]. Более того, контроль над синтаксическими конструкциями, семантика которых заключается в разграничении некого участка кода, берут на себя современные среды разработки, путем форматирования и выделения некорректного синтаксиса. Эта стратегия избавляет оператора от необходимости дублирования названия в тегах и уменьшает время написания эквисемантических конструкций.

A. ***Листинг 4.*** *Альтернативная структура кода разметки без применения декомпозиции.*

```
[ TITLE "Title"]
[ "text-align=center; size=23"
  [ IMG "alt=something; width=90%;
src=url.png" ]
  "\nHello, world!\n"
]
[ "text-align=center; size=5"
  "We have been waiting for you!\n"
]
[ "text-align=center; size=23" "\nAnother
title.\n"
]
[ "text-align=center; size=5"
  "First paragraph.\n"
]
[ "text-align=center; size=5"
  "Second paragraph.\n"
]
[ "text-align=center; size=5"
  "Third paragraph.\n"
]
```

Так же мы предлагаем максимально перенести синтаксическую нагрузку кода на семантическую. Для этого необходимо изменить концепцию языка разметки, сделав его не интерпретируемым языком, а компилируемым. Таким образом, при описании структуры страницы ответственность за соответствие исходного кода документа документации ложится на разработчика. При использовании неправильного в рамках документации синтаксиса, код не будет проходить проверку компилятором.

Для декомпозиции языка и увеличения человекочитаемости исходного кода можно использовать принцип работы таблиц стилей CSS и языка преобразования XSLT, то есть выносить описание стилей и тегов в отдельный файл (листинг 5). Разница между предлагаемым подходом и уже существующим в том, что описание стилей и тегов осуществляется на том же языке, что и описание самой разметки документа. Таким образом, можно декомпозировать исходный код и использовать один компилятор для обработки всех исходных кодов. Кроме того, отсутствие требования изучать синтаксис стороннего языка не влияет на порог вхождения.

B. ***Листинг 5.*** *Альтернативная структура кода разметки с применением декомпозиции..*

```
[CENTER "text-align=center"]
[BLACK “color=black"]
[H1 "CENTER; size=23"; BLACK]
[H2 "CENTER; size=5"; BLACK]
[IMG_SOMETHING_TEMPLATE "alt=something;
width=90%"]
[IMG_SOMETHING_1
"IMG_SOMETHING_TEMPLATE; src=url.png"]


[ TITLE "Title"]
[ H1
  [ IMG_SOMETHING_1 ]
  "\nHello, world!\n"
]
[ H2
  "We have been waiting for you!\n"
]
[ H1
  "\nAnother title.\n"
]
[ H2
  "First paragraph.\n"
]
[ H2
  "Second paragraph.\n"
]
[ H2
  "Third paragraph.\n"
]
```

V. Заключение

Современный подход к представлению информации не решает актуальную проблему повышения человекочитаемости формализованных языков. Некоторые из методов компатификации исходного кода вызывают побочные эффекты, увеличивающие порог вхождения, что только ухудшает проблему восприятия исходного кода.

Нами был предложен подход, более эффективно решающий проблемы современных вариантов представления языка разметки, существенно облегчающий повышающий человекочитаемость и не увеличивающий порог вхождения в язык.

Предложенный нами синтаксис разметки приближен к синтаксису естественного языка, что положительно сказывается на восприятии его человеком. Следовательно, вопрос о необходимости дополнительного учета семантики отпадет, поскольку на приближенном к естественному языку проще выражать идею.

Библиография

[1] T. Berners-Lee, D. Connolly. Hypertext Markup Language - 2.0. RFC 1866 [Online]. Available: https://tools.ietf.org/html/rfc1866

[2] Решение задач на компьютерах [Text] : учеб. Пособие / А. А. Москвитин. - Новосибирск : СибГУТИ, 2006. - 158 с

[3] Л.В. Бизина, Р.С. Алиев. Выявление и анализ значимых объектов изображения. – Труды XVII научной конференции «Математическое моделирование и информатика», М.: «Станкин», 2015. 280с. В 2 т.: том 1, с. 56-58.

[4] I. Hickson, R. Berjon, S. Faulkner, T. Leithead, E. D. Navara, E. O'Connor, S. Pfeiffer. HTML5. A vocabulary and associated APIs for HTML and XHTML. W3C Recommendation 28 October 2014 [Online]. Available: https://www.w3.org/TR/html5/

[5] David A. Watt, William Findlay. Programming language design concepts. John Wiley & Sons Ltd, 2004, 492c.

Semantic features of the markup languages family

R.S. Aliev, A.A. Copiy

***Abstract*— Markup languages allow creation of a flexible form for information representation. They make it possible to display information in a form convenient for a human and are actively used in all spheres of life. Although the main tasks of these languages are to structure representation of information and to facilitate its perception, the markup languages have a number of problems that make them difficult to be perceived by both a human and software. This work deals with problems typical for modern representation of the markup language family as well as with existing methods of their solution. Based on our research, we offer advanced methods as a background for a new original approach to information representation to be proposed and analyzed.**

***Keywords*— markup language, semantic, semantic layout, internet, information representation.**