

УДК 004.896

ПОСТРОЕНИЕ ОНТОЛОГИЙ ДЛЯ ВЭБ-ПРИЛОЖЕНИЙ

А.В. Григорьев, Е.В. Павловский
Донецкий национальный технический университет
grigorie@r5.dgtu.donetsk.ua, pavlovskiy.evgeny@mail.ru

В предлагаемой статье рассматривается специфика представления сайта, выполненного на языке HTML, как типичной модели структурного уровня абстракции САПР. Приводится пример выявления структурной модели в типичном сайте. Показывается, что данная модель структуры соответствует возможностям концептуальной модели предметной области инструментальной оболочки для автоматизации построения САПР, что позволяет сформулировать задачу построения такого рода оболочки.

Введение

Ранее авторами рассматривался подход представления веб-приложений с точки зрения онтологического подхода как модель некоторого объекта предметной области, призванного обеспечить представление знаний для БЗ. В данном случае база знаний относится к САПР веб-приложений.

Нашей общей задачей является создание инструментальной оболочки, способной автоматизировать создание базы знаний о методиках проектирования веб-приложений на требуемых языках программирования (HTML). Данная база знаний и является требуемым САПРом. Такой САПР должен обеспечить синтез требуемого веб-приложения, исходя из параметров, задаваемых конечным пользователем.

Приведенная ранее онтология соответствует стандартному подходу представления веб-страницы. Однако в таком виде она не обеспечивает возможность построения базы знаний о методиках проектирования веб-приложений в силу своей упрощенной и примитивной структуры.

Чтобы возможным было обеспечить построение такой базы знаний, необходимо обеспечить представление, например, страницы сайта с точки зрения стандартных методов представления объектов проектирования в САПР технических объектов. Такие модели делятся по уровню абстракции, например: структурные модели,

функционально-логические модели, количественная микро и макро модели.

Представить модель объекта, как структурную модель означает, ввести отношения: принадлежности к множеству для компонентов модели, класс – подкласс, часть – целое, экземпляр – класс и т.д. Представление подобного рода моделей для произвольных объектов любых предметных областей на уровне структурной модели ранее были рассмотрены автором в статье [2].

Целью предлагаемой работы являются:

- выявление структурной модели, имеющейся в типичном сайте, организованном с помощью средств языка HTML;
- представление данной модели в рамках возможностей концептуальной модели предметной области инструментальной оболочки для построения интеллектуальных САПР.

Для раскрытия данной темы необходимо осветить следующий ряд вопросов:

- дать характеристику языку разметки HTML, как языку строгой иерархии;
- описать роль онтологий, как средств наиболее полного представления страниц HTML;
- рассмотреть типичную логическую структуру страницы сайта;
- построить структурную модель HTML страницы в рамках средств концептуальной модели предметной области инструментальной оболочки для построения интеллектуальных САПР.

1. Язык разметки HTML, как язык строгой иерархии

Сам по себе язык разметки HTML строго структурирован. Язык состоит из множества тегов. Теги могут быть вложены друг в друга и могут иметь атрибуты. Обычно страница сайта разбивается на блоки. Это может быть лента новостей, форма авторизации, переключение языка, гостевая книга и т.д. Каждый блок имеет свои атрибуты и позиционирование относительно других блоков сайта. Распространенной является табличная верстка. Страница сайта разбивается на блоки с помощью таблицы, т.е. каждый отдельный блок – это отдельной ячейке таблицы. В свою очередь, блоки могут содержать неограниченное количество подблоков. Таким образом, каждый блок имеет входы и выходы. Его входами являются любые существенные воздействия среды на синтезируемый объект, выходами – связи объекта со средой. Язык HTML однозначно позиционирует блоки относительно друг другу. Вложенность тегов

друг друга – это мощное структурная особенность языка. На сегодняшний день является актуальным представление информации в виде онтологии. Причем каждый трактует это понятие по-своему. Есть базовое понятие и базовое формальное описание. Для конкретных задач можно формальное описание дополнять какими-либо параметрами.

2. Онтология, как наиболее полное представление страницы HTML

Онтология – это точная спецификация некоторой предметной области. Другими словами это модель данных, которая описывает понятия предметной области, что они означают, как они соотносятся друг с другом, и как они могут или не могут быть связаны друг с другом. Эта модель отображает представление о знании в некоторой предметной области и множество связей между ними.

Формально онтология [1] определяется как $O = \langle X, R, F \rangle$ (базовое формальное определение), где:

X — конечное множество понятий предметной области,

R — конечное множество отношений между понятиями,

F — конечное множество функций интерпретации.

Обычно, выделяют такие отношения между понятиями:

- класс – подкласс;
- часть – целое;
- экземпляр – класс.

Проанализировав существующие структуры данных, сделан вывод, что онтология – это оптимальный вариант для представления и связи компонентов HTML страницы в единую систему.

С точки зрения формального описания онтологии, для представления страницы HTML в виде онтологии, минимально необходимым является знание о составе объекта предметной области (в нашем случае это блоки) и связей между ними (в нашем случае вложенность блоков).

Выполним подобное представление для типичной страницы сайта. На рис. 1 представлен исходный текст главной страницы сайта.

Как видно из текста, сайт разбит на блоки таблицы `id="table_main"`. Содержимое отдельной ячейки этой таблицы – это то, что находится между тегами `<td></td>`. Это главные блоки страницы сайта. Подблоками главных блоков являются элементы, которые находятся в главном блоке.

```
<html>
<head>
<title>Стандартний сайт</title>
</head>
<body>
<table width="960" border="1" id="table_main">
<tr>
<td colspan="3">

</td>
</tr>
<tr>
<td colspan="3">
<!--Боковая панель-->
<td width="200" >
<form action="auth.php" method="post" id="form_auth" >
<table id="table_auth">
<tr>
<td colspan="2">
<!--Форма авторизацни-->
<td colspan="2" style="text-align: center;">
<input type="text" value="Логин" />
</td>
<td colspan="2">
<input type="text" value="Пароль" />
</td>
</tr>
<tr>
<td colspan="2" style="text-align: center;">
<input type="submit" value="Войти" />
</td>
</tr>
</table>
</form>
</td>
<td width="600" >
<p>Здравствуйте! Рады приветствовать Вас на нашем сайте!</p>
</td>
<td width="160" align="center">
<b>Новости</b>
<table id="table_news">
<tr>
<td align="center">
<div class="news_date"></div>07.09.2010
</td>
</tr>
<tr>
<td align="center">
<div class="news">Открылся наш сайт</div>
</td>
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="3" align="center">
<div>Сайт разработан Ивановым И.И.</div>
</td>
</tr>
</table>
</body>
</html>
```

Рисунок 1 – Исходный код главной страницы сайта на языке HTML

3. Логическая структура страницы сайта

На рис. 2 представлена логическая структура страницы сайта. Каждый блок (прямоугольник) – это ячейка главной таблицы (id="table_main"). С помощью этой таблицы сайт разбивается на блоки. Эти блоки и есть ячейки главной таблицы. Другими словами, ячейки главной таблицы – это блоки, которые являются вершиной иерархии блочной структуры сайта.

В свою очередь эти блоки могут иметь подблоки, которые принимают от внешнего блока его атрибуты, а также имеют свои. На многих сайтах при переходе на другие страницы меняется только центральный блок. Центральный блок обязателен, т.к. в нем содержится главная информация. Остальные блоки, такие как верхний или боковой, могут отсутствовать.

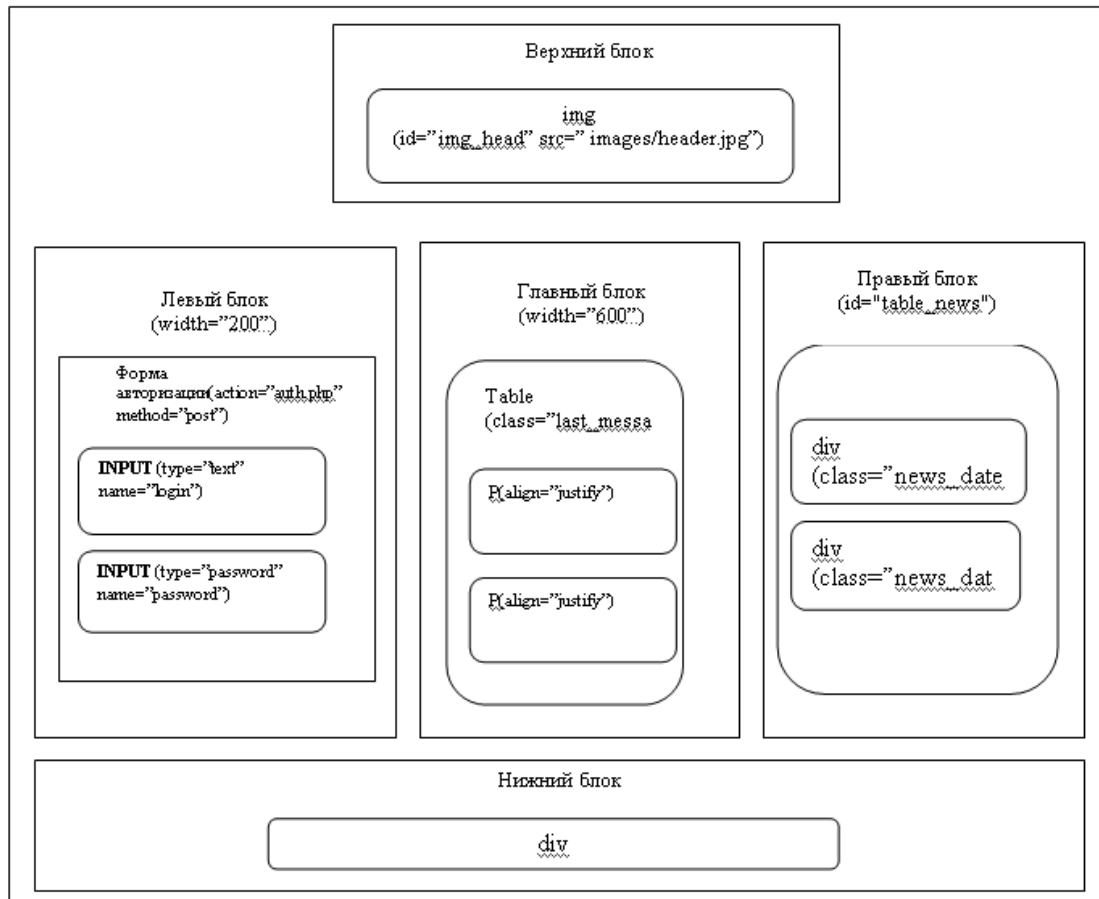


Рисунок 2 – Логическая структура страницы сайта

4. Структурная модель HTML страницы

Многие элементы страницы (теги) имеют атрибуты. Например, ``. Тег `img` – является изображением, адрес которого находится в теге `src`. Также элемент содержит атрибуты ширины и высоты изображения, а также идентификатор `id`. Элемент `` является подблоком главного блока (таблица `id="table_main"`). Атрибуты находятся на внутренней границе блока.

Как известно, заголовок HTML страницы находится в теге `<title>`. Из этого следует, что текст заголовка – это подблок блока `title`. Вход для блока с текстом заголовка страницы – является выходом из блока `title`. Текст заголовка – подблок блока `title` (тега `<title>`).

Вышеизложенные сущности представлены на рис. 3. Остальные элементы находятся в блоке `body`. При синтезе HTML страницы это будет предусмотрено, но на данной стадии блок `body` на схеме не обозначен, т.к. это само собой разумеется. Все что мы видим на странице сайта – находится в теге `<body>`, т.е. все элементы являются подчиненными блоку `body`.

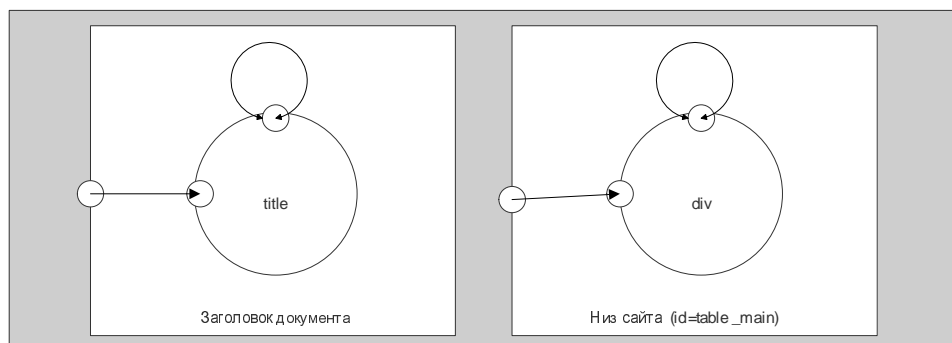


Рисунок 3 – Блок заголовка страницы и блок с автором сайта

Изображение вверху сайта находится в верхнем блоке главной таблицы. К нему на вход подаются атрибуты внешнего блока (рисунок 4). Как видно из листинга страницы (рис. 1), блок с изображением занимает три колонки таблицы. Этот атрибут подается на внутреннюю границу блока изображения.

Текст приветствия находится в центральной ячейке главной таблицы. Текст находится в теге `<p>`, который находится во внешнем блоке главной таблице. Ему на вход передаются атрибута внешнего блока.

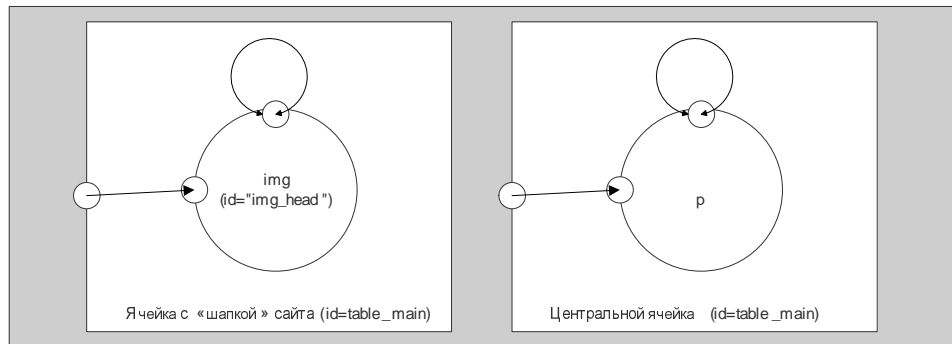


Рисунок 4 – Блок «шапки сайта» и центральный текст главной страницы

Форма авторизации находится во вложенной таблице `id="table_auth"`. Блок `table` у которого `id="table_auth"`, является подблоком блока `form` с `id="form_auth"`, который в свою очередь является подблоком блока главной таблицы `id="table_main"`.

Для полей ввода логина и пароля входами является выход блока пользователь, т.е. информация которую вводит пользователь. Выходы блоков логина и пароля является входами для базы данных.

Поля для ввода – это ассоциативный массив вида `[[name1,value],[name2,value],...,[nameN,value]]`. `Name` – это имя элемента для ввода, `value` – это значение введенное пользователем. Их содержимое отправляется на сервер, после чего сопоставляется с содержимым базы данных.

Как видно из рис. 5, элементы html страницы могут быть вложены друг в друга. Форма авторизации `<form id="form_auth"></form>` вложена в ячейку таблицы `<table id="table_auth"></table>`. Последняя в свою очередь содержится в ячейке главной таблицы `<table id="table_main"></table>`.

Приходим к выводу, что подблок должен иметь вход только от одного внешнего блока, который ближе к нему. Ведь атрибуты дальнего внешнего блока, будут переданы на вход подблоку. Этого достаточно для построения четкой иерархии, которую предусматривает формат HTML.

Из рис. 1 видно, что на странице сайта присутствует лента новостей. Обычно эта информация извлекается из базы данных и в удобочитаемом виде располагается на странице (обычно в таблице).

Новость состоит из даты и самого текста новости. Данные из базы данных возвращаются в массиве. Каждый элемент – это дата и текст новости. Упрощенно – это массива вида `[[div1,div2],[div3,div4]...[divN-1,divN]]`, где каждый элемент – это одна новость.

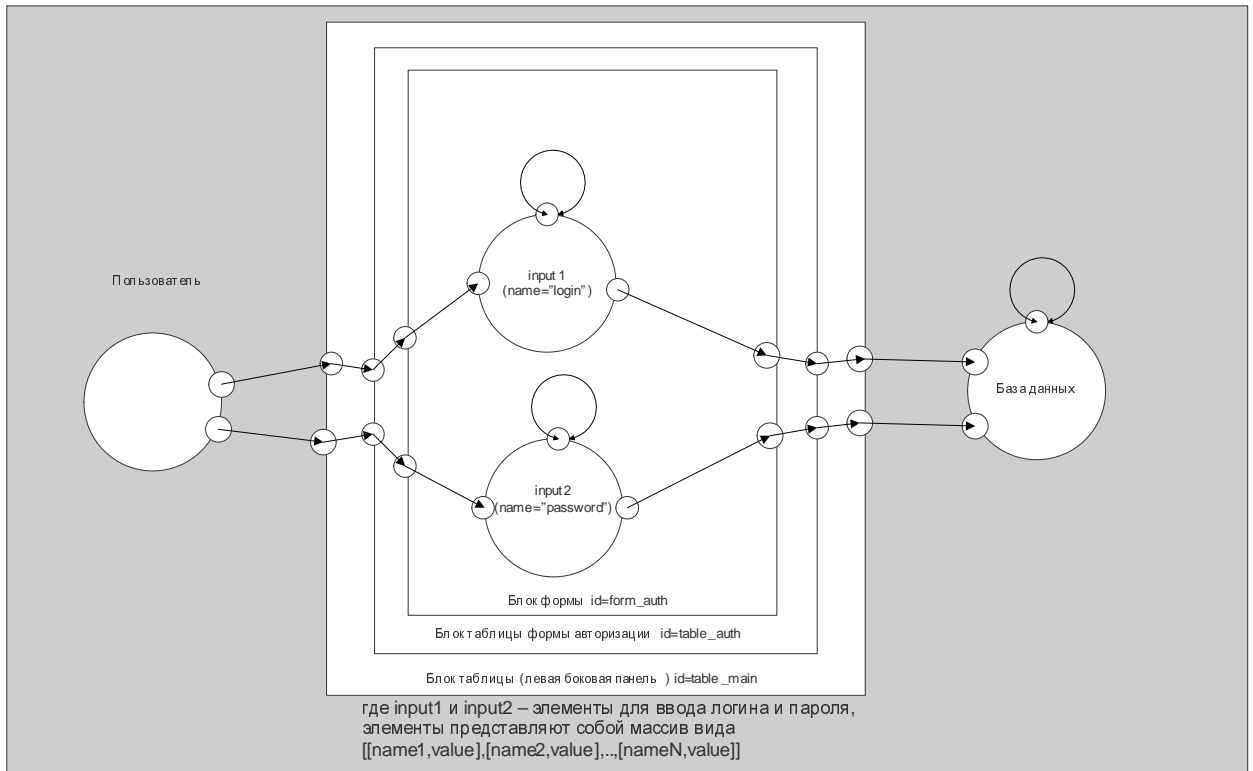


Рисунок 5 – Блок формы авторизации

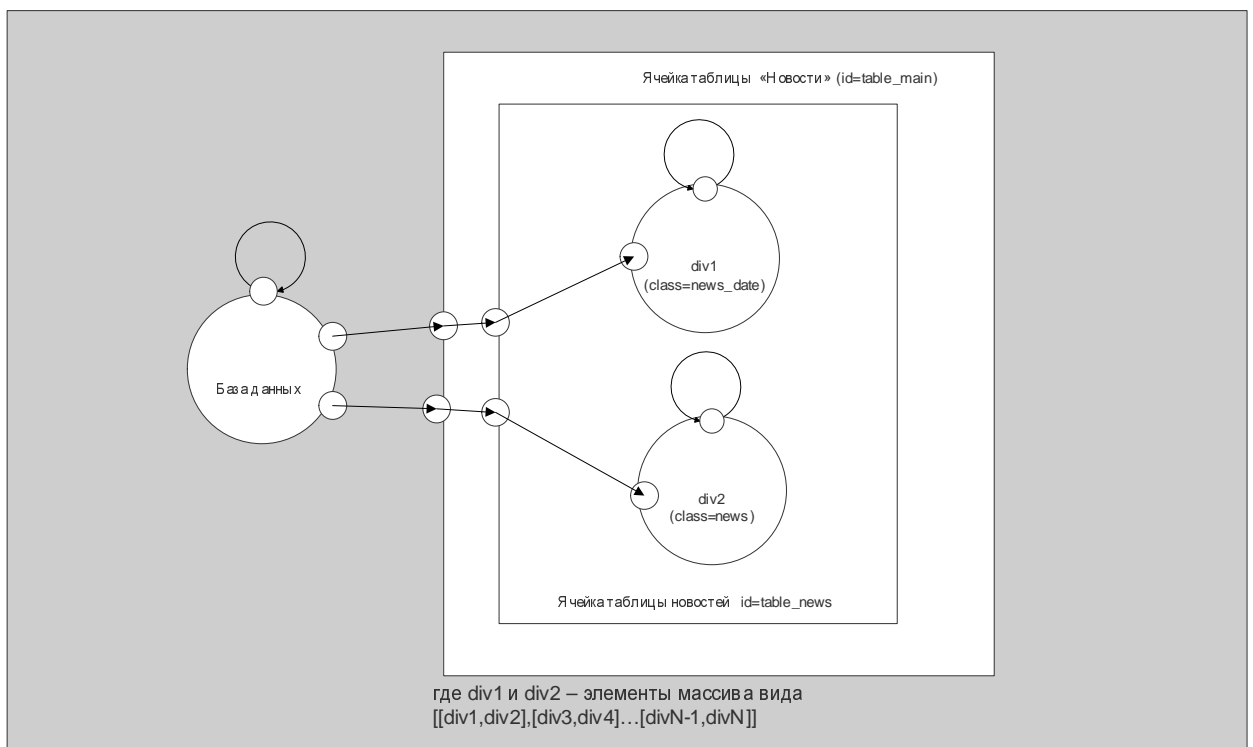


Рисунок 6 – Блок ленты новостей

Из рис. 6 видно, что данные из базы данных подаются на границу внешнего блока главной таблицы id="table_name", затем внешний блок передает данные и атрибуты на подблок id="table_news".

Вывод

Подобное представление структуры html файла позволяет создать систему модулей знаний о методах построения блоков различных типов, имеющихся в html файле. Подобное представление соответствует заданию структуры объекта на универсальном языке формальных спецификаций. При наличии транслятора «язык формальных спецификаций» -> проблемно-ориентированных язык (HTML) можно обеспечить синтез желаемого сайта пользователем, эксплуатирующего готовую базу знаний.

Таким образом, предлагаемый путь разработки инструментального комплекса, способного обеспечить создания в его среде баз знаний для автоматизации синтеза сайтов как интеллектуального САПР потенциально эффективен. Однако стоит отметить, что задача имеет ряд специфических особенностей, что требует разработки соответствующего комплекса средств концептуальной модели, способного учесть специфику задачи.

Список литературы

1. Онтология [Электронный ресурс]. Энциклопедия Википедия. – Режим доступа: <http://ru.science.wikia.com/wiki/Онтология>.
2. Григорьев А.В. Организация временного и пространственного логического вывода в концептуальной модели интеллектуальных САПР. [Электронный ресурс] / А.В. Григорьев – Режим доступа: http://www.nbu.gov.ua/portal/natural/Npdntu_ikot/2008/08gavmis.pdf.

Получено 08.09.2011