

УДК 004.822

АНАЛИЗ ПОДХОДОВ К РЕАЛИЗАЦИИ ИНТЕЛЛЕКТУАЛЬНЫХ CMS

Павловский Е.В., Григорьев А.В.

Донецкий национальный технический университет, г. Донецк
кафедра прикладной математики и информатики
E-mail: grigorie@pmi.dgtu.donetsk.ua, pavlovskiy.evgeny@mail.ru

Аннотация

Павловский Е.В., Григорьев А.В. Анализ подходов к реализации интеллектуальных CMS. В статье рассмотрены достоинства и недостатки существующих систем для построения web-приложений. Рассмотрена задача разработки инструментальной системы для построения интеллектуальных САПР web-приложений. Описана функция данной инструментальной системы, состоящая в построении САПР как экспертной системы, в базу знаний которой вводится методика проектирования сайтов. Вкратце рассмотрен предлагаемый механизм представления знаний.

Общая постановка проблемы

В настоящее время рынок фирм и организаций, которые, предоставляют услуги в области web-технологий, растет с каждым днем. Это положительно сказывается на цене конечного продукта, т.к. конкуренция способствует удешевлению продукта или, по крайней мере, предоставляет клиенту диапазон цены продукта. Эта цифра содержит в себе заработную плату сотрудников, которые задействованы в разработке вэб-приложения.

Актуальной задачей является оптимизация трудозатрат. Сокращая количество персонала, возникает проблема увеличения времени на разработку, а этот параметр, является одним из главных для заказчика. Второй момент заключается в том, что рядовой пользователь сейчас на своем ПК владеет очень большим выбором программных средств для решения тех или иных задач, таких как запись дисков, редактор текстов или графический редактор. У пользователя возникает вопрос необходимости такой системы, с помощью которой он мог бы самостоятельно создать вэб-приложение, ориентированное на его уровень и удовлетворяющее его потребности.

Еще одна проблема, с которой сталкиваются web-разработчики – это повторное использование кода. Трудоемко писать каждое приложение «с нуля», тем более, что прототип подобного приложения уже есть. Таким образом, возникает необходимость внедрения в технологию разработки web-приложений системы автоматического проектирования (САПР) [1].

В настоящее время в области искусственного интеллекта существует тенденция замены инженера по знаниям в процессе построения баз знаний на эксперта предметной области. Это позволит эксперту создавать web-приложения самостоятельно, без привлечения инженера по знаниям.

Таким образом, актуальной задачей является разработка инструментальной оболочки для построения интеллектуальных САПР, в базу знаний которой вводится методика проектирования web-приложений. Построенный САПР позволил бы конечному пользователю синтезировать требуемую САПР по техническому заданию.

Постановка задачи

Предложить путь построения интеллектуальных CMS, для построения которых нет необходимости привлекать инженеров по знаниям, а можно обеспечить возможность ее

построения эксперту в предметной области, имеющему достаточно большой опыт в разработке web-приложений.

Решение задач и результаты исследований

1. Требования к разрабатываемой системе

Объектом проектирования САПР является веб-приложение, или в частном случае сайт, который соответствует всем требованиям эксперта. В процессе синтеза решается задача выбора решения из ограниченного множества решений, удовлетворяющего пользователя. Нужно разработать инструмент для выбора наиболее подходящего решения.

Система должна решить следующие задачи:

- оптимизация трудозатрат;
- сокращение времени на создание конечного продукта;
- создание БЗ с использованием подходов, интуитивно понятных эксперту в предметной области, который не является экспертом в инженерии знаний;
- разработка средств синтеза web-приложений в рамках готовой БЗ, организующих диалог на уровне, интуитивно понятном конечному пользователю, который может даже не являться экспертом в web-разработке;
- синтез сайта на проблемно-независимом языке, на основании БЗ и ответов пользователя;
- конвертация из проблемно-независимого языка в проблемно-зависимый (в частном случае – это язык HTML).

2. Обзор существующих САПР

На сегодняшний день существует ряд CMS для автоматического создания сайтов. Другими словами, это система управления содержимым. Данные системы позволяют добавлять, редактировать и удалять содержимое на сайте. Манипуляции с содержимым проходят посредством административной части, встроенной в ядро системы. Система позволяет в удобном и понятном для пользователя интерфейсе совершать действия с информацией.

Система позволяет выбирать макет (тему) для внешнего вида web-страниц сайта. Этот макет поделен на именованные области. Это нужно для размещения новых сущностей (модулей, страниц и т.д.). Т.е. видимые объекты на странице связаны с именованной областью. Обычно в систему встроены базовые макеты, но пользователь может загрузить в систему любой понравившийся ему макет, разработанный для этой системы. По своей структуре, темы похожи друг на друга. Обычно в верхней части находится область для картинки (логотипа), которая обычно занимает всю ширину. Под или над ней находится область для главного меню. Затем, обычно ниже, находится область для основного содержания, новостей, копирайта и т.д.

С помощью настроек сайта можно изменить язык интерфейса, что позволяет пользоваться системой людям со знанием различных языков. Т.е. базовые интерфейсы системы переведены разработчиками на множество языков.

Среди преимуществ CMS можно выделить следующие:

1. Возможность размещать и редактировать материал пользователю любой квалификации. Для этого в системе существует понятный и доступный интерфейс для работы с информацией. Редактор сам разметит обычный текст с помощью html-тегов. Это предотвращает затраты на привлечения специалиста со стороны.

2. Уменьшение стоимости и времени на разработку сайта, что достигается использованием готовых модулей для системы. Эти модули разрабатываются специально для конкретной системы. Они постоянно тестируются пользователями и обновляются по мере необходимости.

3. Организация совместной работы. Система позволяет работать над одной и той же информацией многим пользователям и ограничивать доступ к материалам в зависимости от прав пользователя. Администратор может редактировать роли пользователя, что положительно сказывается на работе предприятия, каждый отдел которого, имеет доступ только к своим данным.

4. Безопасность системы. Один из важнейших преимуществ CMS. В крупных компаниях, занимающихся разработкой сайтов, уделяется огромное внимание теме безопасности. Это касается как сохранение целостности и конфиденциальности данных, так и работоспособности сайта в целом.

База знаний (БЗ), как таковая, в таких системах отсутствует. Вместо нее профессионалами в области web-программирования разрабатываются модули. Они из себя представляют код на формально-зависимом языке, что ограничивает пользователя средней квалификации в изменении существующих модулей и добавление новых.

Типичная CMS представляет собой проблемно-ориентированный графический редактор, практически лишенный интеллектуальных средств поддержки процесса проектирования web-приложений. Для построения базы знаний мы предлагаем привлекать не инженера по знаниям, а эксперта в предметной области, имеющего достаточно большой опыт разработки web-приложений.

3. Краткое изложение предлагаемого метода представления знаний

Таким образом, возникает задача разработки инструментальной системы для построения интеллектуальных САПР. Т.е. инструментальная система строит САПР как экспертную систему, в базу знаний которой вводится методика проектирования сайтов. Он позволяет конечному пользователю синтезировать требуемый сайт по техническому заданию. Оно с точки зрения продукционной базы знаний [2] рассматривается как «цель вывода». В процессе синтеза, т.е. логического вывода, по заданной цели вывода в продукционной базе знаний строится результат вывода, т.е. готовый сайт.

В режиме диалога с пользователем исключаются некоторые варианты. Это происходит до тех пор, пока не останется других вариантов. Также для ускорения процесса интервью, пользователь может задать связь несовместности между понятиями. Таким образом, если два понятия А и В связаны отношением несовместности, то если А нужно на сайте, то все ветки дерева с понятием В удаляются.

Таким образом, с введением таких диаграмм, система приобретает интеллектуальность, чего не хватает в существующих CMS, так как они рассчитаны на квалифицированных пользователей. Таким образом, можно решить проблему проектирования сайтов пользователем среднего уровня квалификации, который не знает проблемно-ориентированный язык.

Выбор структуры данных для представления знаний – важнейший этап проектирования. Структура должна быть понятна эксперту и при этом должна минимизировать временные и технические затраты на обработку запроса.

Таким запросом может быть:

- навигация;
- быстрый поиск нужного понятия;
- добавление и редактирование понятия;
- перемещение понятия с подчиненными понятиями;
- удаление понятия с подчиненными понятиями.

Как мы видим, имеется достаточно много требований к структуре. В искусственном интеллекте основная цель – хранить знания таким образом, чтобы программы могли извлекать их и работать с ними. Таким образом, самой оптимальной структурой данных является онтология, которая представляет из себя декомпозицию понятий [3]. Она

классифицирует понятия предметной области, задает связи между ними и ограничения. На основании данной онтологии инструментальная оболочка должна построить И-ИЛИ дерево со всеми возможными вариантами связей и атрибутов. Т.е. нужно создать такое дерево, которое будет включать в себя все варианты. Такое представление знаний позволит синтезировать требуемое решение. Т.е. в режиме диалога с пользователем, некоторые из вариантов не будут удовлетворять ответам. И таким образом останутся только удовлетворяющие его варианты.

Достоинства использования БЗ – это разработка типичных сайтов, без использования кода прототипа. Специфика предметной области такова, что может потребоваться создания сайтов различных классов. Значит, система должна перебрать все допустимые варианты связей и атрибутов.

Связи между понятиями могут быть следующими:

- блок – подблок;
- класс-подкласс;
- отношение несовместности (взаимоисключающая связь).

Кратко рассмотрим каждую из них:

1) Блок-подблок – это связь структурного уровня. Она показывает включение одного блока в другой;

2) Отношение несовместности – это связь, которая показывает взаимоисключающие элементы. Т.е. если на этапе диалога с пользователем выясняется, что один элемент будет присутствовать на странице, то автоматически второй удаляется из И-ИЛИ дерева. Это может существенно сократить время опроса пользователя и сузить количество вариантов решений;

3) Класс-подкласс – это связь очень похожа на наследование, которое применяется в объектно-ориентированном программировании. В нашем случае поля класса – это атрибуты примитивов. Если два и более элементов имеют одинаковые атрибуты, то эксперту достаточно связать эти элементы связью класс-подкласс. Мощностью данного метода в том, что можно переопределить атрибуты родителя, а также добавить свои атрибуты, которые отсутствуют в базовом примитиве.

Вид такого И-ИЛИ-дерева приведен на рис. 1.

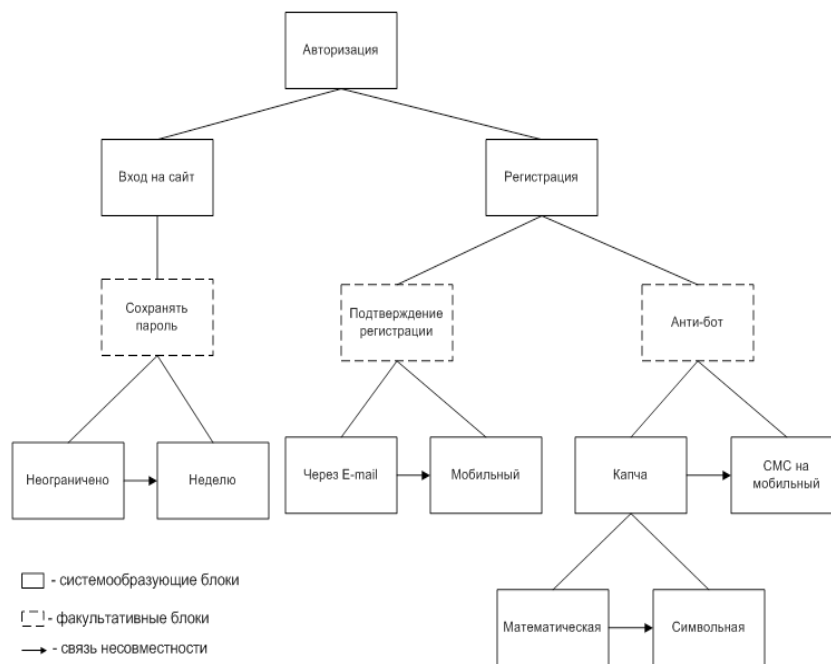


Рисунок 1 – И-ИЛИ-дерево

Таким образом, построенный САПР как экспертная система позволяет конечному пользователю синтезировать требуемый САПР по техническому заданию.

Процесс построения базы знаний разбит на последовательный ввод модулей знаний, от более крупных к более мелким. При синтезе с помощью готовой базы знаний можно выбрать модуль знаний любого уровня и запустить логический вывод.

Модуль знаний разбивается на подмодули [4]:

- внешняя граница как состав внешних параметров, данных, которые отражают связи блока с внешней средой. Это отдельное И-ИЛИ-дерево с отношениями несовместности;
- внутренняя граница как состав внутренних подблоков, тоже как И-ИЛИ-дерево;
- совокупность связей между подблоками по данным, тоже как И-ИЛИ-дерево.

Кроме того, смежные подмодули тоже связаны отношениями несовместности. В процессе вывода в рамках подмодуля знаний система вывода движется в каком-то порядке по всем ИЛИ-узлам. Выбрав какой-то ИЛИ-узел, система вывода делает из него вопрос.

Например, пользователь выбирает один вариант ответа на вопрос. Тогда система по отношениям несовместности в ИЛИ-дерево удаляет в других ИЛИ-узлах те альтернативы, которые несовместимы с наличием не нужного блока.

Выводы

Были рассмотрены функциональные возможности типичных CMS, как систем САПР, решающие задачи построения web-приложений. У каждой CMS есть свои достоинства и недостатки. Самый главный недостаток всех подобных систем – это ориентация на пользователя, который знаком с внутренним устройством сайтов и языком html, что не позволяет среднестатистическому пользователю создать сайт, который бы удовлетворял все его требования.

В предыдущих статьях мы пришли к выводу, что оптимальной структурой данных для представления знаний является онтология. Сейчас выяснили, что базовое формальное описание онтологии – не совсем подходит для представления знаний в данной предметной области.

В предлагаемой статье:

- предложен механизм построения интеллектуальных CMS, для построения которых нет необходимости привлекать инженеров по знаниям, а можно обеспечить возможность ее построения эксперту в предметной области, имеющему достаточно большой опыт в разработке web-приложений.
- в общем виде подробно разобран механизм представления знаний в предлагаемой оболочке.

Полученные результаты дают возможность перейти к следующему этапу проектирования экспертной системы, а именно к рассмотрению методов извлечения знаний.

Список литературы

1. Ли К. Основы САПР (CAD/ CAM/CAE). – СПб.: Питер, 2004. – 560 с.
2. Братко И. Программирование на языке Пролог для искусственного интеллекта: Пер. с англ. – М.: Мир, 1990. – 560 с.
3. Григорьев А.В., Павловский Е.В. Анализ методов построения онтологий для построения экспертных систем по синтезу моделей сложных систем в САПР / Григорьев А.В., Павловский Е.В. // Научные труды Донецкого государственного университета: Серия: “Вычислительная техника и автоматизация”. – 2011. – № 21. – С. 87-94.
4. Григорьев А.В. Семиотическая модель базы знаний САПР / Григорьев А. В. // Научные труды Донецкого государственного технического университета: Серия: «Проблемы моделирования и автоматизации проектирования динамических систем» – 1999. – №10 — С. 30-37.