

Аннотация

К научно-техническому отчету по государственному контракту № П159 от 15 апреля 2010 г.

К настоящему моменту человечество накопило огромное количество разнообразной информации. Все мировые информационные ресурсы представляют собой разрозненные источники данных и задачи структурирования, обработки этой информации становятся все актуальнее. В ответ на это разрабатываются технологии позволяющие унифицировать знания и передать часть работы по интеллектуальной обработке информации сетевым сервисам и агентам. На данный момент большие достижения сделаны в области структурирования информации в универсальном и наиболее полном формате, способном описать все взаимосвязи информации между собой. Основным инструментом для этого являются онтологии.

Онтологий сегодня является одним из самых активно развивающихся формализмов представления знаний в интернете. Эффективный вариант накопления и представления знаний, являющийся, с одной стороны, достаточно выразительным, с другой стороны, естественным и легким для понимания. Онтология определяется как иерархия классов и объектов плюс логическое описание свойств и механизмов взаимодействия этих объектов. Онтология - это формальный способ представления знаний, который состоит из таксономии, набора логических свойств предметной области и правил вывода. Онтология универсальна и способна описывать многие системы знаний. Понятие онтологии было формализовано и стандартизировано в рамках языка OWL (Ontology Web Language) консорциумом W3C.

Также по мере создания баз знаний в универсальных форматах, возникает необходимость использования этих знаний людьми. Это

обуславливает создание простых и универсальных средств доступа к базам знаний. Поскольку использование знаний достигается через пользовательские приложения, которые реализуют логику работы пользователей, необходимость создания инструментов для разработки таких пользовательских приложений для баз знаний становится очевидной и чрезвычайно актуальной.

Данная работа посвящена созданию инструментальных средств для простого и быстрого создания интерфейсов к базам знаний. С помощью этого инструментария разработчики баз знаний смогут быстро развёртывать пользовательские приложения, легко менять логику работы этих приложений в ответ на меняющиеся требования пользователей.

Пользовательские приложения, создаваемые с помощью этого инструментария, принадлежат к классу веб-приложений. Веб-приложение — клиент-серверное приложение, в котором клиентом выступает браузер, а сервером — веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются межплатформенными сервисами. Для веб-приложения не требуется особенных библиотек или программ, пользователю достаточно иметь установленный на своем компьютере браузер.

Ключевым моментом созданного инструментария является общий для всех компонентов язык программирования Libretto. При помощи этого языка происходит создание структуры базы знаний, заполнение базы знаний информацией, создание пользовательских интерфейсов для доступа к базе знаний. Дополнительным плюсом является то, что при

необходимости создания более производительного интерфейса, существует возможность использования языка программирования Java.

В процессе исследования были получены следующие результаты:

- создан компонент загрузки страниц интерфейса в онтологию. Компонент включает в себя онтологию, которая содержит 12 базовых классов для описания структуры интерфейса в виде дерева объектов и 25 базовых функций для манипуляции элементами интерфейса в дереве объектов;
- создан компонент отображения интерфейса из структуры онтологии. Компонент включает в себя 18 Java-классов, реализующих расширение для Play framework, который передает результат выполнения шаблона документа интерфейса пользователю;
- создана подсистема реализации запросов в онтологии по вызову пользовательских событий. Компонент включает в себя классы для реализации пользовательских событий и для реакции на пользовательские события. Предопределено 15 объектов пользовательских событий;
- создано приложения для редактирования структуры онтологий и хранящихся в них объектах состоящее из 19 шаблонов-контролеров и 11 шаблонов-видов.

Руководитель проекта

_____ Москвина А.С.