

ПРЕИМУЩЕСТВА ПРИМЕНЕНИЯ ОНТОЛОГИЙ ДЛЯ РЕАЛИЗАЦИИ ИНТЕРФЕЙСА WEB-РЕСУРСОВ

Бибичков И.Е., Сокол В.В.

Научный руководитель — к. т. н. доц. Шевченко А.Ю.

Харьковский национальный университет радиоэлектроники
(61166, Харьков, пр. Науки, 14, каф. Искусственного интеллекта,
тел. +38 (057) 702-13-37)

email: bibi4kov@gmail.com, тел (093) 667 35 76

The OWL Web Ontology Language is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics. [1].

The main advantage of using of the Ontology Driven Development (ODD)[2] principle is the ability to create more flexible applications, and as a result more dynamic User Interface (UI).

Основная идея представленного концепта – реализация преимуществ использования онтологий: гибкость, слабая связность, расширяемость, с существующими подходами разработки UI для Web-приложений.

Классическая многослойная архитектура разработки современных приложений регламентирует отделение бизнес логики от интерфейса и слоя работы с данными. Следовательно, изменения в модели данных, влекут за собой изменения в бизнес-логике программы а так же, зачастую, необходимость внесения изменений в UI. Применение предложенного подхода, позволяет уменьшить количество необходимых изменений программного кода при изменении функциональных характеристик ПО. Такой эффект достигается через особую модель разработки ПО, построенную на основе ODD подхода.

Для смены предметной области Web-приложения достаточно внести изменения в онтологию, а именно, заменить одноименную онтологию. Данный подход предполагает наличие всего трех онтологий: «Сервисная Онтология», «Онтология Предметной Области» и «Онтология Экземпляров Классов» (рис. 1).

«Сервисная онтология». Основная задача – хранение ключевых классов и свойств, чьими потомками являются классы предметной области. Данная онтология является корневой и определяющей функциональный состав системы. Любые изменения, которые были произведены в онтологии отразятся в функционировании системы на базовом уровне, а точнее, на уровне ядра. Сервисная онтология определяет структуру и задает ограничения для онтологии предметной области (рис. 1). Кроме того, в сервисная онтология определяет поведение базовых элементов UI вне зависимости от «Онтологии Предметной Области».

«Онтология предметной области» (рис. 1) наследуется от «Сервисной Онтологии» и определяет поведение и внешний вид всех объектов, которые можно зарегистрировать в системе. Данная онтология регламентирует структуру информации, которая будет представлена пользователю. «Онтология Предметной области» задаёт свойства и связи для элементов «Онтологии Экземпляров Классов».

«Онтология Экземпляров Классов» (индивидов) (рис. 1) содержит все данные системы в виде объектов: пользовательскую информацию, частные элементы UI. Данная онтология наследуется от «Онтологии Предметной Области»[3]. Экземпляры всех классов добавляются лишь в эту онтологию. Такое выделение индивидов, обеспечивает дополнительную гибкость и универсальность для ПО, построенного на основе онтологий. В случае смены предметной области ПО, нет необходимости перестраивать всю структуру хранения данных. Это объясняется тем, что в онтологии информация хранится в виде объектов, а не в строго структурированном виде, как в классических системах управления базами данных (СУБД). Кроме того, благодаря наследственной зависимости от «Онтологии Предметной Области» для обновления внешнего вида Web-приложения, нет необходимости добавлять дополнительные объекты. Все элементы UI обновляются следом за обновлением «Онтологии Предметной Области»



Рисунок 1 – Схема онтологической модели знаний

Источники:

1. W3C / RDF Schema [Электронный ресурс]. Режим доступа: <https://www.w3.org/TR/rdf-schema/>
2. Ceh, I. Ontology Driven Development of Domain-Specific Languages. [Text] / I. Ceh, M. Crepinsek, T. Kosar, M. Mernik // Comput. Sci. Inf. Syst. 8. 317-342. 10.2298/CSIS101231019C.- 2011
3. Бибичков, И. Модель самоадаптивного ПО, основанная на знаниях из корпоративной памяти [Текст] / Бибичков И.Е., Сокол В.В., Шевченко А.Ю. // Системи озброєння і військова техніка, No 4(52) ISSN 1997-9568., 2017. – с. 137-144