

ТЕЗИСЫ ДОКЛАДА

«ОНТОЛОГИЯ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ»

Информация об авторах

ФИО	Рабчевский Евгений Андреевич	Архипов Егор Сергеевич	Михаил Андреевич Марценюк
Место работы	Пермский Государственный Университет Физический факультет Кафедра Компьютерных Систем и Телекоммуникаций		
Должность	Аспирант, первого года обучения	Студент 5-го курса	зав. кафедры д.ф-м.н, профессор
Роль в проекте	Разработка общей онтологии	Разработка частной онтологии, интерфейса	Научный руководитель
Контакты	evgeny@ranat.ru	saiBro@mail.ru	mrcn@psu.ru
Адрес	Россия Пермь 614059 Постаногова 1-66		

Актуальность, новизна, аналоги.

Каждый производитель технических товаров стремится формализовать и автоматизировать техническую поддержку своего товара. Сегодня, когда большинство как программных, так и аппаратных средств распределены и находятся в сети, необходимо использовать соответствующие средства представления знаний для автоматизации этих процессов.

В данной работе предлагается использовать для этих задач технологии Semantic Web. В этом плане, необходимо упомянуть несколько работ, это российский проект <http://semweb.krasu.ru> – пример сайта, данные которого хранятся в виде OWL-онтологии. Проект <http://kml.mipt.ru/> предлагает разметку научных знаний в машино-читаемом виде (на основе XML) для дальнейших запросов. Наиболее известный (уже классический в литературе по Semantic Web) из зарубежных проектов <http://www.ksl.stanford.edu/people/dlm/webont/wineAgent/> - винный агент, или экспертная система по подбору блюд к вину.

Как за рубежом, так и в России особо востребованы экспертные системы в области медицины, ввиду чрезвычайной сложности предметной области. Проект <http://www.vtc.ru> «Онтология предметной области "Медицинская диагностика"» использует язык Ontolingua.

В данной работе разработана методика создания онтологий, представляющих базу знаний для задач технической поддержки. По методике созданы экспертная система и база знаний для технической поддержки компьютера.

Постановка задачи

1. Разработать методику создания онтологий, представляющих базу знаний для задач технической поддержки.
2. Разработать прототип сетевой экспертной системы технической поддержки компьютера.
3. Реализовать прототип экспертной системы для поддержки компьютера.

Модели и технологии

Как большинство экспертных систем, и интеллектуальных систем вообще, разрабатываемый прототип состоит из следующих функциональных частей:

1. **База знаний.** В качестве средства представления знаний был выбран OWL Ontology Web Language или язык сетевых онтологий, изначально ориентированный на применение в сети.
2. **Модель логического вывода.** Для реализации машины логического вывода используется библиотека Java классов для работы с OWL онтологиями - Jena Semantic Web Framework. Данная библиотека позволяет выстраивать необходимые модели логического вывода.
3. **Интерфейс** реализуется на сервлетах и JSP страницах, компонентах спецификации J2EE.

Методы решения

Очевидно, что, создавая онтологию предметной области, необходимо исходить из того, как будет она использоваться. Однако более конкретных рекомендаций в различных работах по созданию онтологий нет. В нашем случае можно было поступить двумя способами:

1. Разработать онтологию конкретной предметной области, и разрабатывать машину вывода и интерфейс под эту конкретную область.
2. Между онтологией конкретной предметной области и машиной вывода ввести **онтологию методологии принятия решений.**

Для решения задачи по методике создания онтологий для задач технической поддержки был выбран более общий уровень, то есть второй вариант. Такое обобщение можно ввести, так как в задаче технической поддержки любого технического объекта есть некие общие моменты. Так в любой задаче присутствуют пользователь и виртуальный эксперт, задача эксперта помочь пользователю перевести поддерживаемую систему из текущего состояния в целевое. При этом методология принятия решений экспертом может быть распространена на целый класс задач.

Такой подход позволит реализовать разрабатываемый прототип для других задач того же класса.

Итак, **объектом научного исследования** данной работы является **методика создания онтологии предметной области для технической поддержки.**

Общая концепция разработки

В каждой конкретной задаче технической поддержки существуют общие принципы поведения пользователей и экспертов. Для универсализации решения необходимо выделить модели поведения эксперта и пользователя в отдельную онтологию.

Разработанная методика создания онтологий – баз знаний для технической поддержки сама выражена в виде OWL онтологии («общей»). В ней заложены понятия (независимые от предметной области), которыми оперирует эксперт, принимая решения.

В «частной» онтологии описывается сама предметная область. Частная онтология разрабатывается экспертом в предметной области, при этом он «размечает её» в терминах общей онтологии.

Структура системы

Прототип имеет трехуровневую структуру, представленную в таблице 1.

Уровень	Подуровень	Функциональность	Реализация
База знаний	Частная онтология	Формальное описание поддерживаемого объекта, в терминах общей онтологии	OWL онтология
	Общая онтология	Определяет понятия, с которыми работает эксперт	OWL онтология
Модель логического вывода		Обеспечивает наиболее эффективный способ перемещения системы в целевое состояние	Java классы
Интерфейс	Пользователь	Использование	JSP страница

Таблица 1. Трехуровневая структура прототипа.

Работа прототипа

Уровень «общей» онтологии

1. Пользователь имеет дело с конкретной СИСТЕМОЙ.
2. В СИСТЕМЕ, по принципу функциональности выделяются ОБЪЕКТЫ.
3. ОБЪЕКТ может иметь ПОДОБЪЕКТЫ.
4. ОБЪЕКТ обладает набором ПРОБЛЕМ, функциональных элементов, при использовании которых, у пользователя могут возникнуть проблемы.
5. ПРОБЛЕМА имеет набор ФАКТОРОВ, настроек СИСТЕМЫ или внешних факторов, которые так или иначе могут повлиять на ПРОБЛЕМУ.
6. ПРОБЛЕМА находится в одном из указанных СОСТОЯНИЙ.
7. ФАКТОР имеет одно из указанных ЗНАЧЕНИЙ.
8. Каждое состояние всякой проблемы **однозначно** определяется значениями всех влияющих на неё факторов через ОТНОШЕНИЕ_ФУНКЦИОНАЛЬНОСТИ.
9. Одни и те же факторы могут влиять на разные проблемы.
10. В рамках одной ПРОБЛЕМЫ каждый ФАКТОР имеет ПРИОРИТЕТ (для реализации квадруплетной модели ПРОБЛЕМА-ФАКТОР-ПРИОРИТЕТ необходимо было расширить триплетную модель OWL).
11. ФАКТОРУ определено КАК_ИЗМЕРИТЬ_ИЛИ_ИЗМЕНИТЬ его ЗНАЧЕНИЕ.

На рис.1 представлена схема общей онтологии.

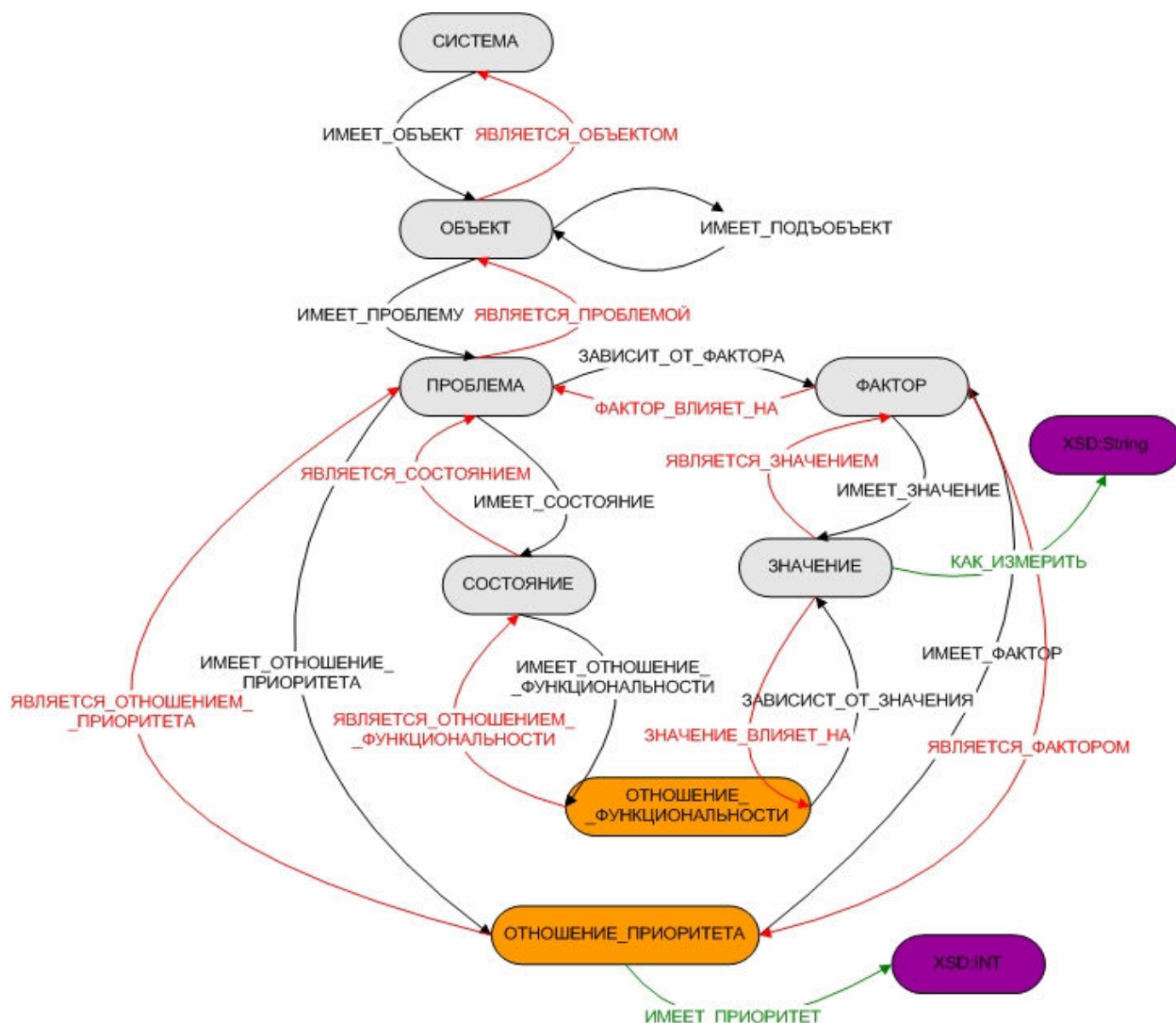


Рис.1 Схема общей онтологии

Уровень логического вывода

1. Есть ТЕКУЩЕЕ и ЦЕЛЕВОЕ СОСТОЯНИЯ.
2. В случае обнаружения нескольких ПРОБЛЕМ, выполняется динамическое вычисление порядка определения ЗНАЧЕНИЙ ФАКТОРОВ (исходя из области пересечения факторных множеств различных проблем).

Уровень интерфейса пользователя

1. Пользователь определяет текущее и целевое состояния.
2. В соответствии с порядком определения значений факторов, пользователь определяет факторы.
3. Виртуальный эксперт подсказывает пользователю, как измерить или поменять значение факторов, чтобы перевести систему целевое состояние.

Результаты

1. Создана методология разработки онтологий для баз технической поддержки в виде OWL-онтологии (ОБЩАЯ ОНТОЛОГИЯ).
2. Создана ЧАСТНАЯ ОНТОЛОГИЯ для технической поддержки КОМПЬЮТЕРА.
3. Разработан ПРОТОТИП СЕТЕВОЙ ЭКСПЕРТНОЙ СИСТЕМЫ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ <http://semanticweb.dev.juga.ru>.