

УДК 681.324.06

Ю.В. Рогушина¹, А.Я. Гладун²

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ПОИСКУ ВЕБ-СЕРВИСОВ В РАСПРЕДЕЛЕННОЙ СРЕДЕ ИНТЕРНЕТ

Рассмотрев базовые составляющие сервис-ориентированных вычислений (WSDL, UDDI, SOAP) и перспективы их развития, можно сделать вывод о том, что автоматизация компоновки веб-сервисов должна базироваться на семантическом описании их функциональных возможностей на основе онтологий. Предлагаются методы расширения онтологий (в частности, для перехода от онтологии предметной области к онтологии веб-сервиса) и установления подобия между различными онтологиями.

Введение

Различные группы пользователей, занимающиеся обработкой и анализом информации, применяют специальную терминологию, которая используется другими сообществами в ином контексте. В то же время в разных сообществах часто встречаются различные обозначения для одних и тех же понятий. Все это значительно усложняет взаимопонимание между ними. Поэтому так важно разрабатывать модели представления знаний, которые обеспечивали бы автоматизированную обработку информации на семантическом уровне в системах управления знаниями (СУЗ). Основная цель СУЗ – сделать знания доступными и повторно используемыми. Онтологический подход к проектированию СУЗ позволяет создавать системы, в которых знания, накопленные внутри некоторого сообщества, становятся доступными для большинства пользователей. Основные преимущества этого подхода: онтология дает пользователю целостный, системный обзор определенной предметной области (ПрО); знания о ПрО представлены единообразно, что упрощает их восприятие. Сегодня развитие информационных технологий требует СУЗ, ориентированных на поиск и интеграцию сервисов и вычислений, которые распределены в сети Интернет. При этом возникает необходимость интеграции сервисов, предоставляемых разработчиками, которые используют разную терминологию.

1. Постановка задачи

Для того чтобы эффективно использовать веб-сервисы, доступ к которым обеспечивает сеть Интернет, необходимо применять средства, автоматически осуществляющие поиск веб-сервисов, которые соответствуют потребностям пользователя, на семантическом уровне. Цель работы состоит в том, чтобы, проанализировав принципы и технологию сервис-ориентированных вычислений и области применения веб-сервисов, предложить критерий сравнения онтологий, характеризующих предметные области веб-сервисов и интересов пользователей, и метод использования этого критерия для поиска наиболее релевантных сервисов.

2. Концепция сервис-ориентированных вычислений

2.1. История развития веб-сервисов

Концепция веб-сервисов возникла в конце 90-х гг. XX в. В настоящее время архитектура, которую она предлагает, стала отраслевым стандартом в сфере информационных технологий. Сегодня, говоря о веб-сервисах, почти всегда подразумевают решения, основанные на протоколах SOAP, UDDI и WSDL. Основные источники стандартов: OASIS (oasis-open.org), W3C (w3c.org), Web Services Interoperability Organization (ws-i.org), Global Grid Forum (ggf.org), Distributed Management Task Force (dmtf.org), которые специализируются на различных аспектах проблемы. В частности, деятельность рабочих групп WS-Desc и XMLP (W3C) непосредственно связана с разработкой XML-спецификаций для описания веб-сервисов, а Web Security Technical Committee (OASIS) работает над стандартизацией механизма авторизации и аутенти-

фикации с использованием XML. В рамках W3C образована рабочая группа Web Services Architecture Working Group, опубликовавшая ряд документов, в том числе глоссарий терминов из области веб-сервисов. В 2003 г. была опубликована рабочая версия документа «Архитектура веб-сервисов».

Веб-сервисы используют XML для обмена данными между приложениями вне зависимости от их операционной системы, аппаратной платформы и разработчика. Веб-сервис – это набор логически связанных функций, которые могут быть программно вызваны через Интернет. В основе веб-сервисов лежат интернет-стандарты, которые разрабатываются совместно компаниями-разработчиками программного обеспечения. Веб-сервис – это программа, идентифицируемая по URI, интерфейс которой может быть определен в виде XML-конструкций.

2.2. Основные стандарты, которые используют веб-сервисы

Веб-сервисы базируются на трех основных веб-стандартах: SOAP (Simple Object Access Protocol) – протоколе для отправки сообщений по HTTP и другим интернет-протоколам; WSDL (Web Services Description Language) – языке для описания программных интерфейсов веб-сервисов; UDDI (Universal Description, Discovery and Integration) – стандарте индексации веб-сервисов (рис. 1).

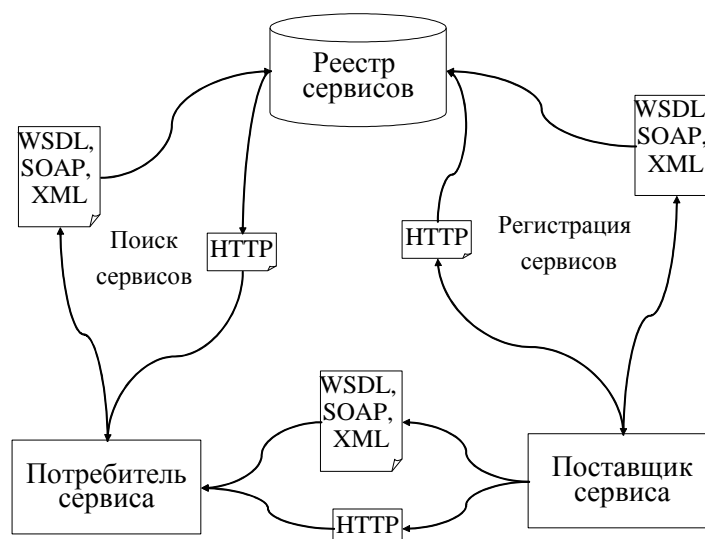


Рис. 1. Взаимодействие между основными участниками сервис-ориентированной архитектуры

Информация о том, какие функции предоставляет данный веб-сервис, содержится в его описании – документе WSDL, а для поиска существующих веб-сервисов предполагается использование специальных реестров, совместимых со спецификацией UDDI. Веб-сервисы представляют собой технологию интеграции интернет-приложений. Другие системы взаимодействуют с веб-сервисом с помощью сообщения в стандарте SOAP. Серверы приложений являются хранилищами веб-сервисов и делают их доступными через протоколы HTTP GET, HTTP POST и HTTP SOAP.

SOAP [1] – стандарт передачи сообщений по Интернету, разработанный фирмой Microsoft для удаленного вызова процедур (RPC, Remote Procedure Call) по протоколу HTTP. Он позволяет передавать информацию по сети в формате XML. При этом могут использоваться любая сеть и любой протокол передачи данных, произвольная информация (заказы, прогноз погоды, выписки об уровне товарных запасов и т. д.), разнообразные вычислительные устройства (в том числе мобильные). Спецификация SOAP определяет «XML-конверт» для передачи сообщений, метод для кодирования программных структур данных в формате XML, а также средства связи по протоколу HTTP.

WSDL [2] – основанный на XML стандарт описания того, как пользоваться сервисом, предложенный консорциумом W3C. Описание веб-сервиса на WSDL содержит технические детали, необходимые для интеграции веб-сервиса в приложение (формат сообщений, операции

и т. п.). На данный момент язык WSDL поддерживают продукт от Microsoft SOAP Toolkit 2.0 (его часть WSDL Generator) и продукт от IBM WSDL Toolkit. Язык описания веб-сервисов (WSDL) только определяет синтаксис того, как веб-сервис может быть вызван, и не говорит ничего о семантике этого сервиса. Это вызывает ряд проблем при использовании веб-сервисов.

Стандарт UDDI [3] предоставляет механизм обнаружения веб-сервисов. UDDI формирует бизнес-реестр (UDDI Business Registry), в котором провайдеры веб-сервисов могут регистрировать свои сервисы, а разработчики – искать необходимые им сервисы. Компании сами регистрируют себя в Business Registry, который представляет собой базу данных общего пользования. UDDI позволяет описывать, интегрировать и публиковать сервисы и сам является особым веб-сервисом, который позволяет пользователям и приложениям находить необходимые им сервисы.

2.3. Сервис-ориентированная архитектура

Сервис-ориентированная архитектура (СОА) – это парадигма проектирования, разработки и управления функциональных модулей (сервисов), каждый из которых доступен через сеть и способен выполнять определенные действия [4]. СОА создает коммуникационную среду для модулей, реализующих прикладную бизнес-логику. Информация о модулях публикуется в такой форме, что их использование не требует знаний о примененных в них решениях и технологиях. От разработчика не требуется знать, как работает программа, необходимо лишь понимать, какие входные и выходные данные нужны и как вызываются эти программы для исполнения. СОА сокращает время и стоимость реализации проектов.

СОА можно реализовать и без использования веб-сервисов, однако сегодня веб-сервисы рассматриваются в качестве основного средства для создания подобной архитектуры. Понятие архитектуры, ориентированной на сервисы, сложилось в ходе развития концепции веб-сервисов, однако существуют и другие подходы к реализации СОА: Java RMI (от Sun Microsystems), CORBA (от консорциума OMG), DCOM (от Microsoft), DCE (предложенный ассоциацией Open Group), программные агенты. СОА имеет следующие характерные особенности: интерфейс функциональных модулей таков, что использование модулей не зависит от технологии или платформы, в рамках которой они реализованы; возможны динамический поиск и подключение нужных функциональных модулей; архитектура базируется на общепринятых отраслевых стандартах.

Сервис-ориентированные вычисления (СОВ) – вычислительная парадигма, которая использует сервисы как фундаментальные элементы для разработки приложений. СОВ базируются на СОА и обеспечивают операции управления сервисами. Разработка системы СОВ – это процесс поиска, подбора и компоновки сервисов, удовлетворяющих требованиям пользователя.

2.4. Компоновка веб-сервисов

Возможность компоновки веб-сервисов часто рассматривают как одно из основных преимуществ их использования. Компоновка состоит в нахождении набора атомарных сервисов, необходимых для реализации функций, которые применяются в запросе пользователя, и определении порядка их выполнения.

Функциональные возможности веб-сервиса определены входами, выходами, предварительными условиями и действиями сервиса. Их обозначают как IOPEs (inputs, outputs, preconditions, effects). Например, для сервиса покупки предварительное условие – это правильный ввод номера кредитной карточки, выход – генерация квитанции, а действие – оплата покупки. Интерактивный книжный магазин может иметь в качестве входов название книги, адрес клиента и номер его кредитной карточки с предварительным условием проверки ее законности. Выходами могут быть электронная квитанция, действия по дебитированию кредитной карточки и отгрузка книги клиенту в его личное владение. Функциональные атрибуты могут описать показатели качества сервиса, например среднее время, затраченное на выполнение покупки в книжном магазине, и время платы за отмену заказа.

Большинство сервисов, необходимых пользователям, формируются вручную с помощью описанных на WSDL описаний атомарных сервисов. Для автоматической компоновки программы должны уметь отбирать нужные веб-сервисы и комбинировать их. Необходимо, чтобы получающийся в результате их комбинирования результат был приемлемым решением поставленной задачи. Информация, содержащаяся в реестре UDDI, недостаточна для автоматической компоновки

веб-сервисов, так как не позволяет интерпретировать их семантику. Поэтому необходимо разрабатывать механизмы отображения семантики сервисов и ее автоматизированного сопоставления с семантикой запросов пользователей. Можно решить проблемы компоновки, связав параметры веб-сервисов с понятиями определенной ПрО и семантическим обоснованием этих понятий [5].

3. Интеллектуальные веб-сервисы и онтологическое представление знаний

Подобно тому, как целью проекта Semantic Web является расширение обычной сети WWW, интеллектуальные веб-сервисы (называемые часто семантическими веб-сервисами, SW-сервисами) [6] расширяют понятие обычных веб-сервисов. Хотя программы могут найти некий веб-сервис в реестре UDDI без помощи человека, они не в состоянии понять, как именно им пользоваться и даже просто для чего он предназначен. Язык описания веб-сервисов WSDL дает инструмент для описания того, каким образом взаимодействовать с тем или иным веб-сервисом, тогда как семантическая разметка предоставляет информацию о том, что и как делает данный сервис. Необходимо снабжать веб-сервисы такими описаниями, чтобы можно было автоматически распознавать их смысл. Ни WSDL, ни UDDI не позволяют программе понять, для чего именно, с точки зрения клиента, служит тот или иной веб-сервис, определить его назначение.

Одним из распространенных средств представления семантики веб-сервисов являются онтологии – представление целостной структуры понятий определенной ПрО в рамках единой системы взаимосвязанных компонентов [7]. В инженерии знаний под онтологией понимается детальное описание некоторой ПрО, которое используется для формального и декларативного определения ее концептуализации. Онтология состоит из терминов, их определений и атрибутов, а также связанных с ними аксиом и правил вывода. *Формальная модель онтологии* O – упорядоченная тройка $O = \langle T, R, F \rangle$, где T – конечное множество терминов ПрО, которое описывает онтологию O ; R – конечное множество отношений между терминами заданной ПрО; F – конечное множество функций интерпретации, заданных на терминах и/или отношениях онтологии O .

Онтологии облегчают компоновку сервисов. Наличие явного представления знаний о ПрО, к которой относится сервис, допускает переформулировку запросов контекстно-зависимым способом и переговоры о возможностях этого сервиса.

Алгоритмы нахождения соответствия между запросом и сервисом, которые используют онтологическое представление знаний, позволяют автоматизировать нахождение семантического подобия между запросом и описанием сервиса, несмотря на синтаксические различия между ними [8]. Для этого запрос согласовывается на основе иерархии понятий предметной области, отображенной в онтологии. Например, запрос об автомобилях соответствует объявлению о транспортных средствах, так как автомобили включены в категорию «транспортные средства». Соответствие между описанием веб-сервиса и запросом обнаруживается, когда все выходы запроса согласованы с выходами описания и все входы описания – со всеми входами запроса, т. е. когда сервис способен удовлетворить потребности запрашивающей стороны и запрашивающая сторона обеспечивает все входы согласованных сервисных потребностей в его действиях.

Самой большой проблемой при обнаружении сервисов является их распределенный характер. Фиксация семантики запросов и исследований сервисов так же, как и контекста предложенного взаимодействия с сервисом, требует адекватных средств представлений сервисов и взаимодействий. Для этого могут быть естественно применены онтологии. Для интероперабельного представления онтологий разработан язык OWL и его модификация для сервисов OWL-S (Web Ontology Language for Services). Интеллектуальный поиск и автоматическая композиция веб-сервисов могут быть сделаны только с использованием таких более мощных возможностей семантического описания веб-сервисов, как предложено в OWL-S. Язык – важный шаг по направлению к более интеллектуальным веб-сервисам. Он обеспечивает онтологическое описание веб-сервиса. Цель разработки OWL-S состоит в том, чтобы сделать возможным использование логического вывода для веб-сервисов, планирование компоновки веб-сервисов, автоматическое применение сервисов программными агентами. OWL-S обеспечивает декларативные описания свойств веб-сервиса и возможности, которые могут использоваться для автоматического обнаружения сервиса.

Используя OWL-S, веб-сервис может сообщать о своих функциональных возможностях потенциальным пользователям. Запрос на обслуживание может быть согласован с объявлениями

веб-сервисов посредством процесса подбора. OWL-S обеспечивает механизм для моделирования бизнес-процессов, но отличается от него по выразительности терминов, представлений, семантики, поддержки поиска и выполнения, обработки ошибок. Описание OWL-S для сервиса состоит из профиля, модели и обоснования сервиса (рис. 2).

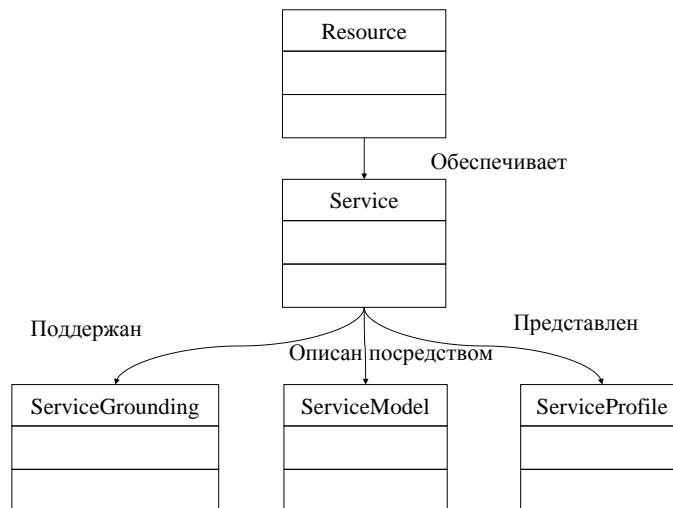


Рис. 2. Представление онтологии веб-сервиса в OWL-S

Профиль сервиса – абстрактная характеристика функций сервиса. Профиль построен на основе контента UDDI, описывающего свойства сервиса, которые необходимы для его автоматического обнаружения, такие, например, как предложение сервиса, его входы и выходы, предварительные условия и дополнительные действия. На основе профиля, который представляет информацию о провайдере, функциональных возможностях и функциональных атрибутах сервиса, могут быть созданы описания и запросы сервиса. По сравнению с UDDI, язык OWL-S имеет большие выразительные возможности.

Для семантического обоснования параметров веб-сервисов используют онтологии различного уровня, для описания семантики источников информации – различные архитектуры.

Подходы, основанные на единой онтологии, применяют одну глобальную онтологию, обеспечивающую совместный словарь для спецификации семантики (рис. 3, а). Все источники информации связаны с глобальной онтологией. Такие подходы могут использоваться в тех случаях, когда источники информации, которые надо объединить, применяют очень похожие взгляды на предметную область.

В *подходах, основанных на множественных онтологиях*, каждый источник информации описан своей собственной онтологией (рис. 3, б). В принципе, каждая из этих прикладных онтологий может быть комбинацией нескольких других онтологий, однако нельзя рассчитывать, что несколько прикладных онтологий используют один и тот же словарь. Так как каждая прикладная онтология может развиваться независимо, отсутствие общего словаря затрудняет их сравнение.

Гибридные подходы подобны подходам, основанным на множественных онтологиях, тем, что семантика каждого исходного текста описана ее собственной онтологией, но для того чтобы сделать локальные онтологии сопоставимыми друг с другом, формируется глобальный словарь для общего использования (рис. 3, в).

Общий словарь (тезаурус) содержит базовые термины (примитивы) предметной области, которые комбинируются в локальные онтологии для того, чтобы описать более сложную семантику. Иногда общий словарь также является онтологией. Подходы, предложенные в настоящее время для исследования семантики сервисов, используют простые архитектуры, которые могут быть классифицированы как подходы, основанные на единой онтологии.

Одна и та же ПрО может иметь несколько онтологий, поскольку информация о ПрО доступна даже экспертам только лишь частично. Любая ПрО характеризуется своей действительностью, т. е. множеством ситуаций, которые имели место в прошлом, имеют место в настоящем

и будут иметь место в будущем [9]. Интеграцию онтологий можно рассматривать как процесс нахождения общности между двумя различными онтологиями A и B и получения новой онтологии C , которая облегчает взаимодействие между компьютерными системами, основанными на онтологиях A и B (в данном случае – между онтологией, на понятия которой ссылаются имена параметров веб-сервиса, и онтологией ПрО, интересующей пользователя). Новая онтология C может быть использована как посредник между системой, основанной на A , и системой, основанной на B . В зависимости от изменений, которые необходимо сделать, чтобы получить C из A и B , можно различать следующие уровни интеграции: соответствие, частичную совместимость, усовершенствование и унификацию. Для того чтобы быть уверенным, что онтологии A и B могут быть интегрированы на некотором уровне, расширение понятий из A и B должно допускать отображение на расширение понятий из C .

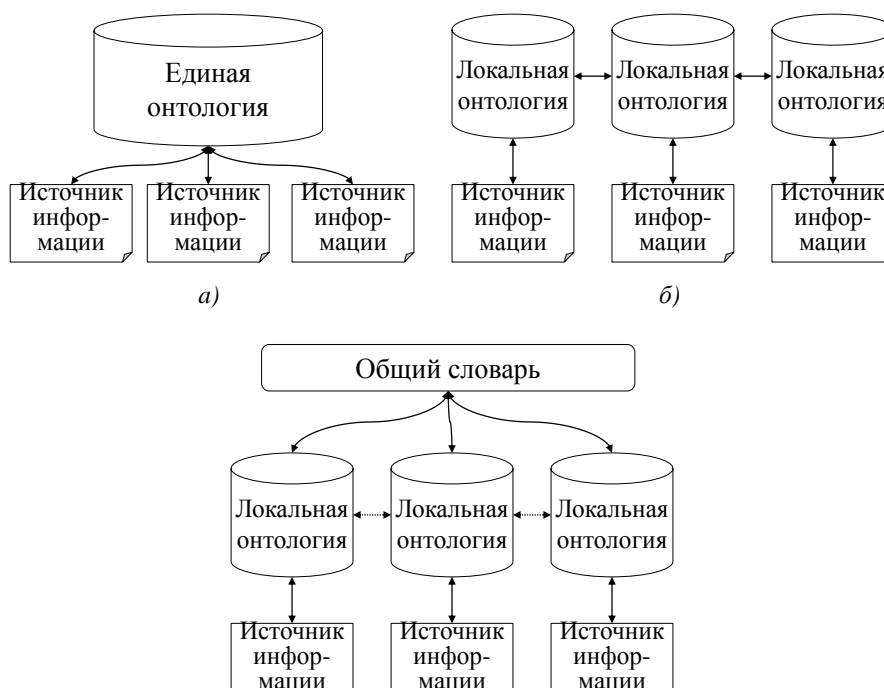


Рис. 3. Различные архитектуры онтологий для информационной интеграции

Соответствие – это отображение понятий и отношений между двумя онтологиями $A = \langle T_A, R_A, F_A \rangle$ и $B = \langle T_B, R_B, F_B \rangle$, которое сохраняет частичный порядок подтипов как в A , так и в B . Если соответствие отображает некоторое понятие $x \in T_A$ или отношение $x \in R_A$ из онтологии A в понятие $y \in T_B$ или отношение $y \in R_B$ в онтологии B , $\exists f(x) = y, \exists g(y) = x$, то говорят, что x и y эквивалентны. Частичная совместимость – это соответствие онтологий A и B , которое поддерживает эквивалентные выводы и вычисления для всех эквивалентных понятий и отношений. Если A и B являются частично совместимыми, то любой вывод или вычисление, выражающие в одной онтологии только соответствующие понятия и отношения, могут быть транслированы в эквивалентный вывод или вычисление в другой онтологии. Соответствие является наиболее слабой формой интеграции: она требует минимальных изменений, но может поддерживать только ограниченные виды взаимодействий. Эта форма интеграции полезна для классификации и информационного поиска, в частности для поиска веб-сервисов, соответствующих запросу, но не поддерживает глубокие выводы и вычисления. Частичная совместимость требует больших изменений, чтобы поддерживать более широкую способность к взаимодействию, хотя могут существовать некоторые понятия или отношения в той или другой системе, которые будут препятствовать полному взаимодействию.

Изложенные выше подходы оставляют открытыми вопросы о том, кто и каким образом формирует онтологии ПрО, с которыми связаны имена параметров веб-сервиса, и как строится описание веб-сервиса в OWL-S (разработчики и провайдеры веб-сервисов не обязаны владеть онтологическим анализом и знать инструментальные средства создания онтологий). В связи с этим актуальной задачей представляется разработка средств и методов автоматизированного формирования онтологий по информационным ресурсам, соответствующим определенному веб-сервису. Важным вопросом является также создание общего словаря (тезауруса) предметной области, обеспечивающего взаимопонимание пользователей и разработчиков веб-сервисов. Кроме того, для нахождения соответствия между онтологиями пользователей и разработчиков веб-сервисов очень актуальна разработка эффективных алгоритмов сравнения онтологий, которые, возможно, являются различными концептуализациями одной и той же ПрО.

В ряде случаев пользователи и разработчики могут воспользоваться готовыми онтологиями, но поиск таких онтологий также является нетривиальной задачей при нестандартном терминологическом базисе [10]. Иногда ПрО-приложения настолько специфичны, что требуют значительного уточнения и расширения самим пользователем. Предлагается использование методов индуктивного обобщения для автоматизированного извлечения онтологических знаний о ПрО из набора информационных ресурсов, релевантных этой ПрО.

4. Алгоритм сравнения запроса пользователя с описанием веб-сервиса

При наличии онтологических описаний как веб-сервисов, так и запросов пользователей возникает проблема их сравнения. Если веб-сервисы и запросы ссылаются на одну онтологию, то можно легко установить, связаны имена параметров атомарных сервисов с одним понятием онтологии или с разными. В противном случае необходимо установить, являются понятия различных онтологий эквивалентными (например, синонимами) или находятся в иерархических отношениях (например, одно понятие является подклассом другого). В общем случае эта задача имеет высокую вычислительную сложность. Рассматриваемый подход к ее решению базируется на предположении, что для описания семантики веб-сервисов используются относительно небольшие и простые по структуре онтологии, терминами которых являются слова одного и того же естественного языка (если описание веб-сервиса дается на языке, не знакомом пользователю, то он не сможет эффективно применить его для своих целей). Предлагается следующий алгоритм нахождения соответствия между запросом пользователя и описанием веб-сервиса. На вход алгоритма поступают формализованные описания онтологий (на языке OWL): множества онтологий веб-сервисов $O_{s[i]}$, доступ к которым обеспечивает пользователю реестр UDDI; онтологии запроса пользователя O_q и онтологии ПрО, для которой $O_{s[i]}$ и O_q являются расширениями. Следует отметить, что в большинстве случаев O_q совпадает с онтологией ПрО, т. е. пользователь просто выбирает одну из онтологий ПрО, на которые ссылаются онтологии доступных веб-сервисов, и только в некоторых случаях он может формировать собственную онтологию (как правило, с помощью объединения нескольких онтологий или упрощения одной из предложенных). Добавлять новые термины нецелесообразно, так как они не используются в описании семантики веб-сервисов, данных их разработчиками. Необходимо найти такой сервис $O_{s[m]}$, для которого $Podob(O_q, O_{s[m]}) = \max_{i=1, n} Podob(O_q, O_{s[i]})$, т. е. коэффициент подобия с запросом пользователя будет максимальным. Вычисление такого коэффициента для каждого из доступных веб-сервисов состоит из следующих этапов.

Этап 1. Построение пересечения множеств терминов онтологий веб-сервиса и запроса $T(O) = T(O_{s[i]}) \cap T(O_q)$.

Этап 2. Если это пересечение не пусто, то построение для каждого термина из $T(O)$ двух множеств – $T_{s[i]}$ - и T_q -терминов, которые связаны с ним в каждой онтологии различными отношениями.

Этап 3. Построение для каждого термина из $T(O)$ пересечения множеств $T_{s[i]}$ и T_q .

Этап 4. Анализ типов отношений между терминами из $T(O)$ и пересечения множеств $T_{s[i]}$ и T_q (все отношения онтологии делятся на иерархические, синонимические и пр.). Для этого используются два словаря отношений: R_{ierarh} и $R_{sinonim}$.

Ситуация 1. Общий термин имеет синонимы в обеих онтологиях. Если для термина $t \in T(O)$ существуют термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$, причем t и t_1 находятся в отношении r_1 в онтологии $T_{s[i]}$, $r_1 \in R_{\text{sinonim}}$, а t и t_2 находятся в отношении r_1 в онтологии T_q , то можно предположить, что термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ также являются синонимами, если онтологии $O_{s[i]}$ и O_q описывают одну и ту же ПрО. Отношение синонимии обладает свойством симметричности, т. е. $\forall r \in R_{\text{sinonim}} r(t_1, t_2) \Leftrightarrow r(t_2, t_1)$. Поэтому если в одной из онтологий между $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ существует отношение $r_1 \in R_{\text{sinonim}}$, то это повышает вероятность того, что рассматриваемые онтологии описывают одну и ту же ПрО. Поэтому значение критерия подобия онтологий следует увеличить на константу K_1 (значение константы зависит от специфики ПрО).

Возможна ситуация, когда в одной из онтологий между $t_2 \in T_{s[i]}$ или $t_1 \in T_q$ существует другое отношение синонимии $r_3 \in R_{\text{sinonim}}$, $r_3 \neq r_1$ (например, «относится к одной группе» вместо «имеют сходные функции»), тогда значение критерия подобия онтологий следует увеличить на константу K_2 , $K_2 \leq K_1$ (значение константы также зависит от специфики ПрО).

Если же между $t_1 \in T_{s[i]}$, $t_2 \in T_{s[i]}$ или $t_1 \in T_q$, $t_2 \in T_q$ существует отношение $r_3 \in R_{\text{ierarh}}$, то это свидетельствует о различии между онтологиями и поэтому значение критерия подобия онтологий следует уменьшить на константу K_3 (значение константы зависит от специфики ПрО). Для простоты формализации будем в дальнейшем считать, что если термины t_1 и t_2 находятся в иерархическом отношении r , то t_1 является подклассом (подмножеством, частью, потомком, филиалом и т. п.) t_2 в зависимости от специфики ПрО.

В случае если между $t_1 \in T_{s[i]}$, $t_2 \in T_{s[i]}$ или $t_1 \in T_q$, $t_2 \in T_q$ существует отношение $r_3, r_3 \notin R_{\text{ierarh}}$, $r_3 \notin R_{\text{sinonim}}$, то изменение коэффициента (увеличение или уменьшение) также зависит от специфики ПрО и устанавливается пользователем для каждого значения отдельно (по умолчанию коэффициент увеличивается на $K_4 < K_2$; например, в задачах, основанных на таксономии ПрО, значение K_2 значительно больше K_1). Отношения между t_1 и t_2 показаны ниже:

$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_1(t_1, t_2) \in O_{s[i]}$,	$K = K + K_1$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_1(t_1, t_2) \in O_q$,	$K = K + K_1$;
$r_1(t, t_1) \in O_{s[i]}$, $r_2(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$, $r_2 \in R_{\text{sinonim}}$,	$r_1(t_1, t_2) \in O_{s[i]}$,	$K = K + K_2$;
$r_1(t, t_1) \in O_{s[i]}$, $r_2(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$, $r_2 \in R_{\text{sinonim}}$,	$r_1(t_1, t_2) \in O_q$,	$K = K + K_2$;
$r_1(t, t_1) \in O_{s[i]}$, $r_2(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$, $r_2 \in R_{\text{sinonim}}$,	$r_2(t_1, t_2) \in O_{s[i]}$,	$K = K + K_2$;
$r_1(t, t_1) \in O_{s[i]}$, $r_2(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$, $r_2 \in R_{\text{sinonim}}$,	$r_2(t_1, t_2) \in O_q$,	$K = K + K_2$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_1, t_2) \in O_{s[i]}$, $r_3 \in R_{\text{sinonim}}$,	$K = K + K_2$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_1, t_2) \in O_q$, $r_3 \in R_{\text{sinonim}}$,	$K = K + K_2$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_1, t_2) \in O_{s[i]}$, $r_3 \in R_{\text{sinonim}}$,	$K = K - K_3$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_2, t_1) \in O_{s[i]}$, $r_3 \in R_{\text{sinonim}}$,	$K = K - K_3$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_1, t_2) \in O_q$, $r_3 \in R_{\text{sinonim}}$,	$K = K - K_3$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_2, t_1) \in O_q$, $r_3 \in R_{\text{ierarh}}$,	$K = K - K_3$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_1, t_2) \in O_{s[i]}$,	$K = K + K_4$;
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3 \notin R_{\text{ierarh}}$, $r_3 \notin R_{\text{sinonim}}$,	
$r_1(t, t_1) \in O_{s[i]}$, $r_1(t, t_2) \in O_q$, $r_1 \in R_{\text{sinonim}}$,	$r_3(t_2, t_1) \in O_{s[i]}$,	$K = K + K_4$;
	$r_3 \notin R_{\text{ierarh}}$, $r_3 \notin R_{\text{sinonim}}$,	

$$\begin{array}{ll}
r_1(t, t_1) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{sinonim}}, & r_3(t_1, t_2) \in O_q, & K = K + K_4; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{sinonim}}, & r_3 \notin R_{\text{ierarh}}, r_3 \notin R_{\text{sinonim}}, & \\
r_1(t, t_1) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{sinonim}}, & r_3(t_2, t_1) \in O_q, & K = K + K_4. \\
r_1(t, t_1) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{sinonim}}, & r_3 \notin R_{\text{ierarh}}, r_3 \notin R_{\text{sinonim}}, &
\end{array}$$

Ситуация 2. Общий термин имеет иерархические отношения в обеих онтологиях. Если для термина $t \in T(O)$ существуют термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$, причем t и t_1 находятся в отношении r_1 в онтологии $T_{s[i]}$, $r_1 \in R_{\text{ierarh}}$, а t и t_2 находятся в отношении r_1 в онтологии T_q , то можно предположить, что термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ также должны находиться в иерархической связи, если онтологии $O_{s[i]}$ и O_q описывают одну и ту же ПрО.

Если между терминами $t_1 \in T_{s[i]}$, $t_2 \in T_{s[i]}$ или $t_1 \in T_q$, $t_2 \in T_q$ существует отношение $r_3 \in R_{\text{ierarh}}$, то это свидетельствует о различии между онтологиями и поэтому значение критерия подобия онтологий следует уменьшить на константу K_2 (значение константы зависит от специфики ПрО). При этом большое значение имеет направленность иерархического отношения между терминами. Поэтому если $t_2 \in T_{s[i]}$ или $t_1 \in T_q$ и в одной из онтологий между ними существует отношение $r_3 \in R_{\text{ierarh}}$, то значение критерия подобия онтологий следует увеличить на константу K_1 (значение константы зависит от специфики ПрО), если отношения направлены в одну сторону, либо уменьшить, если они направлены в различные стороны. Возможные отношения между t_1 и t_2 :

$$\begin{array}{ll}
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_2, t_1) \in O_{s[i]}, & K = K + K_1; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_2, t_1) \in O_q, & K = K + K_1; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_1, t_2) \in O_{s[i]}, & K = K + K_1; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_1, t_2) \in O_q, & K = K + K_1; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_{s[i]}, r_2 \in R_{\text{ierarh}}, & K = K + K_2; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_q, r_2 \in R_{\text{ierarh}}, & K = K + K_2; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_{s[i]}, r_2 \in R_{\text{ierarh}}, & K = K + K_2; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_q, r_2 \in R_{\text{ierarh}}, & K = K + K_2; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_1, t_2) \in O_{s[i]}, & K = K - K_1; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_2, t_1) \in O_q, & K = K - K_1; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_2, t_1) \in O_{s[i]}, & K = K - K_1; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_1(t_2, t_1) \in O_q, & K = K - K_1; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_{s[i]}, r_2 \in R_{\text{ierarh}}, & K = K - K_2; \\
r_1(t, t_1) \in O_{s[i]}, r_1(t_2, t) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_q, r_2 \in R_{\text{ierarh}}, & K = K - K_2; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_{s[i]}, r_2 \in R_{\text{ierarh}}, & K = K - K_2; \\
r_1(t_1, t) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \in R_{\text{ierarh}}, & r_2(t_2, t_1) \in O_q, r_2 \in R_{\text{ierarh}}, & K = K - K_2.
\end{array}$$

Аналогично можно продолжить этот список для случая, когда между $t_1 \in T_{s[i]}$, $t_2 \in T_{s[i]}$ или $t_1 \in T_q$, $t_2 \in T_q$ существуют отношения $r_3 \in R_{\text{ierarh}}$ или $r_3 \notin R_{\text{ierarh}}$, $r_3 \notin R_{\text{sinonim}}$. При наличии между $t_1 \in T_{s[i]}$, $t_2 \in T_{s[i]}$ отношения, не являющегося иерархическим, коэффициент подобия можно не изменять.

Ситуация 3. Общий термин имеет термины, связанные с ним произвольными отношениями, в обеих онтологиях. Если для термина $t \in T(O)$ существуют термин $t_1 \in T_{s[i]}$ и термин $t_2 \in T_q$,

причем t и t_1 находятся в отношении r_1 в онтологии $T_{s[i]}$, а t и t_2 находятся в отношении r_1 в онтологии T_q , $r_1 \notin R_{ierarh}$, $r_1 \notin R_{sinonim}$, то можно предположить, что термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ также являются синонимами, если онтологии $O_{s[i]}$ и O_q описывают одну и ту же ПрО. Поэтому наличие между терминами $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ отношения $r_2 \in R_{sinonim}$ повышает вероятность того, что рассматриваемые онтологии описывают одну и ту же ПрО. Следовательно, значение критерия подобия онтологий следует увеличить на константу K_1 (значение константы зависит от специфики ПрО).

Если же для термина $t \in T(O)$ существуют термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$, причем t и t_1 находятся в отношении r_1 , $r_1 \notin R_{ierarh}$, $r_1 \notin R_{sinonim}$ в онтологии $T_{s[i]}$, а t и t_2 находятся в отношении r_2 , $r_2 \notin R_{ierarh}$, $r_2 \notin R_{sinonim}$, в онтологии T_q , то можно предположить, что термины $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ должны быть каким-либо образом связаны, если входят в одну онтологию. Поэтому наличие между терминами $t_1 \in T_{s[i]}$ и $t_2 \in T_q$ какого-либо отношения r_3 также повышает (хотя и гораздо менее значительно) вероятность того, что рассматриваемые онтологии описывают одну и ту же ПрО. Следовательно, значение критерия подобия онтологий следует увеличить на константу K_2 (значение константы зависит от специфики ПрО). Возможные отношения между t_1 и t_2 приведены ниже:

$$\begin{array}{lll}
 r_1(t, t_1) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \notin R_{ierarh}, r_1 \notin R_{sinonim}, & r_2(t_1, t_2) \in O_{s[i]}, & K = K + K_1; \\
 & r_2 \in R_{sinonim}, & \\
 r_1(t, t_1) \in O_{s[i]}, r_1(t, t_2) \in O_q, r_1 \notin R_{ierarh}, r_1 \notin R_{sinonim}, & r_2(t_1, t_2) \in O_q, r_2 \in R_{sinonim}, & K = K + K_1; \\
 r_1(t, t_1) \in O_{s[i]}, r_1 \notin R_{ierarh}, r_1 \notin R_{sinonim}, & & \\
 r_2(t, t_2) \in O_q, r_2 \notin R_{ierarh}, r_2 \notin R_{sinonim}, & r_3(t_1, t_2) \in O_{s[i]}, & K = K + K_2; \\
 r_1(t, t_1) \in O_{s[i]}, r_1 \notin R_{ierarh}, r_1 \notin R_{sinonim}, & & \\
 r_2(t, t_2) \in O_q, r_2 \notin R_{ierarh}, r_2 \notin R_{sinonim}, & r_2(t_1, t_2) \in O_q, & K = K + K_2.
 \end{array}$$

Этап 5. Строится коэффициент сходства онтологий, который является количественным отображением сходства семантики двух онтологий. При этом учитываются следующие факторы: вхождение одного и того же термина в обе онтологии, нахождение двух терминов в разных онтологиях в одном и том же отношении, нахождение двух терминов в разных онтологиях в отношениях одного типа или разных (например, в иерархическом отношении и отношении синонимии), существование любых отношений (прямых или опосредствованных) между одними и теми же терминами. Для этого используются статистические методы, нечеткая логика, интенциональные отношения и эмпирические правила.

Этап 6. Строится коэффициент подобия запроса и веб-сервиса аналогично этапу 5, но учитываются только те термины из $T(O) = T(O_s) \cap T(O_q)$, на которые ссылаются имена параметров веб-сервиса. Если полученный коэффициент выше определенной пользователем константы – коэффициента доверия, то считается, что веб-сервис удовлетворяет потребностям пользователя и может использоваться при компоновке составного веб-сервиса.

Заключение

Рассмотрев базовые составляющие сервис-ориентированных вычислений в распределенной среде Интернет (WSDL, UDDI, SOAP) и проанализировав перспективы их развития, можно сделать выводы о том, что автоматизация поиска и компоновки веб-сервисов, обеспечивающая значительно более широкое применение этих составляющих, должна базироваться на семантическом описании их функциональных возможностей. Сегодня описание семантики веб-сервисов, как и многих других информационных ресурсов распределенной гетерогенной среды Интернет, связывают с онтологическим подходом к представлению знаний (OWL-S). Однако открытыми остаются как вопросы создания онтологий, адекватно отражающих специфику определенных предметных областей, так и проблемы, связанные со сравнением и установлением соответствий

между различными онтологиями. В данной работе предложено несколько подходов к расширению онтологий (в частности, для перехода от онтологии ПрО к онтологии приложения – веб-сервиса) и к установлению подобия между различными онтологиями.

Список литературы

1. SOAP [Electronic resource]. – Режим доступа: w3c.org/soap/soap.htm.
2. Язык WSDL [Электронный ресурс]. – Режим доступа: http://allxml.h1.ru/articles/SOAP_rpc_wsdl_2.html.
3. UDDI (Universal Description, Discovery, and Integration) [Electronic resource]. – Режим доступа: <http://www.uddi.org/>.
4. Шелякин, П. Архитектуры, ориентированные на сервисы / П. Шелякин [Электронный ресурс]. – Mode of access: <http://xmlhack.ru/texts/COAvsWebservices/COAvsWebservices.html>.
5. Probst, F. Giving Meaning to GI Web-сервис Descriptions / F. Probst, M. Lutzand [Electronic resource]. – Mode of access: <http://musil.uni-muenster.de>.
6. Cowles, P. Web Services and the Semantic Web / P. Cowles [Electronic resource]. – Mode of access: http://ezolin.pisem.net/logic/ws_and_sw_rus.html.
7. Gladun, A. Knowledge Management in the Clinical Multiagent Information System / A. Gladun, J. Rogushina // Proc. of International Conf. «Advanced Information and Telemedicine Technologies for Health» (AITTH`2005). – Vol. 2. – Minsk, 2005. – P.76–80.
8. Describing Web Services using OWL-S and WSDL / D. Martin [et al.] [Electronic resource]. – Mode of access: <http://www.daml.org/services/owl-s/1.0/owl-s-wsdl.html>.
9. Клещев, А.С. Отношения между онтологиями предметных областей. Ч. 1. Онтологии, представляющие одну и ту же концептуализацию. Упрощение онтологии / А.С. Клещев, И.Л. Артемьева // Информационный анализ. – Вып. 1. – Сер. 2. – 2002. – С. 4–9.
10. Гладун, А.Я. Онтологии как перспективное направление интеллектуализации поиска информации в мультиагентных системах е-коммерции / А.Я. Гладун, Ю.В. Рогущина // Proc. of XI-th International Conference «Knowledge-Dialogue-Solution». – Vol. 1. – Varna, 2005. – P. 158–165.

Поступила 17.01.06

¹Институт программных систем НАН Украины,
Киев, пр. Академика Глушкова, 40
e-mail: _jjj_@ukr.net

²Международный научно-учебный центр
информационных технологий и систем НАНУ и МОНУ,
Киев, пр. Академика Глушкова, 40
e-mail: glanat@yahoo.com

Yu.V. Rogushina, A.Ya. Gladun

ONTOLOGICAL APPROACH TO SEARCH OF WEB-SERVICES IN THE DISTRIBUTED INTERNET ENVIRONMENT

Considering the base components of service-oriented computing (WSDL, UDDI, SOAP) and future trends of their development, it is possible to conclude that automation of Web-services composition has to be based on semantic description of their functionality on the ontology base. The methods of ontology expansion (in particular, for pass from domain ontology to ontology of Web-service) and similarity establishment between different ontologies are proposed.