

**КРАТКИЕ СООБЩЕНИЯ
BRIEF PAPERS**

УДК 004.65

КОМПОНЕНТНАЯ АРХИТЕКТУРА СИСТЕМЫ ДОСТУПА К ВЕБ-СЕРВИСАМ**А.М. Дергачев^а, А.А. Дергачев^а**^а Университет ИТМО, Санкт-Петербург, Россия, dam600@mail.ru

Представлена компонентная архитектура системы организации доступа к веб-сервисам. Разработано исполнительное ядро системы – процессор запросов к веб-сервисам. Реализован прототип типового решения проектирования и развертывания инфраструктуры управления веб-сервисами интернет-ориентированных информационно-вычислительных систем. Работа направлена на развитие исследований, проводимых в области формирования плана выполнения запросов к веб-сервисам – плана вызова веб-сервисов. Основными результатами работы являются: компонентная архитектура системы организации доступа к веб-сервисам, прототип типового решения проектирования – процессор запросов к веб-сервисам.

Ключевые слова: веб-сервис, типовое решение проектирования, интернет, среда выполнения.

COMPONENT ARCHITECTURE OF WEB SERVICES ACCESS MANAGEMENT SYSTEM**A.M. Dergachev, A.A. Dergachev**^а ITMO University, Saint Petersburg, Russia, dam600@mail.ru

The component architecture of Web services access management system is presented. The executive kernel of the system is designed –processor of requests to web services. The prototype of design pattern is implemented – the infrastructure of Web services management system for Internet-based information systems. The work is aimed at developing research in the field of formation of Web services requests submission plan, which is named as Web services execution plan. The main results of the work are: component architecture of Web services access management system; prototype of the design pattern –processor of requests to Web services.

Keywords: Web service, design pattern, Internet, runtime environment.

Формальное представление веб-сервисов [1] и концептуальное расширение их функционального описания [2] позволяют единожды формализовать план выполнения запроса в терминах формальных операций и использовать его в дальнейшем при формировании плана вызова веб-сервисов информационно-вычислительных систем, например, распределенных систем анализа данных [3]. Прозрачность формирования плана вызова конкретных экземпляров веб-сервисов может быть обеспечена программной средой выполнения, реализующей соответствующие алгоритмы поиска и выбора сервисов-исполнителей на основе параметров качества обслуживания веб-сервисов [4–6]. Разработка типового решения проектирования среды выполнения может стать основой компонентной реализации системы доступа к веб-сервисам и развертывания на ее основе сервис-ориентированной инфраструктуры интеграции информационно-вычислительных систем [7, 8].

Для программной реализации среды выполнения и проведения эксперимента была разработана представленная на рисунке архитектура системы доступа к динамически меняющемуся составу опубликованных в Интернет веб-сервисов. Описание веб-сервисов, созданных для проведения эксперимента, выполнено на стандартном языке WSDL. Формальные операции, выполняемые веб-сервисами, представлены в двух форматах: без параметров и с входными/выходными параметрами. Например, для веб-сервисов индустрии приборостроения операция «разработать печатную плату» в качестве входного параметра использует принципиальную схему, а в качестве выходного – технологический файл с программой управления производством печатной платы. Для веб-сервисов корпоративных информационных систем страховых компаний операция «получить страховой полис» в качестве входного параметра может использовать код региона, а в качестве выходного – страховой полис. Для хранения WSDL-описаний использованы UDDI-реестры, являющиеся стандартом для публикации-обнаружения веб-сервисов. Взаимодействие с сервисами осуществляется с помощью SOAP-сообщений (XML-сообщений по SOAP-протоколу) поверх протокола HTTP. Ядром реализации системы доступа является разработанный процессор запросов, выполняющий следующие функции: преобразование операций, поиск веб-сервисов, формирование реестра оценок качества обслуживания веб-сервисов [1], формирование и выполнение плана вызова веб-сервисов.

Процессор запросов к веб-сервисам разработан по компонентной технологии и состоит из пяти независимых функционально законченных программных модулей: локатора сервисов, преобразователя операций, монитора транзакций, оптимизатора запросов и исполнителя планов [6]. Локатор сервисов осуществляет выборку WSDL-описаний из UDDI-реестров. Реализация клиентской части модуля выполнена с использованием программного интерфейса WASP UDDI API компании Systinet для разработанного ею платформенно-независимого UDDI-реестра, рассчитанного на использование внутри организации или между организациями-партнерами внутри «виртуального» предприятия. Преобразователь операций в

соответствии с формализованным планом вызова веб-сервисов, являющимся графом зависимостей выполнения формальных операций, выбирает из полученных локатором сервисов WSDL-описаний реальные операции-кандидаты для формирования реального плана вызова веб-сервисов в соответствии с вариантами и типами совпадений. Монитор транзакций отвечает за сбор информации о выполнении каждой реальной операции конкретного веб-сервиса. Целью сбора информации является накопление значений показателей качества обслуживания веб-сервиса в локальном хранилище данных реляционного типа, являющемся частью конкретного экземпляра процессора сервисных запросов. Оптимизатор запросов является главным модулем исполнительного ядра системы организации доступа к веб-сервисам – процессора запросов к веб-сервисам. Его задача состоит в формировании реального плана вызова веб-сервисов с лучшими показателями качества обслуживания. Исполнитель планов осуществляет вызов веб-сервисов посредством SOAP-сообщений для выполнения операций согласно сформированному оптимизатором реальному плану вызова веб-сервисов.



Рисунок. Архитектура системы организации доступа к веб-сервисам

Для подтверждения работоспособности разработанного по компонентной технологии процессора запросов к веб-сервисам был проведен эксперимент. Алгоритмы формирования плана вызова веб-сервисов – алгоритм полной оптимизации (АПО), алгоритм оптимизации операций (АОО) и алгоритм вероятностной оценки (АВО) [4] – реализованы в виде динамически загружаемых программных модулей. При оценке работоспособности процессора запросов к веб-сервисам учитывалось время выполнения различных алгоритмов формирования плана вызова веб-сервисов в зависимости от количества экземпляров реальных операций на одну формальную операцию, сравнивалось время выполнения различных алгоритмов формирования плана вызова сервисов между собой, а также проводилось сравнение идентичности планов, полученных разными алгоритмами. Эксперимент проводился на компьютере, оснащенный процессором AMD Phenom II Six-Core 1075T с тактовой частотой 3 ГГц, 16 ГБ оперативной памяти, жестким диском емкостью 1 ТБ, под управлением операционной системы Solaris 10. Программная реализация системы организации доступа к веб-сервисам развертывалась на сервере приложений Java EE GlassFish 3.1 и СУБД Oracle 10g.

Эксперимент показал, что с увеличением количества реальных операций на одну формальную операцию характер зависимости времени выполнения алгоритмов от количества формальных операций, приходящихся на один запрос, не меняется. При средней сложности запроса, требующего выполнения от 15 до 20 операций, приходящихся на один запрос, время формирования плана вызова веб-сервисов с заданными параметрами качества обслуживания веб-сервисов и принятие решения о его использовании несравнимо меньше, чем поиск и выбор исполнителя из множества конкурирующих веб-сервисов непосредственно пользователем информационно-вычислительной системы. Сравнение идентичности планов вызова веб-сервисов, полученных с использованием статистических данных по алгоритмам АПО и АОО, с планом вызова веб-сервисов, полученного с учетом вероятностной оценки показателей качества обслуживания веб-сервисов по алгоритму АВО, позволяет сделать вывод о целесообразности использования

последнего, так как он позволяет более точно прогнозировать динамику изменения показателей качества обслуживания веб-сервисов, и, как следствие, более объективно выбирать исполнителя из множества конкурирующих веб-сервисов.

Полученные результаты позволяют считать предложенную в работе компонентную архитектуру системы организации доступа к веб-сервисам перспективной, а реализованный процессор запросов к веб-сервисам – прототипом типового решения проектирования и развертывания инфраструктуры управления веб-сервисами Интернет-ориентированных информационно-вычислительных систем. Реализация алгоритмов, представленная загружаемыми модулями процессора запросов к веб-сервисам, позволяет динамически, без перекомпиляции ядра, подгружать для формирования плана вызова веб-сервисов вновь разработанные или улучшенные алгоритмы, что делает предложенную в работе архитектуру перспективной в плане ее открытости и масштабируемости.

1. Дергачев А.М. Проблемы эффективного использования сетевых сервисов // Научно-технический вестник СПбГУ ИТМО. 2011. № 1 (71). С. 83–87.
2. Дергачев А.А. Концептуальное расширение функционального описания веб-сервисов // Научно-технический вестник информационных технологий, механики и оптики. 2014. № 2 (90). С. 174–176.
3. Дергачев А.А. Анализ данных на основе платформы SQL-MAPREDUCE // Научно-технический вестник информационных технологий, механики и оптики. 2014. № 1 (89). С. 66–71.
4. Yu T., Zhang Y., Lin K.J. Efficient algorithms for Web services selection with end-to-end QoS constraints // ACM Transactions on the Web. 2007. V. 1. N 1. art. N 6.
5. Dover D., Dafforn E. Search Engine Optimization Secrets. Indianapolis: Wiley Publishing, Inc., 2011. 456 p.
6. Ouzzani, M., Bouguetta A. Semantic Web Services for Web Databases. Springer Science+Business Media, 2011. 155 p.
7. Марьин С.В., Ковальчук С.В. Сервисно-ориентированная платформа исполнения композитных приложений в распределенной среде // Изв. вузов. Приборостроение. 2011. Т. 54. № 10. С. 21–29.
8. Семерханов И.А., Муромцев Д.И. Интеграция информационных систем на основе технологии связанных данных // Научно-технический вестник информационных технологий, механики и оптики. 2013. № 5 (87). С. 123–127.

<i>Дергачев Андрей Михайлович</i>	– кандидат технических наук, доцент, Университет ИТМО, Санкт-Петербург, Россия, dam600@mail.ru
<i>Дергачев Александр Андреевич</i>	– аспирант, Университет ИТМО, Санкт-Петербург, Россия, dergachev.itmo@mail.ru
<i>Andrey M. Dergachev</i>	– Associate professor, Ph.D., ITMO University, Saint Petersburg, Russia, dam600@mail.ru
<i>Alexander A. Dergachev</i>	– postgraduate, ITMO University, Saint Petersburg, Russia, dergachev.itmo@mail.ru

Принято к печати 04.04.14

Accepted 04.04.2014