

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 004.383

Скакун
Алексей Сергеевич

Модели и алгоритмы удалённого вызова команд

АВТОРЕФЕРАТ

на соискание степени магистра технических наук

по специальности 1-40 80 05 Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

Научный руководитель
Бахтизин Вячеслав Вениаминович
к.т.н., доцент

Минск 2016

КРАТКОЕ ВВЕДЕНИЕ

В связи с широким распространением «облачных» технологий и сервисов всё больше становится популярным разрабатывать веб-приложения в виде двух и более отдельных приложений-компонентов, которые обмениваются между собой данными посредством удалённого вызова команд. Такой подход показал на практике, что при разработке можно задействовать разные команды программистов для реализации отдельных приложений-компонентов, что улучшает качество этих приложений и скорость разработки в целом. Но возникает необходимость разработки методов взаимодействия между этими приложениями, что вынуждает разработчиков тратить дополнительное время при разработке ПО на согласование формата команд и выбор подходящей для этого технологии. Наличие какого-либо универсального решения значительно упростило бы процесс разработки и сократило бы временные затраты, но, к сожалению, это невозможно по ряду причин. Одной из самых главных причин является большое разнообразие языков программирования и программных платформ, а также банальное несоблюдение существующих стандартов удалённого вызова команд, т.е. нет никаких гарантий что новый единый стандарт соблюдался бы всеми разработчиками и поддерживался бы большим количеством программных платформ.

Частичным решением данной проблемы может стать определённый подход к разработке API, который бы позволил программистам не задумываться об используемой технологии удалённого вызова команд. Наиболее популярные программные средства, позволяющие реализовать удалённый вызов команд, на сегодняшний день не дают возможности реализовать такой подход, т.к. в их основе, как правило, лежит какая-то одна технология удалённого вызова команд, с которой приходится работать программисту, при этом ответственность за соблюдение стандартов используемой технологии лежит на плечах разработчика.

Таким образом, необходимо пересмотреть существующий подход к разработке API, и разработать новые модели и алгоритмы, согласно которым будут разработаны новые программные средства, берущие под контроль соблюдение стандартов и позволяющие использовать сразу несколько технологий удалённого вызова команд параллельно. Это решит проблему временных затрат на разработку API, упростит сопровождение программного кода, обеспечит более гибкое общение между клиентами и сервером, а также устранит ряд возможных ошибок при реализации API. Исходя из этого, тема магистерской диссертации является актуальной.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Цель и задачи исследования

Целью диссертационной работы является разработка моделей и алгоритмов удалённого вызова команд и проведение практической проверки наличия преимуществ разработанных моделей и алгоритмов в рамках созданного на их основе программного средства (ПС) удалённого вызова команд над уже существующими аналогами.

Для достижения поставленной цели необходимо решить следующие задачи:

- 1) Исследовать существующие модели, алгоритмы и ПС для удалённого вызова команд с целью определения проблем организации удалённого вызова команд.
- 2) Разработать модели и алгоритмы удалённого вызова команд, предлагающие решения выявленных проблем.
- 3) Разработать ПС на основе созданных моделей и алгоритмов удалённого вызова команд.
- 4) Провести экспериментальные исследования разработанного ПС с целью проверки улучшения значений критериев эффективности у разработанного ПС по сравнению с исследованными ПС для организации удалённого вызова команд.

Объектом исследования является область клиент-серверных веб-приложений.

Предметом исследования является моделирование и алгоритмизация удалённого вызова команд.

Основной гипотезой, положенной в основу диссертационной работы, является улучшение значений критериев эффективности у разработанного ПС по сравнению с исследованными ПС для организации удалённого вызова команд. Разработанные модели и алгоритмы удалённого вызова команд предлагают более эффективные решения проблем организации удалённого вызова команд, чем уже существующие модели и алгоритмы.

Связь работы с приоритетными направлениями научных исследований и запросами реального сектора экономики

Работа выполнялась в соответствии с научно-техническими заданиями и планами работ кафедры «Программное обеспечение информационных технологий» по теме «Разработать модели, методы, алгоритмы для оценки параметров, повышения надежности и качества функционирования аппаратно-программных средств систем и сетей сложной конфигурации и внедрить в современные

обучающие комплексы» (ГБ № 11-2004, № ГР 20111065, научный руководитель НИР – В. В. Бахтизин).

Личный вклад соискателя

Результаты, приведенные в диссертации, получены соискателем лично. Вклад научного руководителя В.В. Бахтизина заключается в формулировке целей и задач исследования.

Апробация результатов диссертации

Основные положения диссертационной работы докладывались и обсуждались на 51-ой научной конференции аспирантов, магистрантов и студентов по направлению «Компьютерные системы и сети: материалы конференции» (Минск, Беларусь, 2015); Международной научно-методической конференции «Дистанционное обучение – образовательная среда XXI века» (Минск, Беларусь, 2015).

Опубликованность результатов диссертации

Скакун, А. С. Модель единого центра обработки удалённых команд / А. С. Скакун, В. В. Бахтизин // Дистанционное обучение – образовательная среда XXI века: материалы IX международной научно-методической конференции (Минск, 3-4 декабря 2015 года). – Минск: БГУИР, 2015. – С. 288.

Скакун, А. С. Единый центр обработки удалённых команд / А. С. Скакун, В. В. Бахтизин // 51-я научная конференция аспирантов, магистрантов и студентов по направлению 4: Компьютерные системы и сети: материалы конференции (Минск, 13-17 апреля 2015 года). – Минск: БГУИР, 2015. – С. 91.

Скакун, А. С. Организация единого центра обработки удалённых команд / А. С. Скакун, В. В. Бахтизин // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: Материалы XVIII Республиканской научной конференции студентов и аспирантов (Минск, 21-25 марта 2015 года). – Гомель: ГГУ им. Ф. Скорины, 2015. – С. 190.

Структура и объем диссертации

Диссертация состоит из общей характеристики работы, введения, трех глав, заключения, списка использованных источников, списка публикаций соискателя и приложений. В первой главе представлен анализ предметной области, выявлены

основные существующие проблемы в рамках тематики исследования, показаны направления их решения. Вторая глава посвящена разработке моделей (модель предоставления доступа к команде через REST-запрос, модель трансляции ошибки в формат запроса, модель распознавания используемого формата вызова команд) и алгоритмов (алгоритм определения используемого HTTP-метода, алгоритм чтения аргументов функции из REST-запроса) для организации удалённого вызова команд. В третьей главе построена система критериев оценки эффективности ПС для организации удалённого вызова команд, разработано ПС на основе созданных моделей и алгоритмов удалённого вызова команд; проведены эксперименты по проверке улучшения значений критериев оценки эффективности у разработанного ПС по сравнению с исследованными ПС для организации удалённого вызова команд.

Общий объем работы составляет 59 страниц, из которых основного текста – 47 страниц, 10 рисунков на 9 страницах, 4 таблицы на 2 страницах, список использованных источников из 12 наименований на 1 странице.

Библиотека БГУИР

ОСНОВНОЕ СОДЕРЖАНИЕ

Во **введении** определена область и указаны основные направления исследования, показана актуальность темы диссертационной работы, дана краткая характеристика исследуемых вопросов, обозначена практическая ценность работы.

В **первой главе** проведен анализ литературы по теме удалённого вызова команд. Сформулированы основные определения области удалённого вызова команд

Детально рассмотрен протокол HTTP, который играет большую роль в обеспечении общения между серверами и клиентскими устройствами. Проанализирован процесс формирования запроса и получения ответа. А также рассмотрены такие аспекты как, защита и безопасность, дополнительные возможности протокола.

Отдельно проанализирован принцип удалённого вызова команд. Рассмотрены характерные черты удалённого вызова, процесс формирования таких вызовов. Также выделены проблемы, которые присущи данной технологии, такие как: неоднородность языков программирования, отсутствие единого решения для разных программных платформ.

Анализируя технологии, реализующие удалённый вызов команд, были выделены наиболее популярные и часто используемые технологии. Среди них – JSON-RPC, XML-RPC и REST. Общим недостатком является необходимость сериализации данных перед отправкой и десериализации при получении команды. Также определены достоинства каждой технологии.

Рассмотрены популярные программные средства, которые реализуют возможность удалённого вызова команд. Рассмотрена их доступность, простота использования и предоставляемый функционал. Выделен ряд достоинств и недостатков. Рассмотренные программные средства предоставляют хороший инструментарий для реализации удалённого вызова команд, но не осуществляют контроля над исполнением. Из-за этого повышается риск нарушения стандартов и возможность допущения ошибок в тех местах, которые могли быть автоматизированы.

Исходя из проведённого анализа сформулированы задачи исследования: разработать модели и алгоритмы, которые помогут ускорить разработку API, обеспечат соблюдение стандартов, предоставят возможность программистам разрабатывать API, не привязывая его к определённой технологии вызова команд, упростят рефакторинг и сопровождение ПО, а также провести экспериментальные исследования полученных результатов.

Вторая глава посвящена разработке моделей и алгоритмов удалённого вызова команд.

Разработанные модели и алгоритмы помогут вести разработку API, не привязываясь к конкретному формату, что значительно сократит время разработки, обеспечит соблюдение стандартов и предотвратит ряд ошибок при обработке поступающих команд.

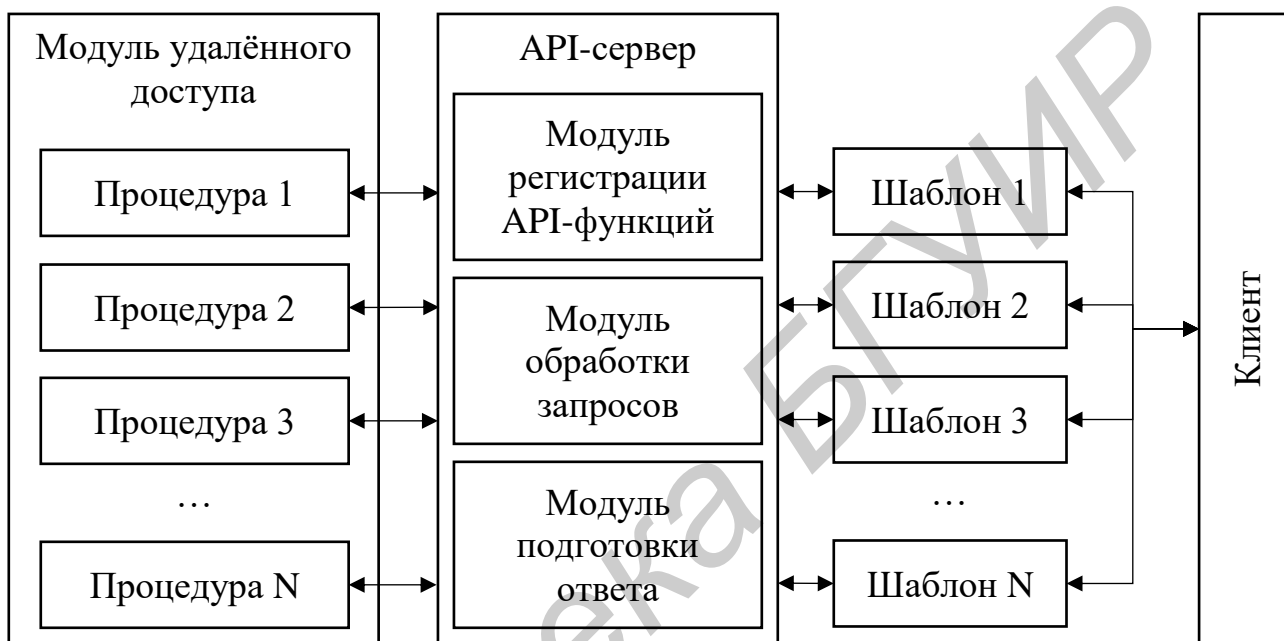


Рисунок 1 - Схема модели предоставления доступа к команде через REST-запрос

Модель предоставления доступа к команде через REST-запрос освободит разработчиков от необходимости составлять шаблоны запросов, которые теперь будут формироваться автоматически, основываясь на уникальных характеристиках команды. Такой подход значительно повысит расширяемость программного кода и ускорит разработку API.

Модель трансляции ошибки в формат запроса поможет избежать нарушения стандартов при отправке клиенту ошибочного результата. Основой алгоритма является работа со словарями кодов ошибок, с помощью которых будет осуществляться перевод ошибки выполнения команды в ошибку, предусмотренную стандартом используемого формата.

Модель распознавания используемого формата вызова команд определяет отдельные модули, которые отвечают только за один конкретный формат. Данная модель делает возможным добавление дополнительных форматов в дальнейшем и

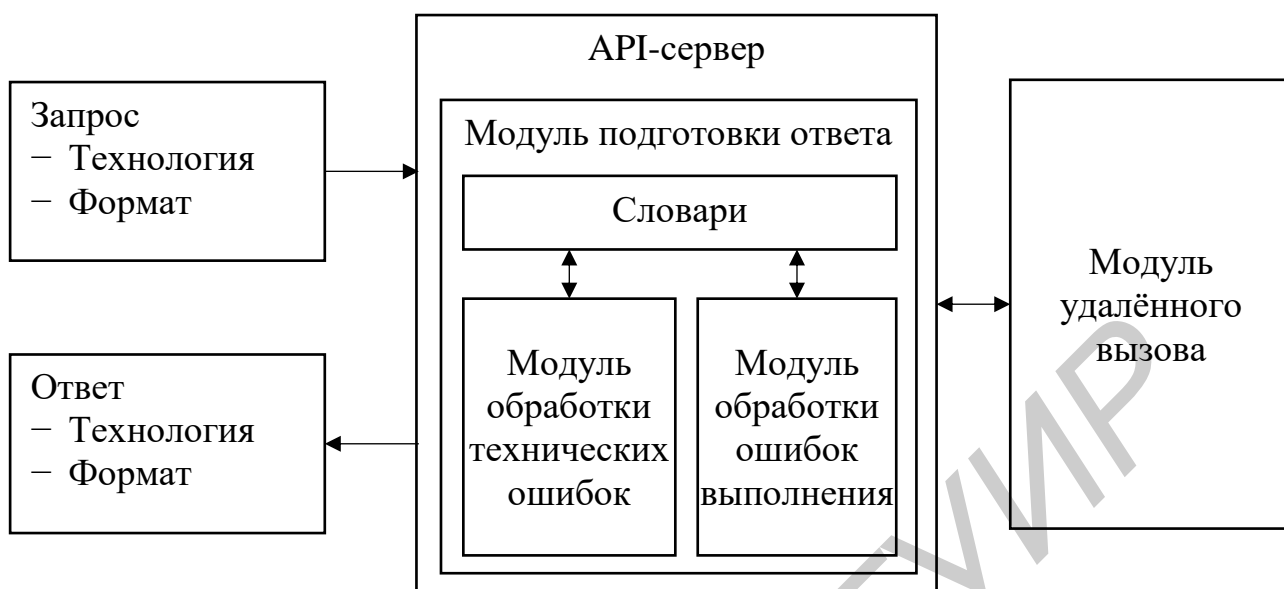


Рисунок 2 - Схема модели трансляции ошибки в формат запроса

изолирует работу с каждым форматом в отдельном модуле, что снижает количество потенциально возможных ошибок.

Алгоритм определения используемого HTTP-метода является частью реализации модели предоставления доступа к команде через REST-запрос. Исходя из того, что разработчикам не требуется составлять шаблоны запросов, нет необходимости упоминать где-либо в программном коде явные значения HTTP-методов. Данный алгоритм определяет необходимый HTTP-метод по названию вызываемой функции или по указанной операции, которая выбирается из предложенных

Алгоритм чтения аргументов функции из REST-запроса автоматизирует извлечение значений переданных аргументов функции из HTTP-запроса. Данный алгоритм учитывает тот факт, что иногда (в частности при операции чтения) один из аргументов, отвечающий за идентификацию, может быть передан не в теле запроса, а в качестве дополнительного фрагмента URL-адреса. Поэтому алгоритм предусматривает особую обработку аргумента с именем id.

В третьей главе разработано ПС для организации удалённого вызова команд, представлены результаты экспериментального исследования разработанного ПС.

Согласно разработанным моделям и алгоритмам было спроектировано программное средство для платформы Node.js, с помощью которого был реализован механизм удалённого вызова команд. В ходе эксперимента было определено, что разработанное программное средство обладает рядом достоинств на фоне других ПС.

Таблица 1 - Количество строк кода для разработанного ПС и Express (Restify) по различному назначению.

Характеристики программного кода	Разработанное ПС	Express (Restify)
Строк кода для всего приложения	48	110
Строк кода для подключения модулей	4	2
Строк кода для инициализации	7	5
Строк кода для реализации доступа к командам	35	100
Строк кода для реализации REST	2	14
Строк кода для реализации JSON-RPC	2	49

Увеличилась скорость разработки API. Аналогичный по функциональности программный код при использовании разработанного ПС занимает 49 строк против 110 при использовании Express или Restify. Что быстрее более чем в 2 раза.

Эксперимент показал, что разработанное ПС позволяет вести разработку API не привязываясь к конкретной технологии удалённого вызова команд. Экспериментально это выразилось в очень малом количестве программного кода, необходимого для реализации конкретной технологии RPC. Используя разработанное ПС для этой задачи требуется всего 2 строки кода для каждой технологии. При использовании Express или Restify для реализации технологий REST и JSON-RPC потребовалось 14 и 49 строк кода соответственно.

Малое количество строк кода для реализации каждой технологии RPC и модульная система разработанного ПС снижают возможность появления ошибок в

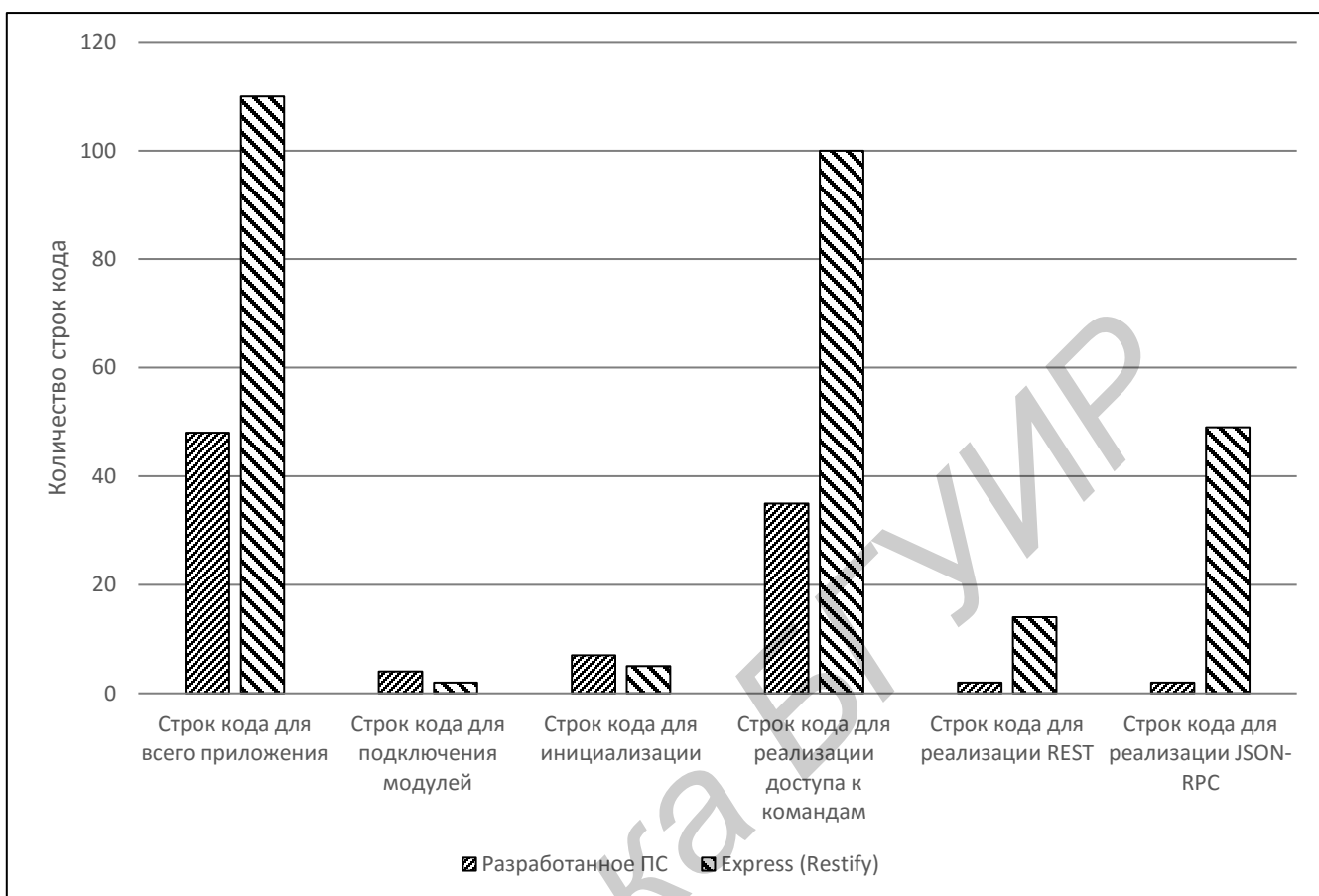


Рисунок 3 - Сравнение скорости разработки API с использованием разработанного ПС и Express (Restify)

реализации технологий RPC и обеспечивают соблюдение стандартов используемых технологий.

Разработанное программное средство представляет собой лишь связующее звено между уже имеющимися программными модулями и веб-сервером, в котором реализована возможность удалённого вызова команд. Таким образом программисту не нужно разрабатывать дополнительную логику работы с сервером и технологиями RPC, что значительно упрощает масштабируемость разрабатываемого API и облегчает его рефакторинг.

ЗАКЛЮЧЕНИЕ

Основные научные результаты диссертации

Проведён анализ литературы по теме удалённого вызова команд. Рассмотрены основные технологии, используемые для осуществления удалённого вызова команд.

Детально рассмотрен протокол HTTP, который играет большую роль в обеспечении общения между серверами и клиентскими устройствами. Проанализирован процесс формирования запроса и получения ответа. А также рассмотрены такие аспекты как, защита и безопасность, дополнительные возможности протокола.

Отдельно проанализирован принцип удалённого вызова команд. Рассмотрены характерные черты удалённого вызова, процесс формирования таких вызовов. Также выделены проблемы, которые присущи данной технологии, такие как: неоднородность языков программирования, отсутствие единого решения для разных программных платформ.

Анализируя технологии, реализующие удалённый вызов команд, были выделены наиболее популярные и часто используемые технологии. Среди них – JSON-RPC, XML-RPC и REST. Общим недостатком является необходимость сериализации данных перед отправкой и десериализации при получении команды. Также определены достоинства каждой технологии.

Рассмотрены популярные программные средства, которые реализуют возможность удалённого вызова команд. Рассмотрена их доступность, простота использования и предоставляемый функционал. Выделен ряд достоинств и недостатков. Рассмотренные программные средства предоставляют хороший инструментарий для реализации удалённого вызова команд, но не осуществляют контроля над исполнением. Из-за этого повышается риск нарушения стандартов и возможность допущения ошибок в тех местах, которые могли быть автоматизированы.

Исходя из проведённого анализа сформулированы задачи исследования: разработать модели и алгоритмы, которые помогут ускорить разработку API, обеспечат соблюдение стандартов, предоставят возможность программистам разрабатывать API, не привязывая его к определённой технологии вызова команд, упростят рефакторинг и сопровождение ПО, а также провести экспериментальные исследования полученных результатов.

Модель предоставления доступа к команде через REST-запрос освободит разработчиков от необходимости составлять шаблоны запросов, которые теперь

будут формироваться автоматически, основываясь на уникальных характеристиках команды. Такой подход значительно повысит расширяемость программного кода и ускорит разработку API.

Модель трансляции ошибки в формат запроса поможет избежать нарушения стандартов при отправке клиенту ошибочного результата. Основой алгоритма является работа со словарями кодов ошибок, с помощью которых будет осуществляться перевод ошибки выполнения команды в ошибку, предусмотренную стандартом используемого формата.

Модель распознавания используемого формата вызова команд определяет отдельные модули, которые отвечают только за один конкретный формат. Данная модель делает возможным добавление дополнительных форматов в дальнейшем и изолирует работу с каждым форматом в отдельном модуле, что снижает количество потенциально возможных ошибок.

Алгоритм определения используемого HTTP-метода является частью реализации модели предоставления доступа к команде через REST-запрос. Исходя из того, что разработчикам не требуется составлять шаблоны запросов, нет необходимости упоминать где-либо в программном коде явные значения HTTP-методов. Данный алгоритм определяет необходимый HTTP-метод по названию вызываемой функции или по указанной операции, которая выбирается из предложенных

Алгоритм чтения аргументов функции из REST-запроса автоматизирует извлечение значений переданных аргументов функции из HTTP-запроса. Данный алгоритм учитывает тот факт, что иногда (в частности при операции чтения) один из аргументов, отвечающий за идентификацию, может быть передан не в теле запроса, а в качестве дополнительного фрагмента URL-адреса. Поэтому алгоритм предусматривает особую обработку аргумента с именем id.

Разработанные модели и алгоритмы позволили увеличить скорость разработки API. Аналогичный по функциональности программный код при использовании разработанного ПС занимает 49 строк против 110 при использовании Express или Restify. Что быстрее более чем в 2 раза.

Эксперимент показал, что разработанное ПС позволяет вести разработку API не привязываясь к конкретной технологии удалённого вызова команд. Экспериментально это выразилось в очень малом количестве программного кода, необходимого для реализации конкретной технологии RPC. Используя разработанное ПС для этой задачи требуется всего 2 строки кода для каждой технологии. При использовании Express или Restify для реализации технологий REST и JSON-RPC потребовалось 14 и 49 строк кода соответственно.

Малое количество строк кода для реализации каждой технологии RPC и модульная система разработанного ПС снижают возможность появления ошибок в реализации технологий RPC и обеспечивают соблюдение стандартов используемых технологий.

Разработанное программное средство представляет собой лишь связующее звено между уже имеющимися программными модулями и веб-сервером, в котором реализована возможность удалённого вызова команд. Таким образом программисту не нужно разрабатывать дополнительную логику работы с сервером и технологиями RPC, что значительно упрощает масштабируемость разрабатываемого API и облегчает его рефакторинг.

Рекомендации по практическому использованию результатов

Полученные результаты формируют теоретическую и практическую базу для разработки ПС для организации удалённого вызова команд. В дополнение к этому, разработанные модели и алгоритмы удалённого вызова команд могут быть использованы для модернизации и дальнейшего развития существующих ПС.

Разработанные модели и алгоритмы удалённого вызова команд могут применяться для создания ПС для организации API, обладающих такими характеристиками, как высокая масштабируемость и надёжность. Программное средство, спроектированное на основе разработанных моделей и алгоритмов, позволит повысить скорость разработки клиент-серверных приложений.

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ

1. Скакун, А. С. Модель единого центра обработки удалённых команд / А. С. Скакун, В. В. Бахтизин // Дистанционное обучение – образовательная среда XXI века: материалы IX международной научно-методической конференции (Минск, 3-4 декабря 2015 года). – Минск: БГУИР, 2015. – С. 288.
2. Скакун, А. С. Единый центр обработки удалённых команд / А. С. Скакун, В. В. Бахтизин // 51-я научная конференция аспирантов, магистрантов и студентов по направлению 4: Компьютерные системы и сети: материалы конференции (Минск, 13-17 апреля 2015 года). – Минск: БГУИР, 2015. – С. 91.
3. Скакун, А. С. Организация единого центра обработки удалённых команд / А. С. Скакун, В. В. Бахтизин // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: Материалы XVIII Республиканской научной конференции студентов и аспирантов (Минск, 21-25 марта 2015 года). – Гомель: ГГУ им. Ф. Скорины, 2015. – С. 190.

Библиотека БГУИР