



УДК 658.512.011.56:004.42

МЕТОД ИМПОРТА И ОБРАБОТКИ ОТКРЫТЫХ ДАННЫХ В LBS-ПЛАТФОРМЕ

М.М. Заславский^a, С.И. Баландин^b^a Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация^b Технологический университет Тампере, Тампере, FI-33101, Финляндия

Адрес для переписки: mark.zaslavskiy@gmail.com

Информация о статье

Поступила в редакцию 27.06.16, принята к печати 24.07.16

doi: 10.17586/2226-1494-2016-16-5-816-822

Язык статьи – русский

Ссылка для цитирования: Заславский М.М., Баландин С.И. Метод импорта и обработки открытых данных в LBS-платформе // Научно-технический вестник информационных технологий, механики и оптики. 2016. Т. 16. № 5. С. 816–822. doi: 10.17586/2226-1494-2016-16-5-816-822

Аннотация

Предложено решение проблемы интеграции геоконтекстной разметки открытых данных в платформе, использующей данные о местоположении (LBS-платформе) путем предоставления разработчикам интерфейсов для запуска повторно используемых модульных инструментов импорта. Реализация предложенного метода выполнена в виде расширения к популярной LBS-платформе Geo2Tag. Метод позволяет удовлетворить требования разработчиков к подсистеме импорта: производительность импорта на уровне существующих запросов платформы, однотипное взаимодействие с импортированными объектами. Исходные коды или исполняемые файлы инструментов импорта могут быть размещены в открытом доступе для других пользователей LBS-платформы, что позволит обеспечить возможность повторного использования и создания библиотек импортеров. В качестве практического примера предложен плагин для импорта открытых данных из музейной информационной системы «Открытая Карелия», использующей Geo2Tag в качестве инструмента обработки геоданных. Проведенные нагрузочные тесты показали, что предложенный метод обеспечивает высокую производительность импорта при количестве параллельных задач импорта, меньшем или равном количеству ядер процессора устройства. Дальнейшее увеличение количества параллельных потоков приводит к значительному снижению производительности, которая, однако, остается достаточно предсказуемой. Разработанный метод применим для широкого класса LBS-платформ, так как он не содержит привязки к особенностям конкретных систем. Метод позволяет преодолеть проблему «холодного старта» сервисов и обеспечивает LBS-платформе и сервисам на ее основе значимое конкурентное преимущество.

Ключевые слова

location-based services, Geo2Tag, LBS-платформа, открытые данные

Благодарности

Работа выполнена при государственной финансовой поддержке ведущих университетов Российской Федерации (субсидия 074-U01).

METHOD OF OPEN DATA IMPORT AND PROCESSING IN LBS-PLATFORM

M.M. Zaslavskiy^a, S.I. Balandin^b^a ITMO University, Saint Petersburg, 197101, Russian Federation^b Tampere University of Technology, Tampere, FI-33101, Finland

Corresponding author: mark.zaslavskiy@gmail.com

Article info

Received 27.06.16, accepted 24.07.16

doi: 10.17586/2226-1494-2016-16-5-816-822

Article in Russian

For citation: Zaslavskiy M.M., Balandin S.I. Method of open data import and processing in lbs-platform. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2016, vol. 16, no. 5, pp. 816–822. doi: 10.17586/2226-1494-2016-16-5-816-822

Abstract

The paper proposes solution for the problem of Open Data integration and geocontext markup in LBS-platform. Solution method is based on providing unified platform-level interfaces for executing modular and reusable Open Data import instruments. The method was implemented as an extension for the popular Geo2Tag LBS-platform. The solution gives the possibility to satisfy developer requirements for import subsystem: import performance is on the same level as a typical client requests to platform and import objects are accessible using platform data manipulation interfaces. Sources or executable files of import instruments can be published in an open access for the other LBS-platform users that enables to reuse of these

instruments and combine them into libraries. Plugin for Open Data import from "Open Karelia" (museum overlay of Geo2Tag) was implemented as a practical application of proposed method. The paper also contains description of load testing of the method implementation which demonstrated that import plugin provides high performance level in case of the parallel tasks number less or equal to the number of processor cores. Further increase of parallel tasks leads to significant decrease of performance, which still keeps predictable enough. Proposed method is applicable for the wide range of LBS-platforms because it has no reference to any specific LBS-platform features. Solution makes it possible to reach significant competitive advantage for LBS platform and its services because the method overcomes «cold start» problem for LBS. It is achieved by the use of Open Data as an initial content for LBS.

Keywords

Location-Based Services, Geo2Tag, LBS-platforms, Open Data

Acknowledgements

The work is supported by the Government of the Russian Federation (grant 074-U01).

Введение

На сегодняшний день наблюдается переход разработки приложений, использующих данные о местоположении (location-based services), на качественно новый уровень. Большое количество разнородных инструментов интегрируется в LBS-платформы – информационные системы, берущие на себя задачи хранения, обработки и представления геоданных. По данным аналитического прогноза «Global LBS Platform Market 2015-2019 – Increased Demand for Location-based Services»,¹ в ближайшие три года ожидается рост потребности в подобных системах среди разработчиков на 22%.

Вторым фактором, меняющим облик разработки LBS, является быстрый рост объема открытых данных² [1]. На сегодняшний день в качестве альтернативы платным геоданным, необходимым пользователям геосервисов, разработчики могут использовать открытые наборы геоданных³. Важно отметить, что зачастую такие наборы данных составляются добровольцами [2] и обладают качеством, сопоставимым с платными аналогами.

Другим качественным изменением в разработке LBS является активное внедрение технологий контекстных вычислений [3]. Согласно прогнозу на 2019 год,⁴ этот фактор приведет к росту рынка LBS более чем в два раза за счет интеграции технологии контекстной разметки с существующими технологиями LBS-платформ [4].

Эффективное сочетание всех трех вышеперечисленных факторов в рамках одной LBS-платформы позволило бы максимизировать удобство для разработчиков. Однако подобная интеграция осложняется следующими причинами:

- форматы хранения и представления открытых данных в различных наборах открытых данных не единообразны и могут быть слабоструктурированы;
- форматы хранения и представления данных в LBS-платформах ориентированы на высокопроизводительную обработку геоданных [5].

Существующие методы организации геоконтекстной разметки для произвольных открытых данных в рамках LBS-платформ можно разделить на две группы: импорт данных силами администраторов LBS-платформы и самостоятельное подключение внешних наборов данных разработчиками сервисов.

В первую группу входят методы, в которых импорт и конвертация данных осуществляется силами администраторов LBS-платформы. Они имеют существенный недостаток, так как разработчики зависят от администраторов LBS-платформы. Данная зависимость проявляется, например, в том, что решение о сроках, проведении и формате импорта принимает администратор. Достоинством данного подхода является высокая скорость импорта. Регулярный импорт открытых данных силами администраторов платформы реализован, например, в системе 2ГИС⁵.

Импорт данных силами разработчиков сервисов позволяет им самостоятельно выполнять интеграцию любых наборов данных, но при этом разработчики вынуждены использовать внешние вычислительные ресурсы и иметь инфраструктуру сопровождения, так как изменения, вносимые в LBS-платформу, могут нарушить совместимость с разработанными решениями. Подобный подход широко применяется в LBS-интерфейсах популярных картографических провайдеров Google Maps и Яндекс Карты^{6,7}.

¹ Global LBS Platform Market 2015-2019 - Increased Demand for Location-based Services // Business Wario. URL: <http://www.businesswire.com/news/home/20150716005694/en/Increased-Demand-Location-Based-Services-Improve-Global-LBS> (дата обращения 26.06.2016)

² Market value Open Data to reach 286 billion by 2020 // EUROPEAN DATA PORTAL. URL: <http://www.consultancy.uk/news/3019/market-value-open-data-to-reach-286-billion-by2020> (дата обращения 26.06.2016)

³ LinkedGeoData. URL: <http://linkedgeo.org/> (дата обращения 26.06.2016)

⁴ Location based services market to reach \$43.3bn by 2019, driven by context aware mobile services // Juniper Research. URL: <http://www.juniperresearch.com/press-release/context-and-location-based-services-pr2> (дата обращения 26.06.2016)

⁵ 2ГИС. URL: <http://info.2gis.ru/> (дата обращения 26.06.2016).

⁶ Google Maps. URL: <https://www.google.ru/maps> (дата обращения 26.06.2016).

⁷ Яндекс Карты. URL: <https://tech.yandex.ru/maps> (дата обращения 26.06.2016).

В данной работе предлагается решение проблемы интеграции геоконтекстной разметки открытых данных в LBS-платформе путем предоставления разработчикам интерфейсов для запуска повторно используемых модульных инструментов импорта. Данный метод позволяет избежать недостатков обеих существующих схем импорта открытых данных в LBS-платформы. За счет использования интерфейсов платформы скорость импорта будет сопоставима со скоростью для первого подхода, при этом передача ответственности за проведение и параметры импорта разработчикам сервисов позволит повысить удобство использования LBS-платформы, что является ее существенным конкурентным преимуществом. Помимо этого, использование метода позволит новым сервисам решить проблему «холодного старта» [6] – отсутствие начальных данных для реализации основного сценария использования сервиса.

Постановка задачи

Приведенный в предыдущем разделе краткий обзор существующих методов организации работы с открытыми данными в рамках LBS-платформ продемонстрировал, что найденные решения не являются универсальными для различных типов открытых данных и зачастую требуют существенных временных затрат на реализацию инструментов импорта или организации процедуры импорта данных. Исходя из этого, целью настоящей работы было создание нового метода, который предоставит пользователям возможность осуществлять работу с открытыми данными (в том числе и геоконтекстную разметку [7]) напрямую через методы LBS-платформы. Для преодоления недостатков существующих решений метод должен обладать следующими свойствами:

- возможностью повторного использования программных решений на базе подхода для различных наборов открытых данных;
- производительность импорта и работы с открытыми данными должны соответствовать уровню работы с обычными геоданными LBS-платформы;
- работа с открытыми данными должна осуществляться только через интерфейсы LBS-платформы.

Предлагаемый способ решения

Решением задачи организации высокопроизводительной работы с произвольными открытыми данными является предоставление интерфейсов на уровне LBS-платформы, позволяющих создавать и запускать пользовательские инструменты импорта данных. Данный метод позволяет удовлетворить все идентифицированные требования разработчиков к подобной системе:

1. запуск пользовательских инструментов в рамках платформы позволяет использовать вычислительные ресурсы LBS-платформы, что обеспечивает производительность импорта на уровне существующих запросов платформы;
2. интеграция инструментов импорта с LBS-платформой позволяет обеспечить однотипное взаимодействие с импортированными объектами, так как в процессе импорта можно вести запись объектов напрямую в базу данных (БД) платформы;
3. исходные коды или исполняемые файлы инструментов импорта могут быть размещены в открытом доступе для других пользователей LBS-платформы, что позволит обеспечить возможность повторного использования и создания библиотек импортеров.

К недостаткам предлагаемого метода можно отнести возникающие риски, связанные с информационной безопасностью, так как пользовательские дополнения представляют собой угрозу для экземпляров LBS-платформы [8], в случае если их выполнение происходит на одном и том же вычислительном узле. Также необходимо реализовать процедуры ограничения вычислительных ресурсов и приоритизации работы пользовательских инструментов импорта, так как в противном случае нельзя гарантировать обеспечение требований доступности и производительности основных сценариев использования экземпляра LBS-платформы.

С практической точки зрения данный метод можно реализовать двумя способами: в виде механизма подключаемых пользовательских дополнений или в виде прямых изменений разработчиками сервисов исходного кода LBS-платформы. Наиболее целесообразным является первый способ, так как для активно используемых LBS-платформ изменение исходного кода напрямую разработчиками сервисов зачастую невозможно в силу лицензионных ограничений и требований к доступности запущенных экземпляров платформы.

LBS-платформа Geo2Tag

LBS-платформа Geo2Tag [7] представляет собой программное обеспечение с открытым исходным кодом, предназначенное для создания пользовательских геосервисов по модели «Платформа как сервис» (Platform as a Service) [9]. Основными задачами платформы являются администрирование геосервисов, хранение и анализ геоданных.

С технической точки зрения Geo2Tag представляет собой WSGI-приложение [10] на базе технологий с открытым исходным кодом. В его основе лежит Python-фреймворк Flask [11], использующий пере-

ляционную БД MongoDB¹ и веб-сервер Apache². Все взаимодействия с платформой, включая администрирование и создание новых геосервисов, осуществляются при помощи интерфейсов прикладного программирования, отвечающих стандарту «Representational State Transfer» (REST)³. Процесс авторизации пользователей реализован на основе протокола OAuth [12] для фреймворка Flask, поддерживаются провайдеры Google и Facebook.

Geo2Tag является апробированным инструментом решения практических задач. Например, в России данная платформа использовалась для создания геосервисов на музейных информационных системах «Открытая Карелия» и «Виртуальная Сортавала» [13].

Поскольку исходный код и все базовые технологии LBS-платформы Geo2Tag относятся к программному обеспечению с открытым исходным кодом, то авторы настоящей работы выбрали ее для апробации разработанного метода.

Разработанное решение

В качестве реализации метода использования открытых данных в LBS-платформе Geo2Tag авторами был создан API для импорта открытых данных на базе существующей подсистемы пользовательских плагинов. Данные интерфейсы включают в себя общий алгоритм импорта, абстрактные классы для различных этапов импорта и REST-интерфейсы для управления процедурой импорта. Реализация вышеописанного метода составила 1113 строк кода и доступна для свободного скачивания в составе исходного кода платформы Geo2Tag⁴.

Алгоритм импорта открытых данных в рамках добавленного API использует следующие входные данные: идентификатор канала (именованного множества данных), ссылка на набор открытых данных и название сервиса Geo2Tag, в БД которого необходимо произвести импорт. Алгоритм состоит из четырех стадий.

- Получение данных. Этап включает в себя все действия по установлению соединения, авторизации и загрузки данных от источника открытых данных.
- Разделение массива открытых данных на отдельные элементы. Происходит начальная обработка массива путем его десериализации.
- Трансляция отдельных элементов открытых данных в формат Geo2Tag. Происходит извлечение пространственно-временных атрибутов из исходного элемента открытых данных и их запись в аналогичные атрибуты нового элемента геоданных Geo2Tag (точки). Помимо этого, в новой точке также фиксируются метаданные о текущем сеансе импорта – ссылка на набор открытых данных и время импорта.
- Запись точек в БД сервиса.

Абстрактные классы API-импорта открытых данных включают в себя логику выполнения каждого этапа алгоритма импорта и шаблоны для реализации REST-интерфейсов управления процедурой импорта. Подробности реализации изложены в работе [7].

Поскольку предложенный в работе метод включает в себя геоконтекстную разметку открытых данных, то для его успешной реализации в рамках LBS-платформы Geo2Tag необходимо предоставить инструменты для этой операции. Так как информация о местоположении может храниться в элементе открытых данных не только в цифровом виде, но и в форме почтового адреса либо в форме географического словесного описания, то возникает проблема получения географических координат по текстовому описанию. Для ее решения был разработан плагин, решающий задачу прямого геокодирования. Плагин расширяет возможности LBS-платформы Geo2Tag REST-интерфейсами:

- /instance/plugin/geocoding/service/<string:serviceName>/job – создание задачи на пакетное геокодирование данных в определенном канале сервиса serviceName;
- /instance/plugin/geocoding/service/<string:serviceName>/job/<string:jobId> – управление созданной задачей.

В качестве практического примера использования разработанного API был разработан плагин для импорта открытых данных из музейной информационной системы «Открытая Карелия», использующей Geo2Tag в качестве инструмента обработки геоданных. Такой источник данных был выбран, так как его объекты содержат в себе информацию о местоположении в виде географических координат и информацию о времени в интервальном формате с поддержкой дат до нашей эры, что, в свою очередь, позволяет апробировать разработанный подход к организации работы с открытыми данными в рамках LBS-платформы на данных сложной структуры [13]. Для получения данных использовался REST-интерфейс

⁸ MongoDB for GIANT Ideas // MongoDB. URL: <https://www.mongodb.com/> (дата обращения 26.06.2016)

⁹ The Apache HTTP Server Project. URL: <https://httpd.apache.org/> (дата обращения 26.06.2016)

¹⁰ Representational State Transfer (REST) // URL: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm (дата обращения 26.06.2016)

¹¹ Geo2tag is an Open Source LBS Platform. URL: <http://geo2tag.org> (дата обращения 26.06.2016)

nearest_objects (ближайшие объекты) «Открытой Карелии». Данный интерфейс позволяет получить список объектов системы, обладающих географическими координатами и находящихся не дальше одного километра от заданной точки.

Для исследования принципиальной применимости и производительности разработанного метода был поставлен эксперимент по измерению скорости одновременного импорта данных в LBS-платформу на примере разработанного плагина импорта. В качестве моделируемой ситуации рассматривался одновременный импорт открытых данных для небольшого количества различных сервисов. Для наглядности в качестве испытательного стенда использовался низкопроизводительный бытовой компьютер со следующими характеристиками:

- двухъядерный процессор Intel(R) Core(TM) i3-2310M CPU @ 2.10GHz;
- оперативное запоминающее устройство 3 ГБ.

В рамках эксперимента была проведена серия однотипных операций импорта при количестве одновременных задач импорта в интервале 1–10. Такой интервал был выбран для проверки как случаев, когда существующие возможности для параллельной обработки задач импорта превышают нагрузку, так и для обратных случаев, когда количество параллельно исполняемых задач импорта на порядок превышает количество ядер процессора.

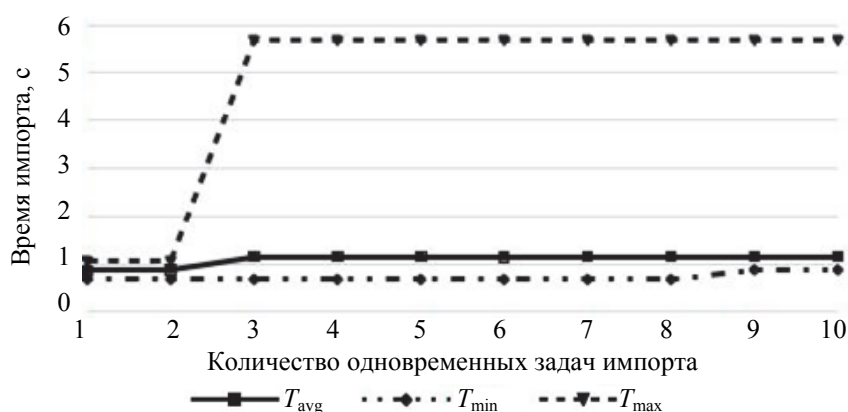


Рис. 1. Кривая зависимости времени импорта от количества одновременно выполняющихся задач импорта

Результаты эксперимента были проанализированы и представлены в виде зависимостей времени выполнения задачи импорта и количества импортируемых объектов в секунду от числа одновременных задач импорта.

Полученные в ходе эксперимента зависимости минимального, среднего и максимального времени импорта от количества одновременных задач (рис. 1) показывают, что, несмотря на ограничение максимального числа параллельных потоков выполнения и на используемую в языке Python технологию Global Interpreter Lock (GIL) [14], среднее время импорта растет медленно. Необходимо также заметить, что при превышении количеством одновременных задач импорта числа ядер процессора возникает резкий рост максимального времени импорта. Данное явление вызвано тем, что в проведенном эксперименте не использовались дополнительные инструменты диспетчеризации задач, кроме встроенного механизма управления соединениями в веб-сервере Apache и GIL, что приводит к неэффективной работе в случае, когда на одно ядро приходится более одной задачи импорта. Однако при этом величина максимального времени импорта остается практически неизменной с ростом числа одновременных задач импорта в диапазоне 3–10, что можно объяснить работой кэширования хранилища MongoDB [15]. Таким образом, динамика времени работы задачи импорта при параллельном исполнении показывает принципиальную возможность использования разработанного метода в LBS-платформах и устойчивость экспериментальной системы к повышению нагрузки.

График зависимости скорости импорта от числа одновременно выполняющихся операций (рис. 2) показывает, что при одновременной обработке более двух задач импорта происходит резкое падение количества одновременных задач импорта, которое компенсируется эффективной работой БД, как было показано выше. Однако этот эффект не может компенсировать снижение производительности при дальнейшем росте числа одновременных задач импорта, так как при отсутствии механизма диспетчеризации при определенном уровне нагрузки задачи импорта начнут исчерпывать вычислительные ресурсы системы, необходимые для успешной работы других компонентов платформы. В связи с этим важным направлением для дальнейшего развития метода организации работы и импорта открытых данных в рамках LBS-платформ является разработка методов планирования выполнения задач импорта и квотирования ресурсов для них.

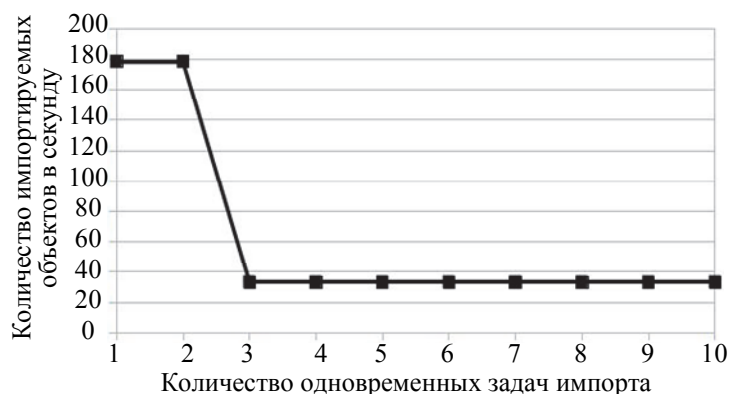


Рис. 2. Зависимость количества импортируемых в секунду объектов от количества одновременных задач импорта

Заключение

Был разработан метод организации импорта и геоконтекстной разметки открытых данных для хранилища LBS-платформ и подход к его программной реализации. Реализация предложенного метода была выполнена в виде расширения к популярной LBS-платформе Geo2Tag. Соответствующий код добавлен к основному коду платформы Geo2Tag. Работоспособность метода апробирована на музейных сервисах, а полученные количественные и качественные замеры показали эффективность предложенного решения.

Разработанный метод для работы с открытыми данными в рамках LBS-платформ позволяет добиться быстрого подключения и геоконтекстной разметки произвольных наборов данных, содержащих информацию о местоположении. При этом основные усилия для подключения и импорта новых источников данных необходимо прикладывать самим пользователям LBS-платформ, что позволяет пополнять базу данных LBS-платформы интересными для пользователей данными без затрат со стороны администраторов и разработчиков платформы.

Практическое внедрение разработанного метода в LBS-платформу Geo2Tag показало, что он позволяет решать проблему «холодного старта» новых сервисов, при этом обеспечивая высокую производительность при импорте начального набора данных.

Проведенные нагрузочные тесты показали, что предложенный метод обеспечивает высокую производительность импорта при количестве параллельных задач импорта, меньшем или равном количеству ядер процессора устройства. Дальнейшее увеличение количества параллельных потоков приводит к значительному снижению производительности, но тем не менее сохраняет ее в хорошо предсказуемом «коридоре производительности». Повышение эффективности для большего количества параллельных потоков является интересной практической задачей организации эффективной диспетчеризации и плинурется как следующий этап работы. Кроме того, в дальнейшем планируется реализация пользовательских интерфейсов для работы с плагинами импорта открытых данных при реализации неполной загрузки импортируемых данных с целью снижения требований к объему хранилища, а также добавление новых источников открытых данных.

Литература

1. Kitchin R. *The Data Revolution: Big Data, Open Data, Data Infrastructures and their Consequences*. Sage, 2014. 285 p.
2. Karam R., Melchiori M. A crowdsourcing-based framework for improving geo-spatial open data // *Proc. 2013 IEEE Int. Conf. on Systems, Man, and Cybernetics*. Manchester, UK, 2013. P. 468–473. doi: 10.1109/SMC.2013.85
3. Dey A.K. Understanding and using context // *Personal and Ubiquitous Computing*, 2001. V. 5. N 1. P. 4–7. doi: 10.1007/s007790170019
4. Jiang B., Yao X. Location-based services and GIS in perspective // *Computers, Environment and Urban Systems*. 2006. V. 30. N 6. P. 712–725. doi: 10.1016/j.compenvurbsys.2006.02.003
5. Zhao J., Zheng C.M., Zhou D. Design and implementation of a location-based service platform // *Proc. 10th Int. Conf. on Advanced Communication Technology*. Phoenix Park, 2008. V. 1. P. 529–533. doi: 10.1109/ICACT.2008.4493818
6. Lam X.N., Vu T., Le T.D., Duong A.D. Addressing cold-start problem in recommendation systems // *Proc. 2nd Int. Conf. on Ubiquitous Information Management and Communication*. Suwon, 2008. P. 208–211. doi: 10.1145/1352793.1352837
7. Zaslavskiy M., Mourontsev D. Implementation of the new

References

1. Kitchin R. *The Data Revolution: Big Data, Open Data, Data Infrastructures and their Consequences*. Sage, 2014, 285 p.
2. Karam R., Melchiori M. A crowdsourcing-based framework for improving geo-spatial open data. *Proc. 2013 IEEE Int. Conf. on Systems, Man, and Cybernetics*. Manchester, UK, 2013, pp. 468–473. doi: 10.1109/SMC.2013.85
3. Dey A.K. Understanding and using context. *Personal and Ubiquitous Computing*, 2001, vol. 5, no. 1, pp. 4–7. doi: 10.1007/s007790170019
4. Jiang B., Yao X. Location-based services and GIS in perspective. *Computers, Environment and Urban Systems*, 2006, vol. 30, no. 6, pp. 712–725. doi: 10.1016/j.compenvurbsys.2006.02.003
5. Zhao J., Zheng C.M., Zhou D. Design and implementation of a location-based service platform. *Proc. 10th Int. Conf. on Advanced Communication Technology*. Phoenix Park, 2008, vol. 1, pp. 529–533. doi: 10.1109/ICACT.2008.4493818
6. Lam X.N., Vu T., Le T.D., Duong A.D. Addressing cold-start problem in recommendation systems. *Proc. 2nd Int. Conf. on Ubiquitous Information Management and Communication*. Suwon, 2008, pp. 208–211. doi: 10.1145/1352793.1352837

- REST API for open source LBS-platform Geo2Tag // *Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference*. St. Petersburg, Russia, 2015. P. 125–130. doi: 10.1109/AINL-ISMW-FRUCT.2015.7382981
8. Park N., Kim H., Kim S., Won D. Open location-based service using secure middleware infrastructure in web services // *Lecture Notes in Computer Science*. 2005. V. 3481. N II. P. 1146–1155.
 9. Boniface M., Nasser B., Papay J., Phillips S.C., Servin A., Yang X. et. al. Platform-as-a-service architecture for real-time quality of service management in clouds // *Proc. 5th Int. Conf. on Internet and Web Applications and Services*. Barcelona, Spain, 2010. P. 155–160. doi: 10.1109/ICIW.2010.91
 10. Gardner J. The Web Server Gateway Interface (WSGI) / In: *The Definitive Guide to Pylons*. Apress, 2009. 513 p.
 11. Grinberg M. *Flask Web Development: Developing Web Applications with Python*. O'Reilly Media, 2014. 258 p.
 12. Leiba B. OAuth web authorization protocol // *IEEE Internet Computing*. 2012. V. 16. N 1. P. 74–77. doi: 10.1109/MIC.2012.11
 13. Balandina E., Balandin S., Koucheryavy Y., Mouromtsev D. Innovative e-tourism services on top of Geo2Tag LBS platform // *Proc. 11th Int. Conf. on Signal-Image Technology and Internet Systems*. Bangkok, Thailand, 2015. P. 752–759. doi: 10.1109/SITIS.2015.11
 14. Beazley D. *Understanding the python GIL* // *PyCON Python Conference*. Atlanta, Georgia, 2010.
 15. Banker K. *MongoDB in Action*. Manning Publications, 2011. 287 p.
 7. Zaslavskiy M., Mouromtsev D. Implementation of the new REST API for open source LBS-platform Geo2Tag. *Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference*. St. Petersburg, Russia, 2015, pp. 125–130. doi: 10.1109/AINL-ISMW-FRUCT.2015.7382981
 8. Park N., Kim H., Kim S., Won D. Open location-based service using secure middleware infrastructure in web services. *Lecture Notes in Computer Science*, 2005, vol. 3481, no. II, pp. 1146–1155.
 9. Boniface M., Nasser B., Papay J., Phillips S.C., Servin A., Yang X. et. al. Platform-as-a-service architecture for real-time quality of service management in clouds. *Proc. 5th Int. Conf. on Internet and Web Applications and Services*. Barcelona, Spain, 2010, pp. 155–160. doi: 10.1109/ICIW.2010.91
 10. Gardner J. The Web Server Gateway Interface (WSGI). In *The Definitive Guide to Pylons*. Apress, 2009, 513 p.
 11. Grinberg M. *Flask Web Development: Developing Web Applications with Python*. O'Reilly Media, 2014, 258 p.
 12. Leiba B. OAuth web authorization protocol. *IEEE Internet Computing*, 2012, vol. 16, no. 1, pp. 74–77. doi: 10.1109/MIC.2012.11
 13. Balandina E., Balandin S., Koucheryavy Y., Mouromtsev D. Innovative e-tourism services on top of Geo2Tag LBS platform. *Proc. 11th Int. Conf. on Signal-Image Technology and Internet Systems*. Bangkok, Thailand, 2015, pp. 752–759. doi: 10.1109/SITIS.2015.11
 14. Beazley D. Understanding the python GIL. *PyCON Python Conference*. Atlanta, Georgia, 2010.
 15. Banker K. *MongoDB in Action*. Manning Publications, 2011, 287 p.

Авторы

Заславский Марк Маркович – аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, mark.zaslavskiy@gmail.com

Баландин Сергей Игоревич – кандидат технических наук, адъюнкт-профессор, адъюнкт-профессор, Технологический университет Тампере, Тампере, FI-33101, Финляндия, Sergey.Balandin@fruct.org

Authors

Mark M. Zaslavskiy – postgraduate, ITMO University, Saint Petersburg, 197101, Russian Federation, mark.zaslavskiy@gmail.com

Sergey I. Balandin – PhD, Adjunct Professor, Tampere University of Technology, Tampere, FI-33101, Finland, Sergey.Balandin@fruct.org