

УДК 004.043

М. А. Навроцкий, Д. И. Муромцев
Университет ИТМО

Н. А. Жукова, А. И. Водяхо
Санкт-Петербургский государственный электротехнический
университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Архитектурные решения для порталов научно-технической информации

Описывается архитектура разработанного класса систем, семантических образовательных порталов. Представлено множество типовых архитектурных решений, которые могут быть использованы в зависимости от требований к portalу и предметной области. Особенности предлагаемых архитектурных решений является использование сервисного подхода при разработке модулей и подсистемы. Такой подход позволяет настраивать систему под требования каждого конкретного подразделения и предметную область. Основным достоинством предлагаемого подхода является возможность использования стандартизованных интерфейсов для взаимодействия сервисов, что позволяет при необходимости заменить любой модуль. Каждый сервис реализован таким образом, что он может быть упакован в отдельный, изолированный контейнер, что позволяет развертывать сервисы в разных облаках или хостах и более экономно использовать имеющиеся ресурсы. Рассматриваются вопросы проектирования и разработки программной платформы для реализации научно-образовательных порталов для различных предметных доменов.

Научно-образовательные порталы, онтология, открытые связанные данные, поиск в открытых данных

В последние 10 лет критически возрос объем публикуемого в Интернете контента, который включает в себя научные и образовательные данные. Источниками таких данных являются, в частности, университеты, издательства, научные сотрудники и исследователи [1]. Они публикуют данные о публикациях, научных проектах, открытых образовательных курсах, научных результатах и т. п. Эти источники данных состоят из *связанных открытых данных* (Linked Open Data, LOD) и non-LOD-источников. Получение описанных данных позволяет решать многие проблемы, такие, как расчет и оценка показателей эффективности работы сотрудников и подразделений, подготовка заявок на гранты, конференции и т. п.

Сотрудники и студенты университета могут использовать имеющиеся источники данных, доступ к которым осуществляется через порталы для решения научных и образовательных (повседневных) задач. Однако в настоящее время этого не происходит по следующим причинам: пользователи могут не знать о существовании части источников; поиск

в множестве источников может занимать много времени и сил; пользователь может не знать, с чего ему начать поиск (отсутствует точка входа).

Доступ к опубликованным данным, предоставляемым различными источниками, может обеспечиваться за счет интеграции этих данных, предоставления доступа к ним через единый портал. С учетом того, что источники могут быть не только простыми каталогами, но и крупными сложными порталами связанных данных, использование интегрированных данных может приводить к информационной избыточности при выполнении запросов пользователей. Решить проблему избыточности можно за счет адаптации полученной информации к потребностям каждого конкретного пользователя. При этом должны учитываться интересы пользователя в контексте решаемой задачи. Это может быть обеспечено за счет использования онтологического подхода к описанию предметной области и алгоритмов адаптации.

Описанные решения позволяют разработать научные порталы открытых данных для отдельных подразделений университета. Портал инте-

грирует данные из источников разных типов (LOD-источники и non-LOD-источники), определяет информационное пространство для каждого пользователя, формирует представление научной и образовательной активности пользователей.

Проведем анализ существующих составов данных университетов в образовании и науке. Linked Universities – это альянс европейских университетов, который занимается распространением своих открытых данных в формате связанных данных и публикует онтологические словари для описания этих данных. Связанные данные – это множество принципов публикации сырых данных в Интернете с добавлением к ним адреса и ссылки таким образом, чтобы их было легко найти, получить к ним доступ и реализовать их повторное использование. Идея состоит в том, что данные от различных университетов и организаций могут вносить вклад в общее пространство данных.

Кроме того, сообщество Linked Universities считает, что потенциал использования связанных данных в сферах образования и науки выходит далеко за пределы личной выгоды для каждого университета. При этом наибольший положительный эффект можно получить только посредством предоставления межуниверситетских данных, которые могут быть агрегированы, интегрированы и сопоставлены. В то же время связанные данные, как и Интернет, опираются на свободное и открытое распространение и определенное количество самоорганизации, при этом обмен опытом и сотрудничество в развитии платформ связанных данных университетов может существенно помочь достижению этой цели, а именно созданию *сети университетских данных*.

Многие университеты разрабатывают собственные LOD-порталы, которые могут использоваться в качестве источников данных для разрабатываемого портала, в частности:

- LODUM – проект Вестфальского университета имени Вильгельма; портал публикует научные и образовательные данные в формате RDF;
- Оксфорд публикует результаты своей научной деятельности;
- портал университета Саутгемптона содержит исследовательские данные;
- Боннский университет открывает доступ к своей научной деятельности;
- Гарвардский университет предоставляет доступ к публикациям.

Linked Universities можно использовать в двух направлениях: как источники открытых данных и использовать словари для описания этих данных.

Общий подход к решению задачи можно определить следующим образом: портал подключается к открытым источникам данных с учетом требований подразделения и домена (настройка системы); при регистрации пользователь указывает свои интересы в виде списка ключевых слов; при запросе пользователя к системе портал формирует множество данных (результатирующую выборку) и фильтрует данные с учетом интересов пользователя; система формирует страницу выдачи, снабжая данные ссылками на источники данных, где пользователь может просмотреть информацию более детально.

Типовые варианты реализации доменных порталов научно-технической информации. Можно выделить 2 типовых подхода к построению портала: иерархия систем по предметной области, иерархия систем по применению в образовании.

При внедрении портала специалист по внедрению и настройке определяет требуемую архитектуру и уровень проработки предметной области портала.

На рис. 1 представлена иерархия систем с учетом проработки предметной области и предназначения студенческого научного портала (СНП).

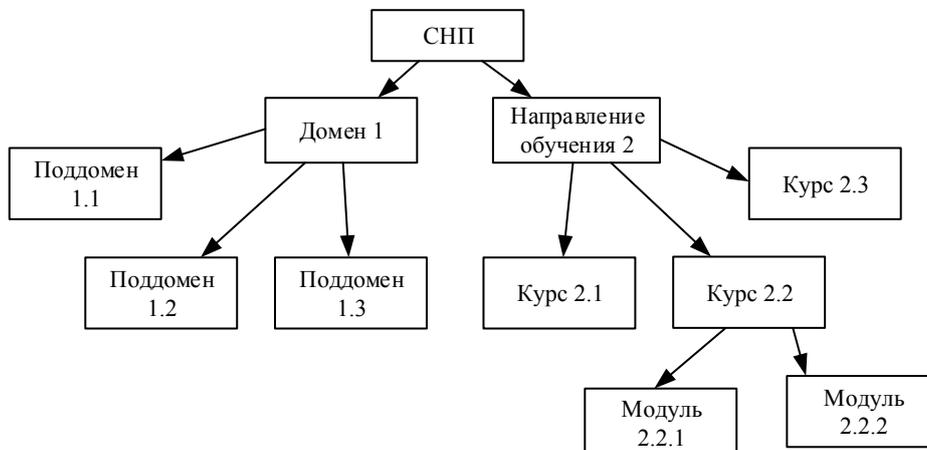


Рис. 1

С учетом специфики требований для каждого уровня образовательного процесса можно детализировать получаемые данные до требуемого уровня. Требования для получаемых данных на уровне образовательного курса и на уровне конечного пользователя будут отличаться. Детализируя предметную область можно получить необходимые данные.

Кроме того, выделяются параметры системы (рис. 2) с учетом требований, предъявляемых к системе.

На верхнем уровне системы параметры можно классифицировать по следующим признакам: по реконфигурированию описанных моделей, по типу архитектуры, по скорости обработки запросов.

С учетом требований выбирается необходимая архитектура, уточняются состав сервисов, требования и настройки. Например, в случае большого количества источников запросы к источникам данных будут выполняться в течение длительного времени. При необходимости может использоваться кэширование, что позволяет сохранять запросы и наборы данных и при повторном запросе получать данные из кеша, что значительно ускорит процесс получения набора данных. В зависимости от типа требуемого кэширования может использоваться либо локальная онтология, либо специализированная база данных.

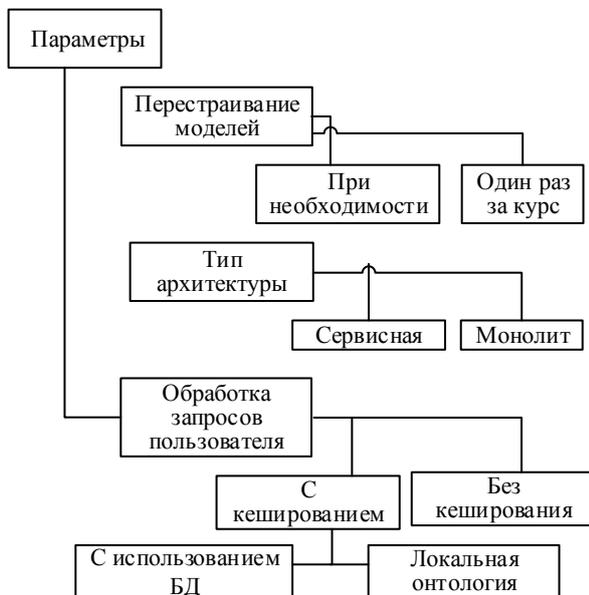


Рис. 2

Проектирование и разработка программной платформы для доменных порталов научно-технической информации. Проектные решения при разработке системы описываются на языке

UML [2]. Для системы реализуется микросервисная архитектура [3]. Сервисы, из которых состоит СНП, и зависимости между ними представлены на рис. 3.

Данная структура включает в себя следующие элементы:

Client – легкий клиент, используемый для взаимодействия пользователей с системой. Это может быть HTML5-приложение, мобильное приложение или внешнее приложение.

Glossary server – сервер словаря, используемый для заполнения словаря предметной области, предоставления значений терминов, добавленных в словарь. Словарь уточняет предметную область.

Portal server – сервер портала, используемый для регистрации пользователей, формирования поисковой выдачи. Главная точка входа в приложение: принимает запросы клиента и вызывает другие сервисы.

Configuration module – модуль конфигурации системы, который используется для формирования списка источников данных, конфигурирует запросы, записывает и сохраняет настройки системы.

Local ontology – локальная онтология, которая хранится в формате OWL, используется для описания пользователей (интересы пользователей), описания поисковой выдачи. Онтология хранит в себе статистику использования системы: список запросов пользователей и ответов источников, что позволяет использовать ее для оценки работы системы.

Adaptation module – модуль адаптации поисковой выдачи к текущему пользователю. Учитывая множество интересов пользователя, модуль уточняет поисковую выдачу перед выводом пользователю.

Translate module – модуль перевода слова из словаря или поискового запроса. Для перевода используется DBpedia.

Sources base – база (список) источников данных, которая распространяется вместе с системой и может пополняться пользователями с последующим распространением.

Department's semantic database – семантическая база подразделения, описывающая семантические данные сотрудников подразделения в RDF. Это позволяет искать данные не только в открытых данных, но и внутри организации: перенести знания в подразделении.

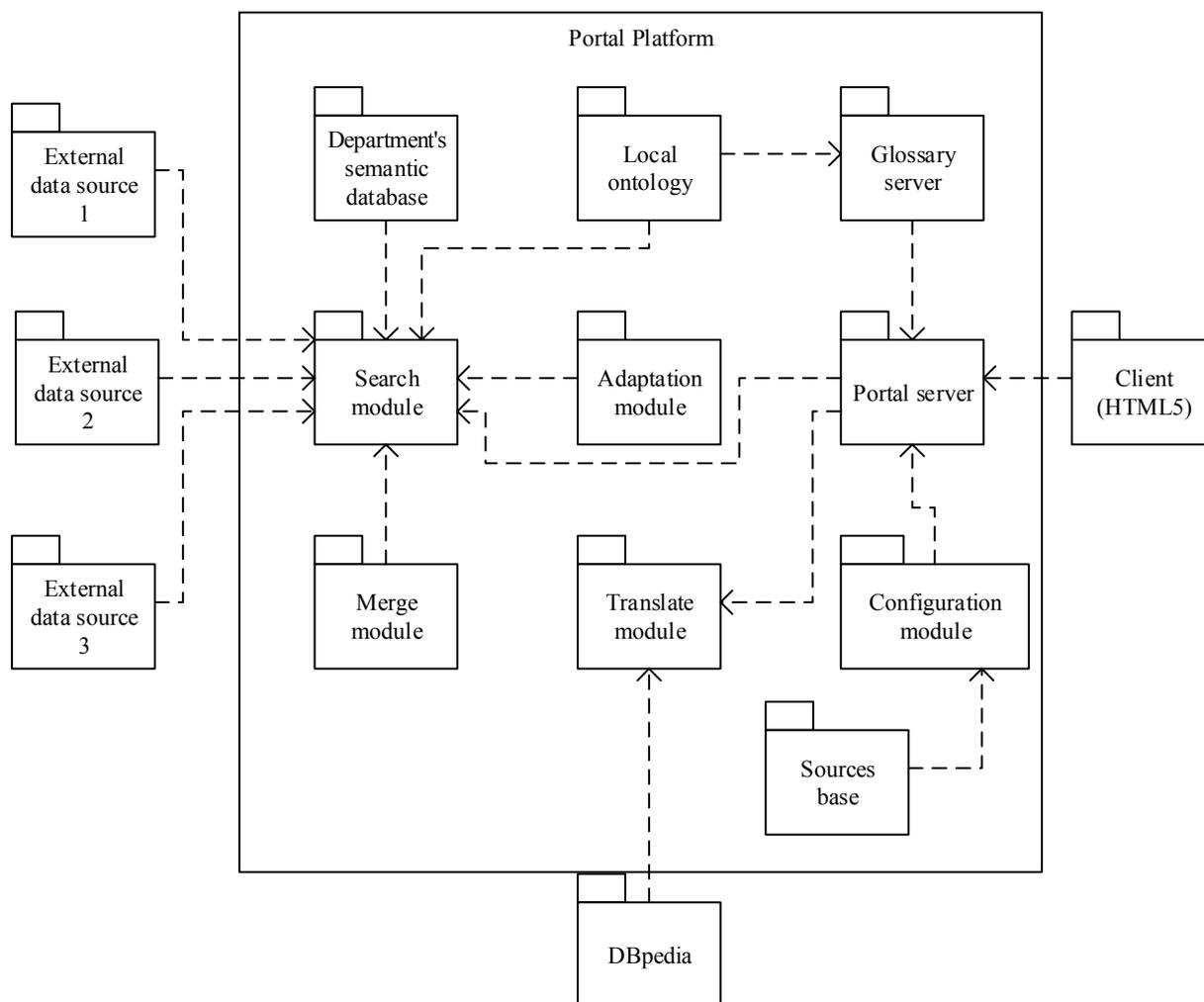


Рис. 3

Search module – модуль поиска данных по запросу пользователя. Модуль последовательно строит запросы к источникам данных и собирает наборы данных из них.

Merge module – модуль слияния всех полученных данных.

DBpedia – портал открытых данных, используемый для перевода слов. Содержит в себе базу Википедии.

External data source – открытый источник данных, используемый для реализации поиска наборов данных по запросу пользователя.

Сервисы СНП и протоколы их взаимодействия представлены на рис. 4.

Все взаимодействие между сервисами реализовано на базе стандартных протоколов HTTP и TCP. Протокол TCP используется для реализации доступа к локальной базе данных, а HTTP – для реализации взаимодействия между микросервисами. Исключением является сервис поиска и

адаптации (Search and adaptation service), взаимодействие которого с локальной онтологией (Local ontology) реализуется с помощью специальной библиотеки и интерфейса [4], [5]. Использование этой библиотеки обусловлено тем, что на данный момент она является наиболее развитым средством взаимодействия между онтологическими моделями и языком программирования.

Внешние источники данных, как и семантический портал подразделения, предоставляют интерфейс для выполнения запросов на базе протокола SPARQL. Аналогично строится и взаимодействие с DBpedia. Остальные сервисы реализуют собственные RESTful-интерфейсы.

Использование таких стандартных протоколов позволяет заменять, модернизировать, убирать сервисы в зависимости от требований предметной области.

Каждый из сервисов СНП работает в рамках своей изолированной среды.

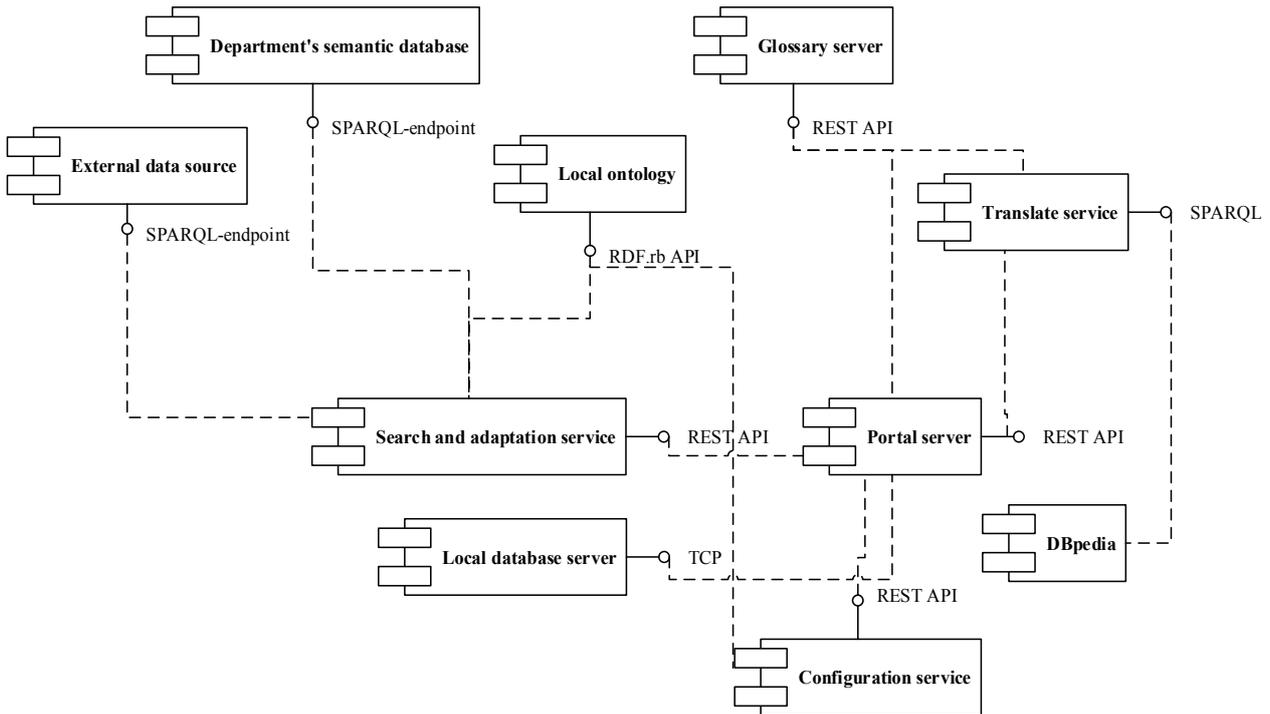


Рис. 4

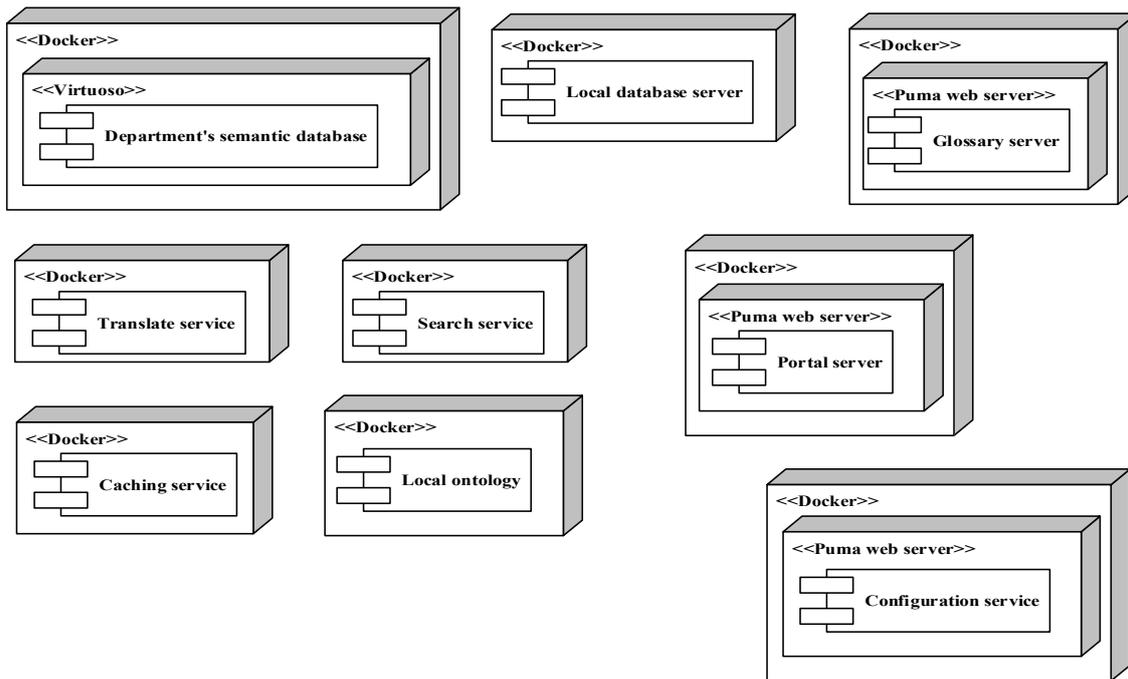


Рис. 5

Сервисы и их среды выполнения представлены на рис. 5.

Семантический научный портал имеет распределенную микросервисную архитектуру, что позволяет каждому из сервисов работать на отдельном сервере. Для управления запуском и развертыванием сервисов было выбрано решение, предоставляемое проектом Docker [6]. Данная система содержит множество программных средств для декларативного создания виртуальных машин.

Каждый сервис СНП работает в рамках отдельного docker-контейнера. Внутри контейнера сервисы используют другие программные фреймворки и среды:

- Virtuoso – окружение, настроенное на использование Virtuoso [7]. Внутри контейнера автоматически запускается программное средство Virtuoso. Virtuoso universal server – это сервер, используемый для запуска баз данных RDF. Реализует точку входа SPARQL для доступа к данным через запросы.

– Puma web server – окружение, настроенное на запуск сетевых приложений с использованием web-сервера Puma. Puma – современный web-сервер, позволяющий запускать несколько экземпляров одновременно, что повышает скорость доступа к данным [8].

Эталонные архитектуры доменных порталов научно-технической информации. Архитектура может варьироваться в зависимости от требований подразделения, предметной области и

условий использования системы. Можно выбрать следующие функции:

- наличие модуля перевода (он необходим, если язык запросов пользователей отличается от языка, на котором хранятся данные в открытых источниках);
- наличие словаря (он нужен для уточнения предметной области);
- наличие семантического портала организации (если организация имеет собственный семан-

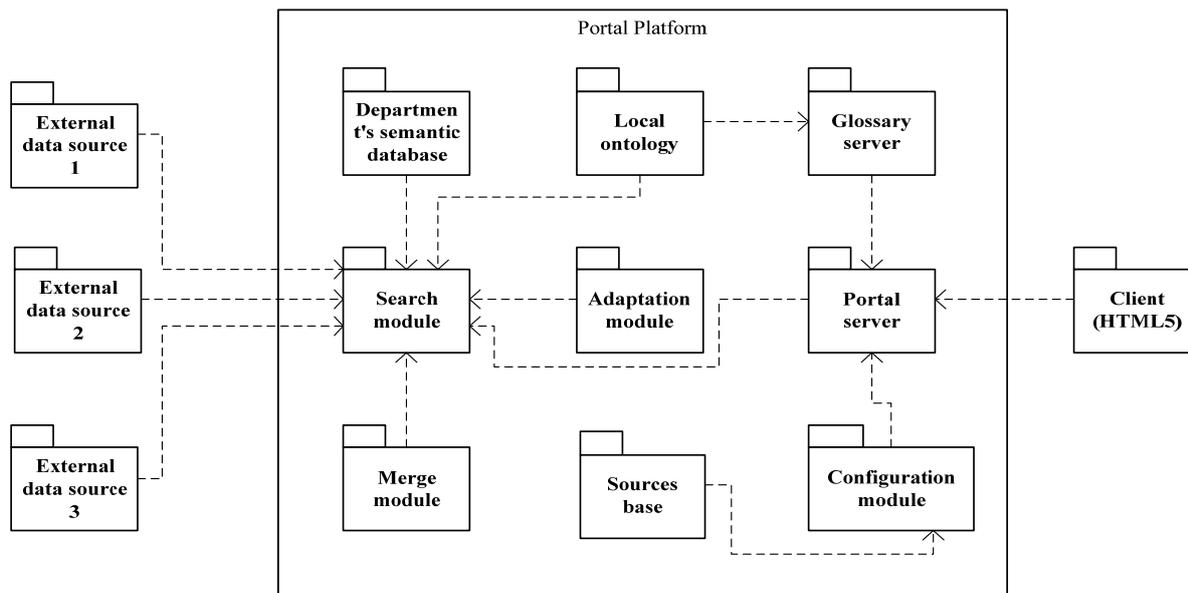


Рис. 6

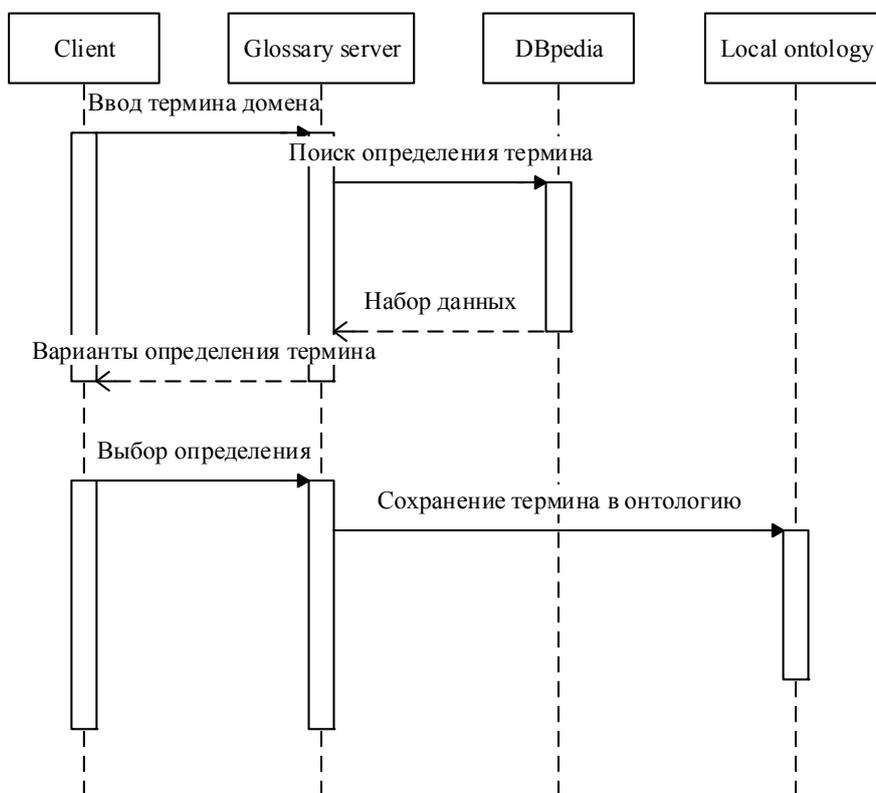


Рис. 7

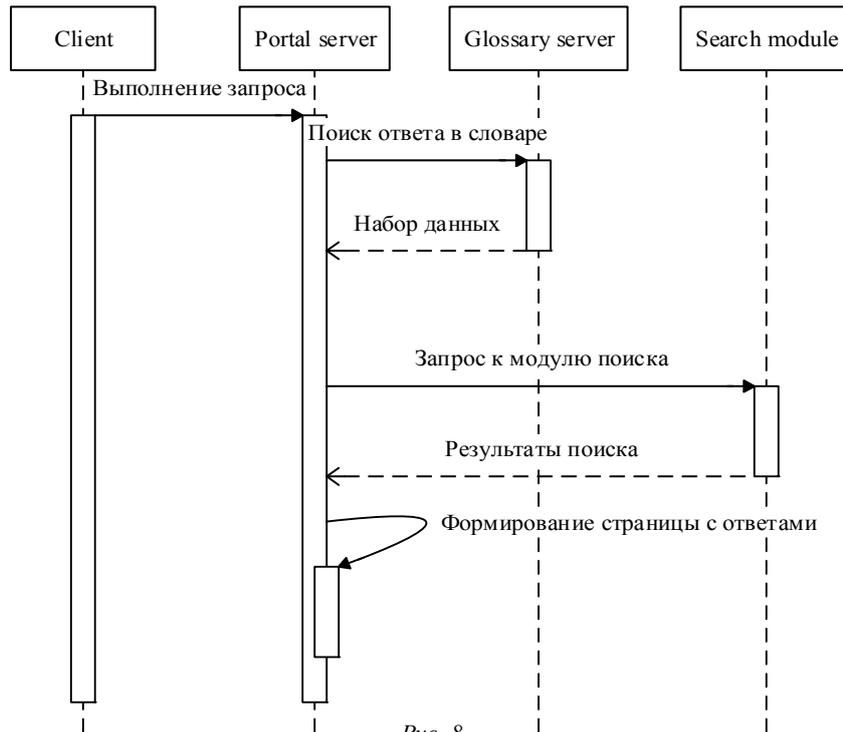


Рис. 8

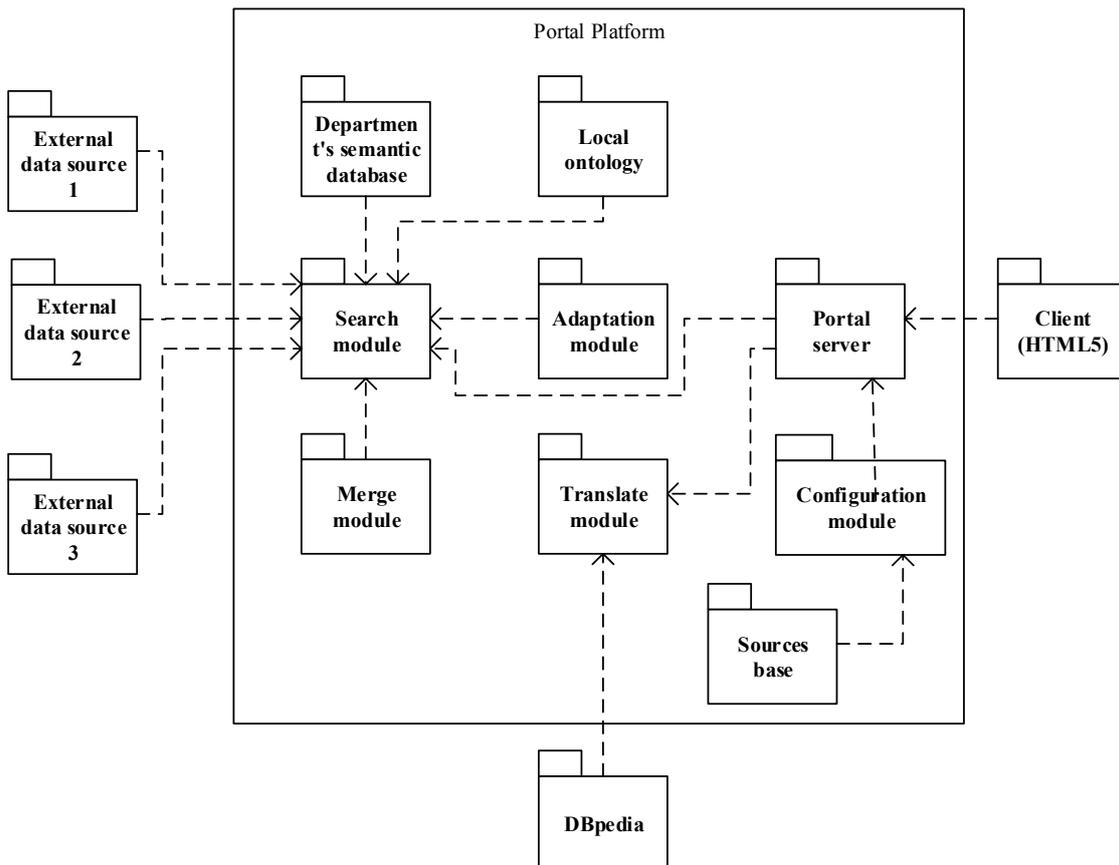


Рис. 9

тический портал, то поиск по нему повысит перенос знаний внутри организации);

– наличие модуля адаптации (если поиск использует интересы пользователей, то это повысит

качество получаемых данных (пользователь будет получать только интересующие его данные).

Архитектура без модуля перевода. Если перевод терминов не требуется (язык терминов совпа-

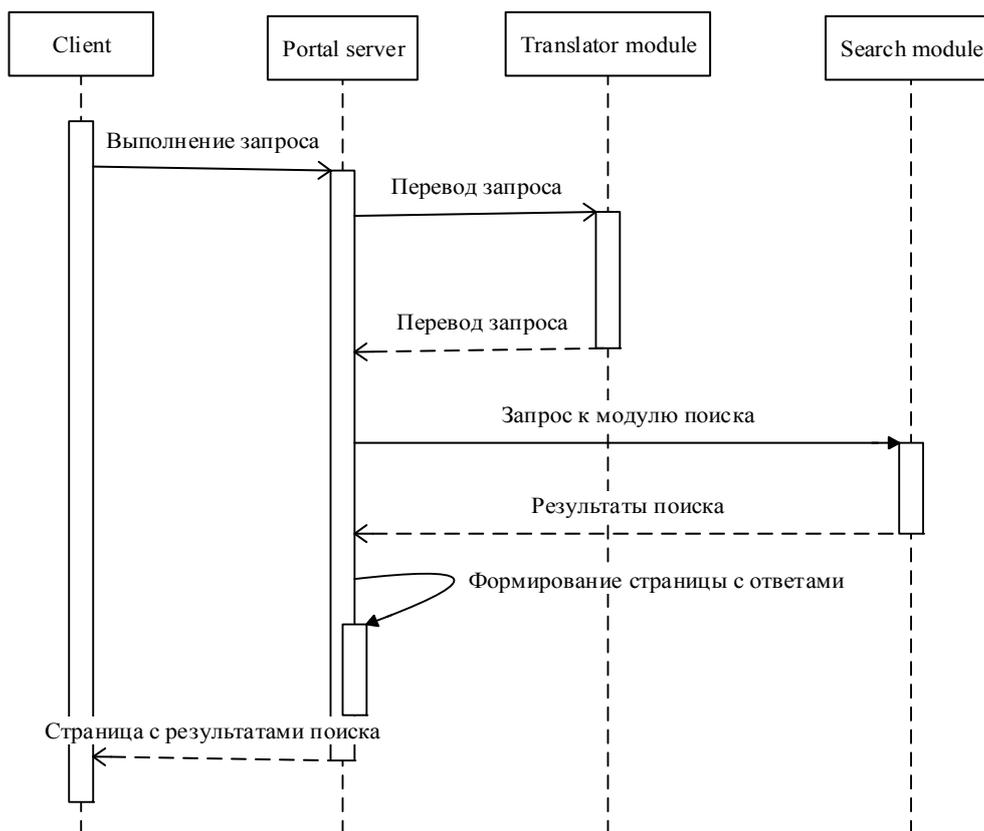


Рис. 10

дает с языком источников данных или используется только английский язык), то модуль перевода можно исключить (рис. 6).

При этом сценарий перевода будет отсутствовать полностью. Сценарий заполнения словаря без перевода представлен на рис. 7, и главное его отличие – в отсутствии перевода термина.

Сценарий запроса без перевода представлен в виде диаграммы последовательности нотации UML (рис. 8). В данном случае система реализует запросы к источникам данных без перевода ключевых слов.

Архитектура без словаря. Если для предметной области не требуется словарь, то при развертывании системы и поиске им нужно пренебречь (рис. 9). При этом сценарий заполнения словаря будет отсутствовать полностью.

Сценарий запроса без словаря описан при помощи диаграммы последовательности (рис. 10). Отличие в том, что при поиске система будет учитывать только данные, полученные из источников данных.

Архитектура без семантического портала. Если организация не имеет своего семантического портала, то при развертывании системы от не-

го можно отказаться. Архитектура в этом случае представлена на рис. 11.

Сценарии без использования семантического портала. Сценарий регистрации показан на рис. 12. Главное отличие – в отсутствии получения данных из семантического портала организации. Данные о пользователе будут состоять только из тех, которые он вводит сам.

Сценарий поиска без семантического портала представлен на рис. 13. В данном случае данные ищутся только во внешних источниках.

Архитектура без адаптации. Если требуется получать «сырые» данные без адаптации, можно отказаться от модуля адаптации (рис. 14).

Сценарий без использования механизма адаптации данных. Сценарий поиска без использования механизма адаптации описан при помощи диаграммы последовательности нотации UML (рис. 15). В данном случае полученные данные не адаптируются под пользователя, а только сливаются.

Минимальная архитектура. Можно определить минимальную архитектуру портала, при которой он будет выполнять только базовые функции поиска. Минимальная архитектура определяет минимальный набор сервисов, при котором портал будет функционировать.

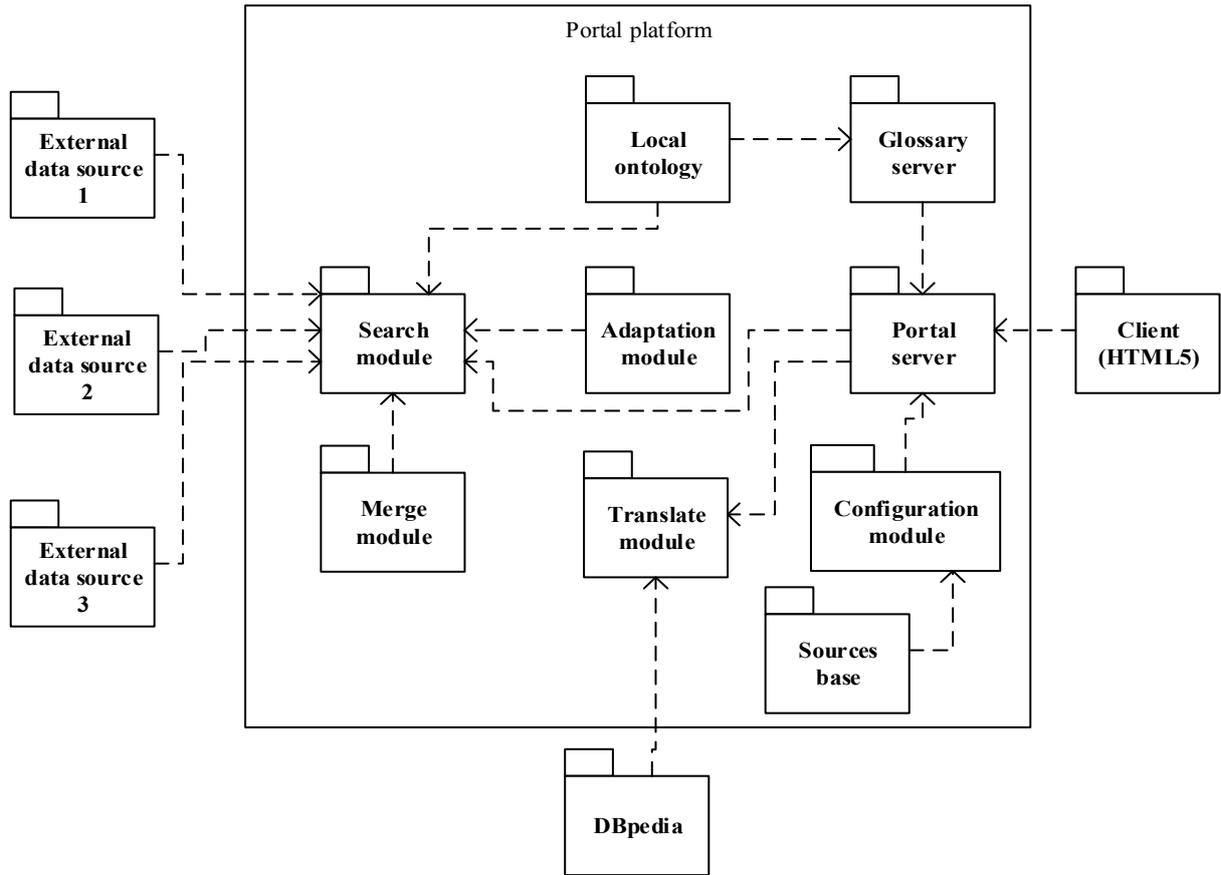


Рис. 11

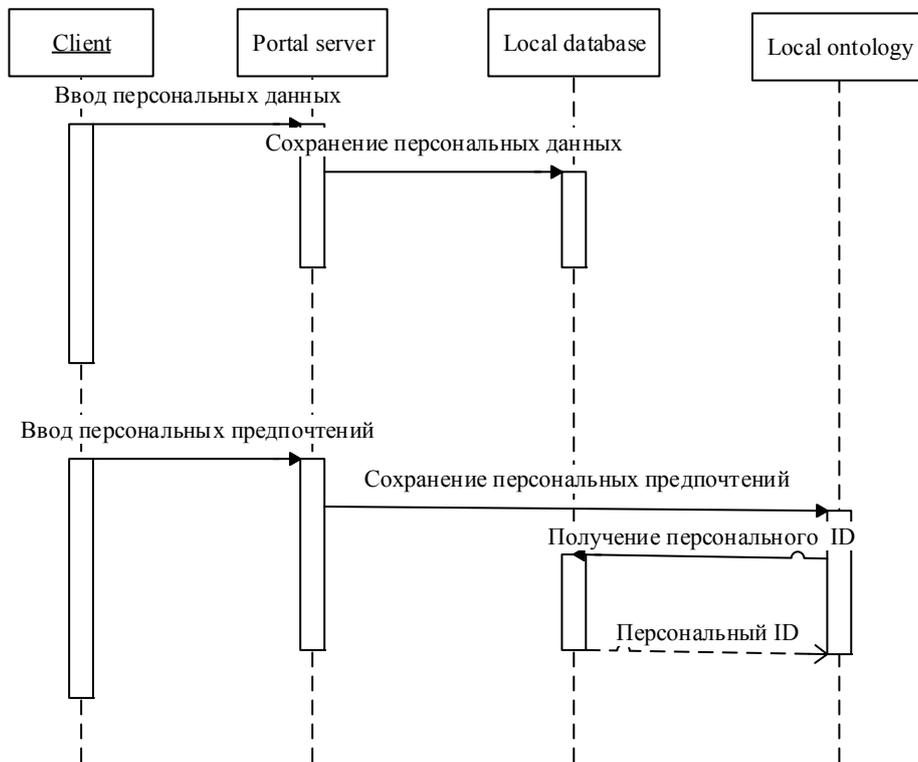


Рис. 12

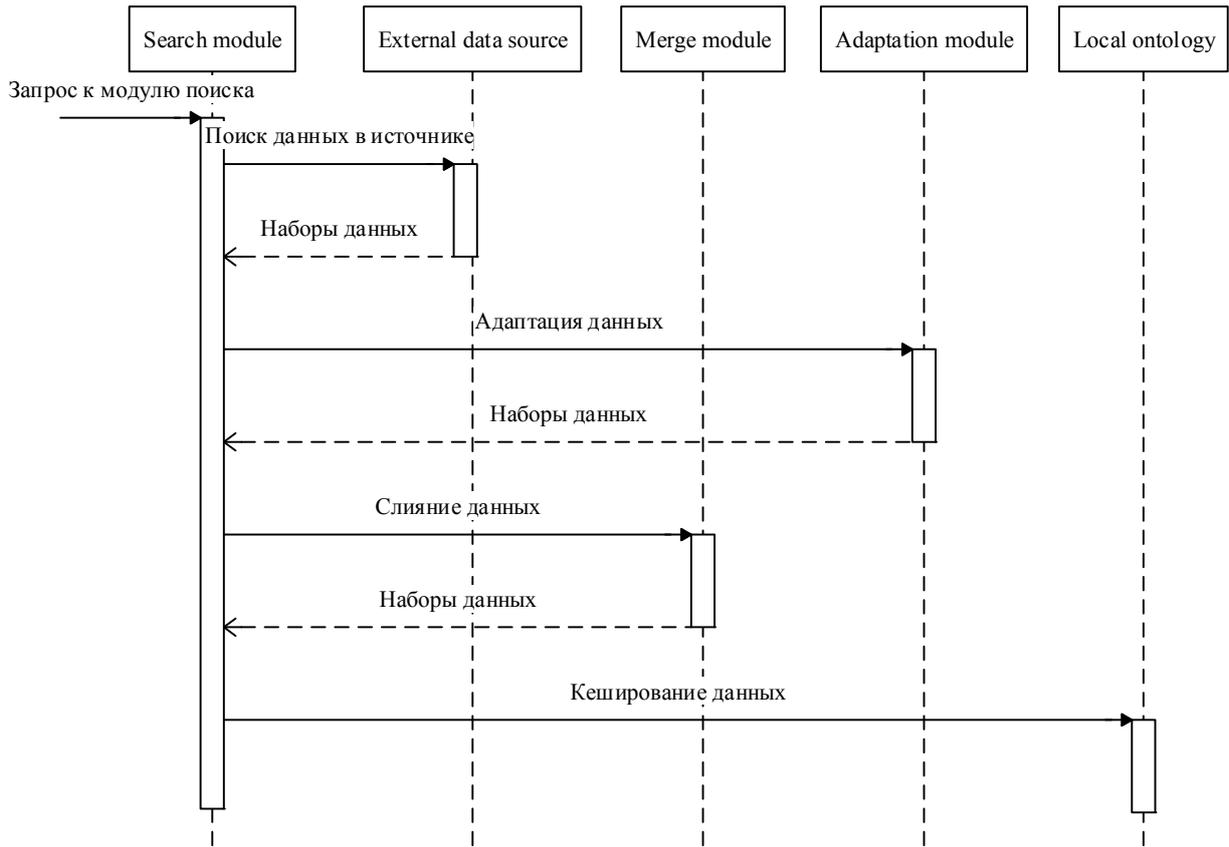


Рис. 13

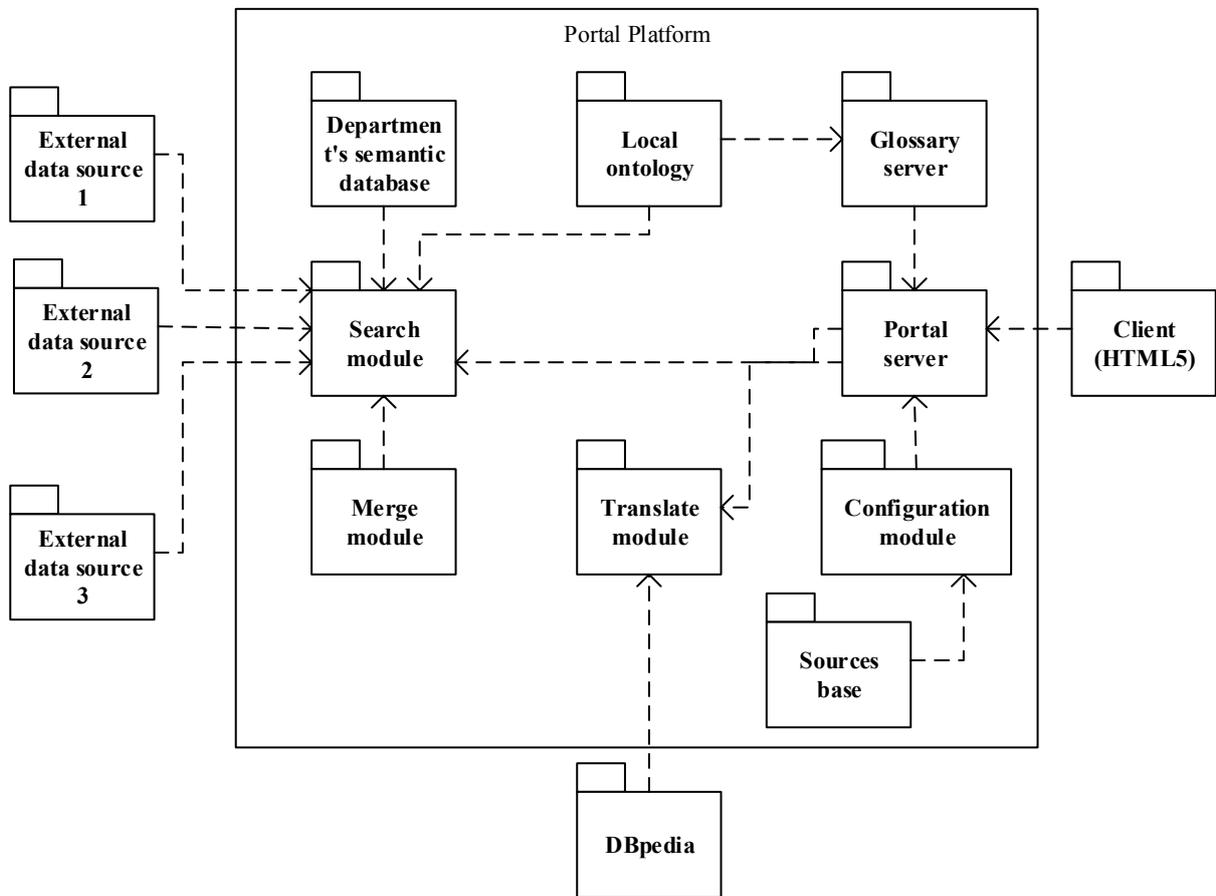


Рис. 14

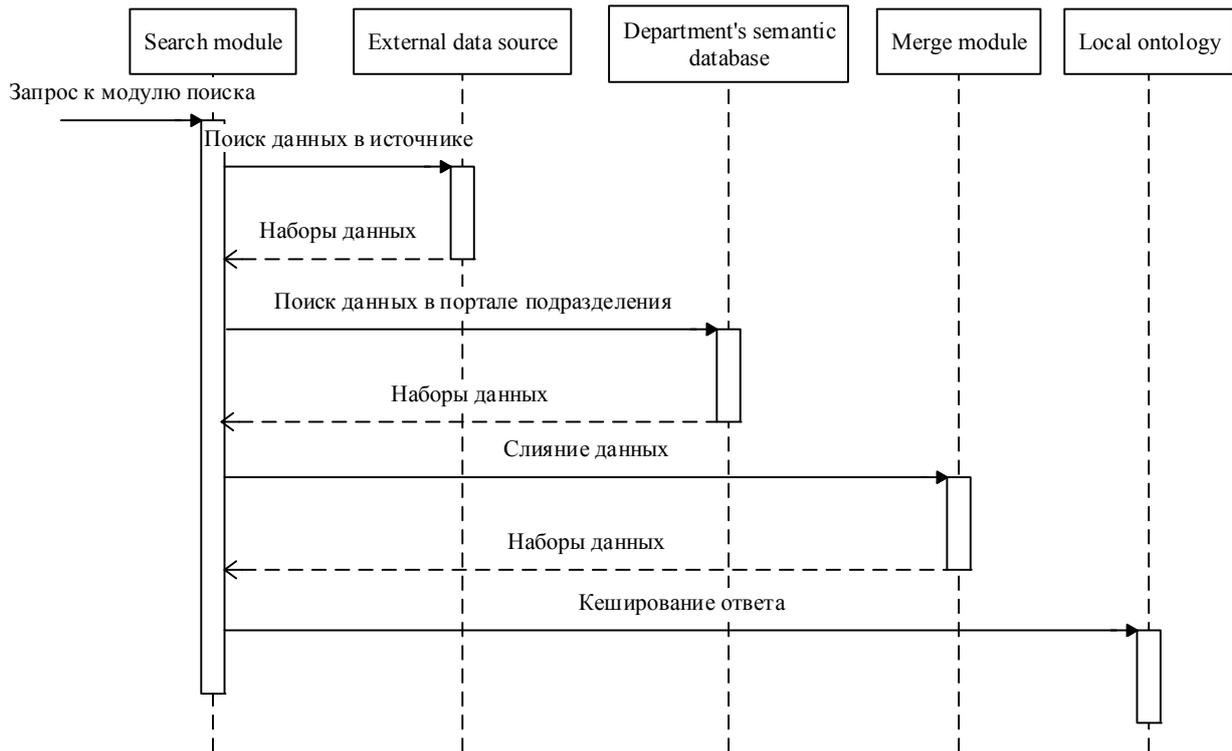


Рис. 15

При этом будут отсутствовать сценарии перевода, заполнения словаря. Измененные сценарии представлены далее. Сценарий регистрации состоит только из ввода данных пользователей, отсутствует взаимодействие с семантическим порталом организации. В сценарии запроса отсутствует взаимодействие с семантическим порталом и сервисом перевода термина. Также нет взаимодействия со словарем.

Сценарий поиска также не поддерживает взаимодействие со словарем, сервисом перевода и семантическим порталом организации.

Предлагаемые архитектурные решения могут быть использованы для построения научно-образовательных порталов образовательных учреждений высшего профессионального образования при реализации образовательных программ, связанных с техническими направлениями, относящихся к уровню магистерской и аспирантской подготовки.

СПИСОК ЛИТЕРАТУРЫ

1. Ахмедов А. Э., Смольянинова И. В., Шаталов М. А. Формирование индивидуальной образовательной траектории в условиях непрерывного образования // Территория науки. 2015. № 5. С. 7–11.
2. Eriksson H.-E., Penker M. Business modeling with UML. New York: John Wiley & Sons, 2000. 459 p.
3. Ньюмен С. Создание микросервисов. СПб.: Питер, 2016. 304 с.
4. Bendiken A., Lavender B. RDF.rb is a pure-Ruby library for working with Resource Description Framework (RDF). URL: <https://rubygems.org/gems/rdf>.
5. RDF.rb: Linked Data for Ruby. URL: <https://libraries.io/github/bendiken/rdf>.
6. Merkel D. Docker: Lightweight linux containers for consistent development and deployment. URL: <http://www.linuxjournal.com/content/docker-lightweight-linux-containers-consistent-development-and-deployment>
7. Virtuoso Open-Source Edition. URL: <https://virtuoso.openlinksw.com/dataspace/doc/dav/wiki/Main/>.
8. Puma: A Modern, Concurrent Web Server for Ruby. 2017. URL: <http://puma.io>.

M. A. Navrotsky, D. I. Muromtsev
University ITMO

N. A. Zhukova, A. I. Vodyaho
Saint Petersburg Electrotechnical University «LETI»

ARCHITECTURAL SOLUTIONS FOR OF SCIENTIFIC INFORMATION PORTALS

A new class of information systems – educational scientific information portals is described. The set of reference architectural solutions, which can be used for different purposes, depending on the requirements and the subject domain, is presented. Specific feature of the proposed architectural solutions is the use of a service oriented approach for building the subsystem and modules. This approach allows to tune the system according to the requirements of each organization and the subject domain. An essential advantage of the suggested approach as compared to the existing one is the possibility of using standard interfaces for services interaction, making it possible to replace any module when necessary. Each service can be realized in such a way that they can be packed into separate containers, which enables services to be deployed in different clouds or hosts in order to optimize the resource usage. The questions of developing and building a software platform for domain-specific research and educational portals implementation are also discussed in the paper.

Scientific informational portals, ontology, open linked data, search in open data

УДК 004.451.44:004.942

С. А. Даденков, А. Н. Кокоулин, Е. Л. Кон
Пермский национальный исследовательский политехнический университет

Разработка имитационной модели циклического алгоритма планирования задач реального времени

Разрабатывается имитационная модель циклического алгоритма планирования задач на основе приоритетов (round robin priority driven preemptive scheduler), используемого на узлах-датчиках, -контроллерах, -исполнительных механизмах распределенных fieldbus-сетей мягкого реального времени, в том числе LonWorks и BacNet. Предлагаемая модель отличается от известных аналитических и имитационных аналогов учетом ранее не анализируемых значимых факторов функционирования алгоритма: индивидуальных номеров, приоритетов, конфигурационных параметров задач. Модель создается в системе имитационного моделирования AnyLogic и дополняется элементами визуализации результатов оценки вероятностных и временных характеристик обработки задач. Для разработки использован графический инструментарий диаграмм состояний и переходов совместно с программным расширением модели. Указанное позволило создать легко масштабируемую и корректную модель, пригодную для оценки характеристик и проектирования структуры планировщика задач реального времени.

Циклический алгоритм планирования задач, реальное время, промышленные fieldbus-сети, имитационное моделирование, вероятностные и временные характеристики

Циклический алгоритм (Round-robin) широко используется программными планировщиками задач в вычислениях реального времени. Распространение алгоритм получил за счет детерминированности интервалов времени решения задач в системах мягкого и жесткого реального времени. Различные модификации алгоритма применяются при пакетном планировании задач передачи дан-

ных в компьютерных и сенсорных сетях, для сбора, обработки и передачи технологической (измерительной и управляющей) информации в распределенных системах управления (Distributed control system), в том числе построенных на основе сетевых fieldbus-технологий. Соответственно применяются алгоритмы: Round Robin, Weighted Round Robin, Deficit Round Robin and