

Distributed Knowledge Base System

Kulykovska N. A., Timenko A. V., Ilyashenko M. B., Kirichek G. G.

Zaporizhzhya National Technical University.

Zaporizhzhya, Ukraine

Abstract. One of the main problems of any distributed system is the analysis of the properties of the obtained data and their further use for logical reasoning. To achieve this goal and accelerate the processing time is necessary in the first phase of the system for the development of the intellectualization control data. The creation of efficient and adaptive distribution can significantly speed up the processing velocity. Thus, the relevant direction of the research is distributed knowledge based systems. The goal of the work was the models of the distributed knowledge based systems and methods for structuring the system. The novelty of the work consisted in the identification of the main artifacts of a distributed knowledge based system. The formal model of the system consisted of a set of ontologies; lots of services; a set of events that describe the processes of the system; semantic service; a set of composite services and knowledge base. In the system under study, the semantic web technologies were used to examine the data, their processing and inference. The completeness and effectiveness of the system was determined by a multitude of ontologies. The main difference in the distributed knowledge based systems was the use of the service approach and ontologies in knowledge engineering. In this case, three methods for structuring the system have been proposed. The time spent for publishing the service in each method is different, but the complexity of the provider is also different.

Keywords: distributed system, knowledge engineering, data, ontology, service, semantic service.

DOI: 10.5281/zenodo.3244818

Sistemul distribuit, bazat pe cunoștințe

Culicovscaia N. A., Timenco A. V., Ilișenco M. B., Chricec G. G.

Universitatea Națională tehnică din Zaporijie

Zaporijie, Ucraina

Rezumat. Una dintre sarcinile principale ale sistemului distribuit este analiza datelor obținute și utilizarea ulterioară a acestora pentru raționamentul logic. Aceasta este o sarcină dificilă datorită dinamicii frecvente și eterogenității informațiilor. Scopul lucrării este de a prezenta un sistem distribuit bazat pe cunoștințele. Principala diferență a sistemelor distribuite bazate pe cunoștințele este utilizarea unei abordări de servicii și a unei inginerii de cunoștințe. În acest sistem, studiul datelor, prelucrarea acestora și inferențele logice au loc prin utilizarea tehnologiilor web semantice. Scopul a fost realizat prin dezvoltarea unui model structural și formal al sistemului, precum și a metodelor de structurare a obiectelor sistemului. Universalitatea modelului conceptual constă în prezentarea acestuia sub forma unei diagrame de clasă UML, care va face posibilă implementarea componentelor sistemului ținând cont de orice cerințe ale sferei IT și ale proceselor de afaceri. Modelul oficial reflectă funcțiile dinamice ale sistemului pentru organizarea fluxului de informații între obiecte. Noutatea științifică a rezultatelor obținute constă în identificarea principalelor artefacte ale unui sistem distribuit bazat pe cunoaștere, și anume setul de servicii, setul de ontologii, serviciul semantic, setul de evenimente care descriu procesele sistemului și baza de cunoștințe. Completitudinea și eficacitatea sistemului este determinată de setul de ontologii și constă în trei tipuri: ontologia aplicației, ontologia domeniului și ontologia sarcinilor. Utilizarea ontologiilor este motivată de faptul că acestea conțin informații structurate despre caracteristicile funcționale și non-funcționale ale serviciului și a zonei subiectului, oferind flexibilitate descrierii datelor. Prin implementarea ontologiilor, se face o tranziție în procesarea inteligentă a datelor.

Cuvinte-cheie: sistemul distribuit, componentele sistemului distribuit, ingineria cunoștințelor, date, ontologie, service, web semantic, modelul sistemului.

Распределенная система, основанная на знаниях

Куликовская Н. А., Тименко А. В., Ильяшенко М. Б., Киричек Г. Г.

Запорожский Национальный Технический Университет

Запорожье, Украина

Аннотация. Одной из основных задач распределенной системы является анализ полученных данных, и дальнейшее их использование для логических рассуждений. Это является сложной задачей из-за часто встречающейся динамики и разнородности информации. Целью работы является представление распределенной системы, основанной на знаниях. Главным отличием, распределенных систем, основанных на знаниях, является использование сервисного подхода и инженерии знаний. В

рассматриваемой системе изучение данных, их обработка и логический вывод происходит путем использования технологий семантического веба. Поставленная цель достигнута за счет разработки структурной и формальной модели системы, а также методов структуризации объектов системы. Универсальность концептуальной модели заключается в представлении ее в виде диаграммы классов UML, что позволит реализовывать компоненты системы с учетом любых требований ИТ-сферы и бизнес-процессов. Формальная модель отражает динамические функции системы для организации потоков информации между объектами. Научная новизна полученных результатов заключается в определении основных артефактов распределенной системы, основанной на знаниях, а именно, множества сервисов, множества онтологий, семантического сервиса, множества событий, которые описывают процессы системы, и база знаний. Полнота и эффективность системы определяется множеством онтологий и состоит из трех видов: онтология приложения, доменная онтология и онтология задач. Использование онтологий аргументировано тем, что они содержат в себе структурированную информацию о функциональных и нефункциональных характеристиках сервиса и предметной области, придают гибкость описанию данных. Путем внедрения онтологий осуществляется переход в интеллектуальной обработке данных. Впервые предложены методы структуризации распределенной системы, основанной на знаниях, в зависимости от возможностей и ресурсов разработчика сервиса. Создание онтологий довольно трудоемкий и специфический процесс, поэтому в большинстве случаев отказываются от семантических описаний. С другой стороны, попытки внедрить онтологии оказываются тщетными, потому что многие системы не имеют возможности работать с таковыми. Разработанные методы позволяют организовать такую систему, которая может создавать онтологии сервисов и аккумулировать их извне.

Ключевые слова: распределенная система, компоненты распределенной системы, инженерия знаний, данные, онтология, сервис, семантический веб, модель системы.

ВВЕДЕНИЕ

В современном обществе существует необходимость в повышении качества и скорости обработки данных в распределенных системах (РС) [1, 2]. В связи с этим возрастает значение распределенных систем, ориентированных на эффективное управление данными [3, 4], как средства решения этой проблемы. Это является сложной задачей из-за динамики и разнородности информации, что накладывает очень частые изменения в локальные значения, которые влияют на общие глобальные свойства всей системы. Создание интеллектуальных и адаптивных РС позволяет значительно ускорить скорость обработки данных.

Тем не менее, РС в условиях развития современных технологий ставят акцент на свойствах интероперабельности и масштабируемости. Такое направление связано со стремительным ростом технологии блокчейн и интернета вещей (IoT). Как прогнозируют аналитики, к 2025 году на долю блокчейн-приложений будет приходиться 10% мирового валового внутреннего продукта [5]. К 2020 году будет создано более 30 миллиардов устройств IoT [6]. Такие технологии меняют представления данных, модернизируют способы и формы коммуникации, производства и потребления информации [7]. Основные изменения заключаются в доминировании горизонтальных связей, структурообразующей роли информации, децентрализации всех ви-

дов данных, доступных в любое время на каждом устройстве. Пользователь не должен заботиться о конкретных технологиях, используемых для предоставления мощности вычислений или хранения данных, поэтому можно говорить о том, что пользователь работает с некоторой информацией об удаленном ресурсе.

Таким образом, актуальным направлением исследований является модели и методы распределенных систем, основанных на знаниях (PCO3).

МЕТОДЫ, РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

В PCO3 с целью изучения данных, их обработки и использования внедряются технологии семантического веба. Семантический веб не заменяет возможности РС, а дополняет их: в то время, как задачей большинства современных веб-сервисов является обеспечение связи между приложениями. Семантический веб решает более сложную проблему – повышает релевантность поиска информации в глобальной сети [8, 9]. Для решения этой задачи необходимо построить сеть, состоящую не из разнородных, несовместимых между собой документов, а из семантически структурированных объектов, с описанием связей и взаимоотношений между ними.

Другой аспект семантической сети предполагает такие методы понимания веб-контента, как поведение человека [10]. Это связано с машинным обучением и искус-

ственным интеллектом. Семантическая сеть в основном пытается научить компьютер понимать и обрабатывать данные.

В семантической сети представление знаний – это систематизированная методика описания на машинном уровне того, что знает человек-эксперт, специализирующийся в предметной конкретной области [11].

Существует много способов описания знаний. Например, знание может быть выражено в исходном коде программы или неявно в весовых коэффициентах нейронной сети. Но исследования в области представления знаний направлены на разработку явных и декларативных моделей представления [12]. Ясность представления означает, что знание должно содержаться в базе знаний в явном и недвусмысленном виде. Декларативность означает, что смысл описанных знаний можно установить без необходимости рассматривать, как оно применяется при логическом выводе.

Существует определение системы, основанной на знаниях – это программа для компьютера, которая оперирует со знаниями в определенной предметной области с целью выработки рекомендаций или решения проблем [13].

Такие системы могут служить полной заменой человека-эксперта в данной предметной области или же быть интеллектуальным помощником человека, принимающего решения. Рассматриваемые системы предназначены для решения так называемых трудно формализуемых задач [14], т.е. таких задач, которые обладают одной или несколькими из перечисленных характеристик:

- не существует алгоритмического решения задач;
- задачи не могут быть заданы в числовой форме;
- цели не могут быть заданы в терминах точно определенной числовой функции.

Особенностями систем, основанных на знаниях, отличающими их от систем с алгоритмическим подходом, являются [13]:

- моделирование не столько природы определенной предметной области, сколько механизма мышления эксперта при решении задач данной предметной области;
- формирование выводов, основываясь на тех знаниях, которыми располагает система;

- знания в системах представлены на некотором специальном языке и содержатся в базе знаний системы.

Системы, основанные на знаниях, применяются для решения задач интерпретации, прогноза, диагностики, проектирования, планирования, мониторинга, наладочных работ, оказания помощи при ремонте, обучения и контроля в разнообразных проблемных областях [14, 15, 16, 17].

Из известных работ по применению систем, основанных на знаниях, для анализа и повышения производительности компьютерных систем, можно отметить следующие:

- в [18] приведен общий обзор и сравнение подходов, применяемых в экспертных системах для анализа производительности компьютерных программ;
- в [19] описано использование производственных правил для выявления причин недостаточной производительности распределенных веб-приложений и выдаче рекомендаций по улучшению программы;
- в [20] предложено использование системы, основанной на знаниях, для проектирования программных систем реального времени;
- в [21, 22] рассмотрены средства семантического анализа кода, для интеллектуального рефакторинга в IDE IntelliJ IDEA;
- в [23, 24] рассмотрено применение семантических технологий в организации семантического GRID.

Целью работы является представление модели распределенной системы, основанной на знаниях. Главным отличием, распределенным систем, основанных на знаниях, является использование сервисного подхода и инженерии знаний. В данной статье мы рассмотрим методы РСОЗ, которые являются синтезом существующих технологий распределенных систем и семантических сетей.

Поставленная цель достигается путем формального описания модели системы, а также разработки методов структуризации системы. Научная новизна полученных результатов заключается в определении основных артефактов распределенной системы, основанной на знаниях, а именно, множества сервисов, множества онтологий, семантического сервиса, множества событий, которые описывают процессы системы, и база знаний. Полнота и эффективность системы определяется множеством онтологий и состоит из трех

видов: онтология приложения, доменная онтология и онтология задач. Использование онтологий аргументировано тем, что они содержат в себе структурированную информацию о функциональных и нефункциональных характеристиках сервиса и предметной области, придают гибкость описанию. Путем внедрения онтологий осуществляется переход в интеллектуальной обработке данных.

Впервые предложены три метода РСОЗ: нисходящий, восходящий и смешанный. В виду причин того, что РСОЗ зависит от операций между сервисом, семантическим сервисом и базой знаний, а это в свою очередь программные объекты, чтобы сравнить каждый метод мы примем каждую операцию за единичное действие, стремящееся к наименьшему времени выполнения. Следовательно, чтобы оценить затраченное время для каждого метода, достаточно посчитать количество выполненных операций для формирования РСОЗ.

Преимуществом нисходящего метода является большая вероятность точного и релевантного поиска, быстрого роста базы знаний т.к. онтологии генерируются одним объектом системы – семантическим сервисом. При автоматической генерации онтологий появляется возможность упростить процесс представления знаний РСОЗ.

Так же, при переходе существующей РС на РСОЗ все трудозатраты сводятся к разработке доменных онтологий, которые в дальнейшем используются как базис для знаний. Затраченное время на процесс регистрации сервиса занимает 8 итераций.

В восходящем методе сервис может развиваться независимо от других ресурсов или их онтологий, поэтому при таком построении архитектуры сравнительно легко вносить изменения в сервисы, добавлять новые и т.д. (улучшаются качественные характеристики архитектуры: расширяемость, масштабируемость, способность к эволюционным изменениям).

Однако процессы добавления и выделения доменных онтологий являются очень затруднительными. При данном построении затраченное время будет состоять из 9 итераций.

При смешанном методе регистрация нового узла выполняется в 15 итераций. Но при таком увеличении затраченного времени, он наследует все преимущества двух описанных ранее методов. Семантический сервис выполняет функции валидации и верификации

онтологий, при этом база знаний аккумулирует данные извне.

Результаты статьи имеют научную и практическую ценность в области РС. Предложенные концептуальная и формальная модели РСОЗ описывают структурную организацию системы и однозначно определяют ее основные компоненты. Универсальность концептуальной модели заключается в представлении ее в виде диаграммы классов UML, что позволит реализовывать компоненты системы с учетом любых требований ИТ-сферы и бизнес-процессов. Формальная модель отражает динамические функции системы для организации потоков информации между объектами. Приведены временные UML диаграммы структуризации и представления знаний об объектах системы, которые формализуют каждый этап работы РСОЗ при различных условиях создания онтологий сервисов.

I. МЕТОДЫ ИССЛЕДОВАНИЯ

РС состоит из информационной и аппаратной части. Информационная составляющая определяет интероперабельность компонент системы (сервисы, сетевые протоколы, реестры и т.д.), а аппаратная часть возможность распределенной обработки данных (компьютерные сети, хранилища, компьютерные ресурсы и т.п.).

В виду разнородности информации и множественности технологий разработки РС, представим каждый объект взаимодействия внутри системы как законченный программно-технический компонент, который характеризуется некоторым набором требований и характеристик для выполнения определенной задачи, а именно как сервис. Сервис, зарегистрированный в системе, получает запрос на предоставление определенных услуг и возвращает результат. Существует множество технологий, обеспечивающих создание и сопровождение сервисов в распределенных вычислительных системах: технология XML веб-сервисов, сервисы REST и др., но в большинстве из них в основе лежит сервис-ориентированная архитектура (COA) (рисунок 1). [25, 26].

COA обеспечивает масштабируемость и гибкость растущих систем и является промежуточным слоем между бизнесом и информационными технологиями. COA обеспечивает работу в распределенной гетерогенной среде, что является показателем универсальности и совместимости с другими системами.

Описанию моделей и методов СОА уделено большое количество работ. Основными разработчиками стандартов в данном направлении являются такие корпорации как IBM, OASIS, OMG, The OpenGroup, также в интеграции СОА с существующими системами посвящены работы таких зарубежных авторов, как N.Josuttis, M. Juric, S. Carter.

В то же время, при всех преимуществах СОА, остается нерешенным вопрос формализации и автоматизации управления информацией для эффективной передачи данных

между сервисами и отделение функциональности сервисов от используемых данных. Так- же СОА имеет ограничения, обусловленные семантической бедностью используемой модели данных XML, среди которых: сложная и дорогостоящая поддержка для масштабных систем с разнородными данными; достаточно хрупкие правила трансформации данных, ограничивающие динамику системы; отсутствие решения проблем качества данных, дублирования данных.

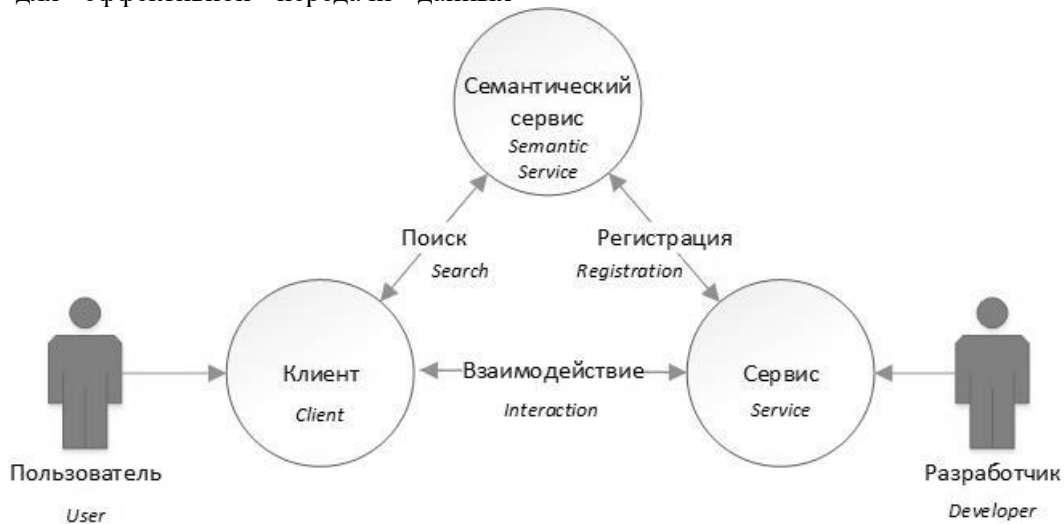


Рис. 1. Сервис-ориентированная архитектура.¹

В свою очередь, структура системы, основанной на знаниях, приведена на рис. 2. Основными ее компонентами являются [15, 27, 28, 29, 30]:

- база знаний. База знаний предназначена для хранения знаний о предметной области. Ее конкретный вид сильно зависит от выбранной модели представления знаний. Наличие этого компонента является главным отличием систем, основанных на знаниях, от других типов программ;
- машина вывода. Машина вывода формирует ответ на запрос пользователя с помощью базы знаний. Принцип ее работы также очень сильно зависит от выбранной модели представления знаний.
- редактор базы знаний – программа для изменения содержимого базы знаний;
- интерфейс пользователя – механизм, с помощью которого происходит общение пользователя и системы.

В отличие от РС, где для разработки и функционирования системы достаточно участия эксперта в предметной области и про-

граммиста, в разработке систем, основанных на знаниях, участвуют:

- эксперт в предметной области, задачи которой будет решать создаваемая система;
- инженер по знаниям – специалист по разработке систем, основанных на знаниях. Он помогает эксперту выявить и структурировать знания, необходимые для работы создаваемой системы, а также выбирает модель представления знаний и инструментальные средства;
- программист. Задачей программиста является разработка инструментальных средств.

PCOЗ стремится к организации системы с наименьшим участием человека в ее функционировании.

Такое направление предполагает полностью представить данные системы и всех ее компонентов в виде базы знаний. Объединяя ключевые элементы РС с системами, основанными на знаниях, можно получить модель PCOЗ.

-

¹ Appendix 1

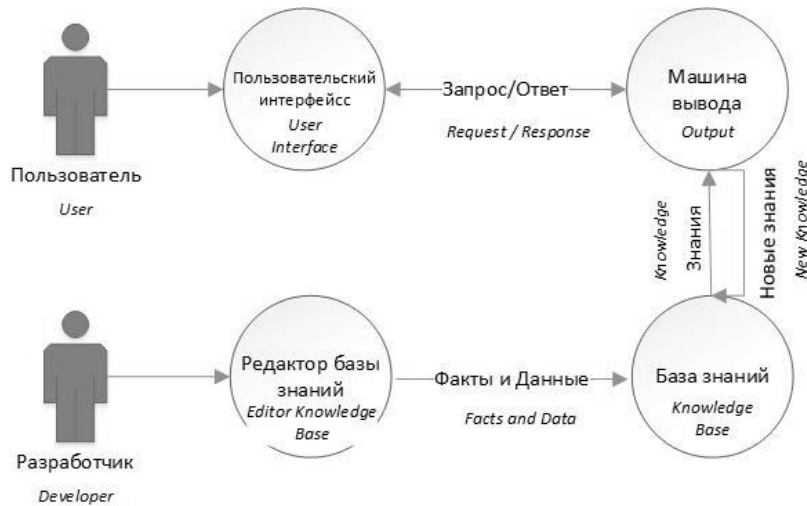


Рис. 2. Структура системы, основанной на знаниях. ²

На рисунке 3 приведена структурная модель PCOЗ. Выделим основные артефакты PCOЗ как сервис, система, событие, разработчик, семантический сервис. Из диаграммы видно, что класс сервис состоит из двух элементов: описание сервиса и онтология приложения. Представление данных в виде знаний отражено через такие элементы: доменная онтология, онтология приложения, онтология задачи и база знаний.

II. ФОРМАЛЬНАЯ МОДЕЛЬ PCOЗ

Структурная модель PCOЗ (англ. Distributed Knowledge Based System – DKBS) позволяет описать формальную модель системы, построенной на данной архитектуре:

$$DKBS = \langle O, S, E, SS, Syst \rangle, \quad (1)$$

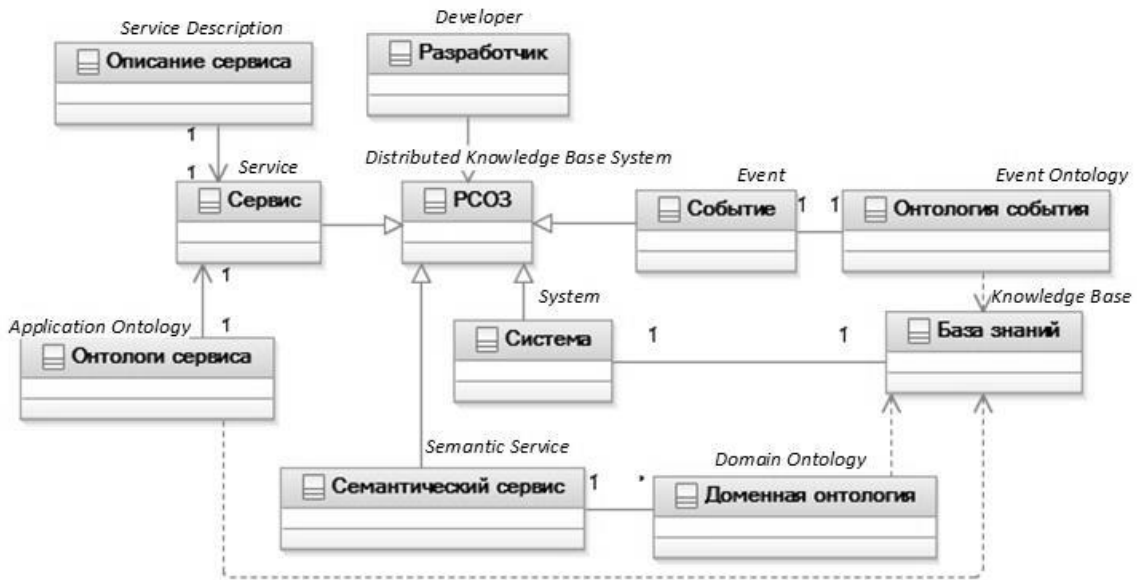


Рис. 3. Распределенная система, основанная на знаниях. ³

где O - множество онтологий;
 S - множество сервисов;
 E - множество событий, которые описывают процессы системы;
 SS -- семантический сервис;

$Syst$ - система из композитных сервисов и базы знаний.

Полнота и эффективность PCOЗ зависит от множества онтологий, которые обеспечивают ее семантический принцип:

$$O = \{O_{Task}, O_{Application}, O_{Domain}\}, \quad (2)$$

где O_{Task} – онтология событий PCOЗ;

$O_{Application}$ – онтология сервисов;

O_{Domain} – онтология предметной области применения PCOЗ.

Рассматривая все виды онтологий [31, 32, 33, 34], мы остановили свой выбор именно на трех, потому что онтология сервиса является моделью сервиса и описывает его функциональные и нефункциональные характеристики. Доменная онтология является неким базовым набором знаний для каждой предметной области. Онтология задачи хранит в себе информацию о всех функциях и действиях системы.

Каждая онтология имеет свои выразительные возможности в зависимости от функционального назначения [35, 36, 37]. Доменные онтологии описывают словарь терминов конкретной предметной области и формально задаются четырьмя конечными подмножествами: понятий (концептов), связей, аксиом и функций интерпретации. В доменных онтологиях отражаются данные конкретной PCOЗ. Онтология задачи состоит из словаря терминов, специализируют задачи и действия в PCOЗ. Всякая задача имеет различные состояния и этапы ее выполнения. Главным отличительным признаком для всех понятий онтологии задачи является время. Онтологии сервиса наиболее специфические онтологии, кроме всех основных понятий, содержит специализированные термины и экземпляры сервисов системы. Семантический сервис включает в себя доменные онтологии и набор модулей работы с сервисами и их онтологическим описанием. Множество событий мы сформировали как набор утверждений, которые могут быть применены к PCOЗ и набор аксиом их появления.

В PCOЗ режим решения задачи приобретает другой характер. В системе функционируют сервисы. Их роль - выполнять некие задачи и возвращать результат. PCOЗ стремится к автономной работе без участия пользователя. Особенностью PCOЗ является возможность сгенерировать и объяснить, как был получен результат, ссылаясь на использованные для его получения знания. Для реализации такой возможности используется подсистема формирования пояснений, кото-

рая используется как пользователем системы, так и сервисами.

III. МЕТОДЫ PCOЗ

Под процессом инженерии знаний PCOЗ понимается процесс конструирования базы знаний, который состоит из следующих этапов:

1. Идентификация и сохранение онтологий объектов системы.
2. Валидация и верификация онтологий.
3. Объединения онтологий и вывод логических рассуждений.
4. Сохранение и поиск онтологий.

Использование онтологий как отличительных структурных элементов PCOЗ определяет критерии построения архитектурной модели. Первый критерий – это использованием единой базы знаний, созданной из онтологий. При этом она может предоставлять доменные онтологии для определенных задач, аккумулировать знания и делать логические выводы.

Второй критерий - это множество онтологий сервиса. Каждый информационный ресурс описывается своей собственной онтологией. Сервис может публиковаться с уже разработанной онтологией либо с онтологией, созданной средствами системы.

Учитывая этапы процесса инженерии знаний и критерии построения системы, рассмотрим методы структурирования PCOЗ.

1. Нисходящий метод. Данный метод лучше применять при переходе от существующей PC к PCOЗ, когда уже реализовано определенное количество работающих сервисов. Он основан на использовании единых доменных онтологий. Каждый сервис системы должен быть описан в соответствии с доменной онтологией, которая связана с семантическим сервисом. Доменных онтологий может быть несколько для отдельных структур системы.

Начальными условиями такого метода являются наличие множества сервисов, семантического сервиса, множества доменных онтологий и системы (3).

Дальнейшее взаимодействие компонент системы, происходит следующим образом: генерируется онтология приложения регистрируемого сервиса с помощью семантического сервиса и соответствующей доменной онтологии. Аксиомы и функции получения знаний данной предметной области уже заданы в семантическом сервисе (4).

$$\langle S, SS, O_{Domain} \rangle, \quad (3)$$

$$S + SS + O_{Domain} \xrightarrow{f_{SS}(S)} O_{Application} \quad (4)$$

На рисунке 4 приведена диаграмма последовательности действий данного метода. Разработчик обращается к системе для регистрации своего сервиса. Система, не находя онтологии приложения, обращается к семантическому сервису для создания таковой. Семантический сервис, имея доменную онтологию, созданную экспертом, генерирует онтологию

приложения, прикрепляет ее к сервису, который теперь регистрируется в системе.

2. Восходящий метод. При первичной разработке PCOЗ целесообразней уделить внимание созданию онтологий сервисов отдельно от всей системы. Каждый сервис описывается своей собственной онтологией, которую реализует разработчик (рисунок 5). Данный метод можно описать следующим образом, где начальными условиями являются наличие множества сервисов, семантического сервиса, множества онтологий сервисов и системы.

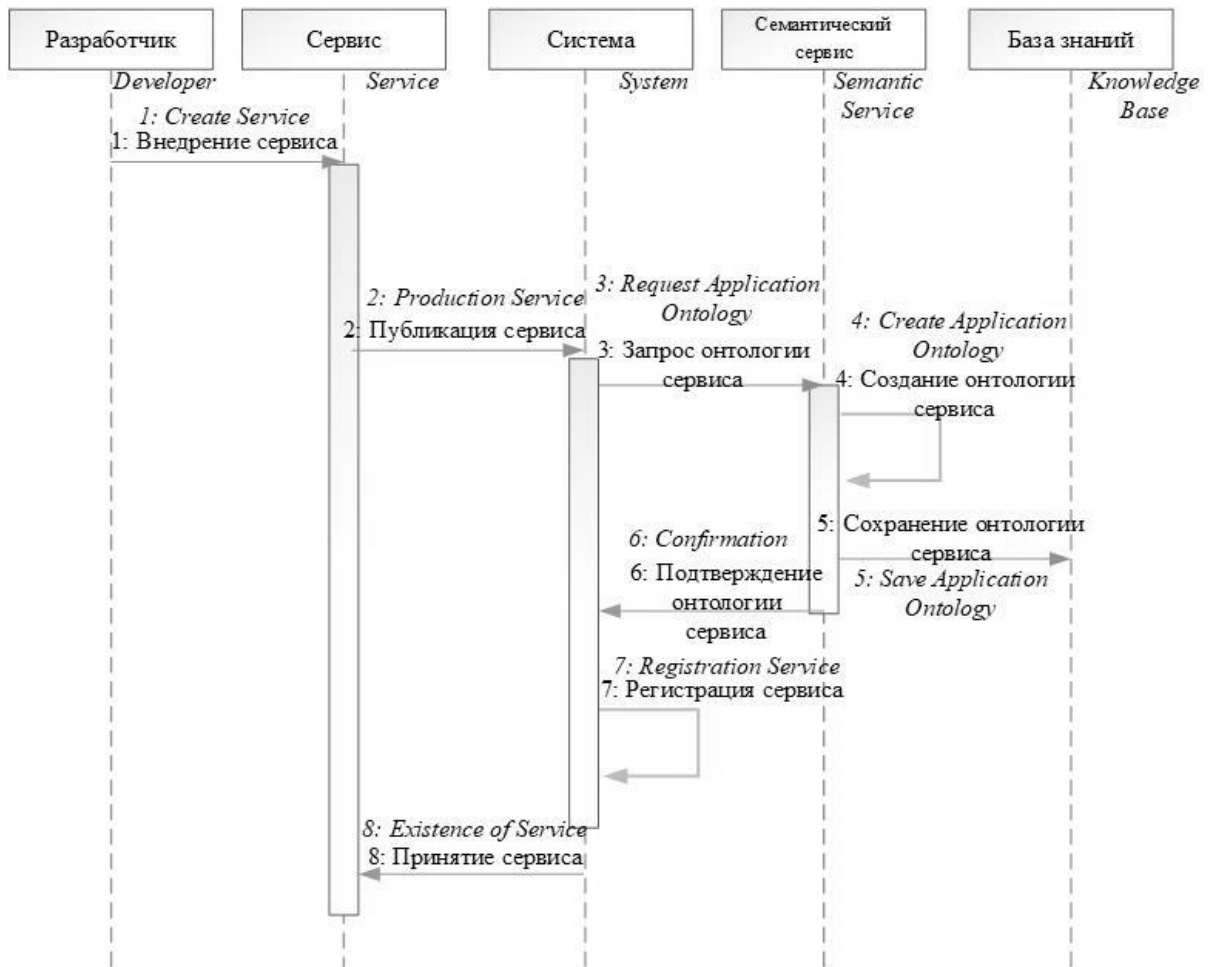


Рис. 4. Нисходящий метод PCOЗ. ⁴

⁴ Appendix 1

В восходящем методе сервис может развиваться независимо от других ресурсов или их онтологий, поэтому при таком построении архитектуры сравнительно легко вносить изменения в сервисы, добавлять новые и т.д.

(улучшаются качественные характеристики архитектуры: расширяемость, масштабируемость, способность к эволюционным изменениям). Однако процессы добавления и выделения доменных онтологий являются очень затруднительными.

$$\langle S, SS, O_{Application} \rangle, \quad (5)$$

$$SS + O_{Application} \xrightarrow{f_{SS}(O_{Application})} true / false. \quad (6)$$

$$SS + O_{Application} \xrightarrow{f_{SS}(O_{Application})} O_{Domain}. \quad (7)$$

В данном методе работа эксперта и инженера по знаниям заключается в наполнении семантического сервиса функциями верификации онтологий. База знаний формируется из полученных онтологий.

3. Смешанный метод – онтология сервиса не генерируется средствами семантического сервиса на основе доменной онтологии, которую получает разработчик (рисунок 6).

$$\langle S, SS, O_{Domain} \rangle, \quad (8)$$

$$S + O_{Domain} \xrightarrow{f_{provider}} O_{Application}, \quad (9)$$

$$SS + O_{Application} \xrightarrow{f_{SS}(O_{Application})} true / false. \quad (10)$$

В данном методе необходимо использование знаний эксперта для формирования доменной онтологии. В таком случае, начальный этап будет самый затратный, но дальнейшая разработка онтологий сервисов будет упрощена.

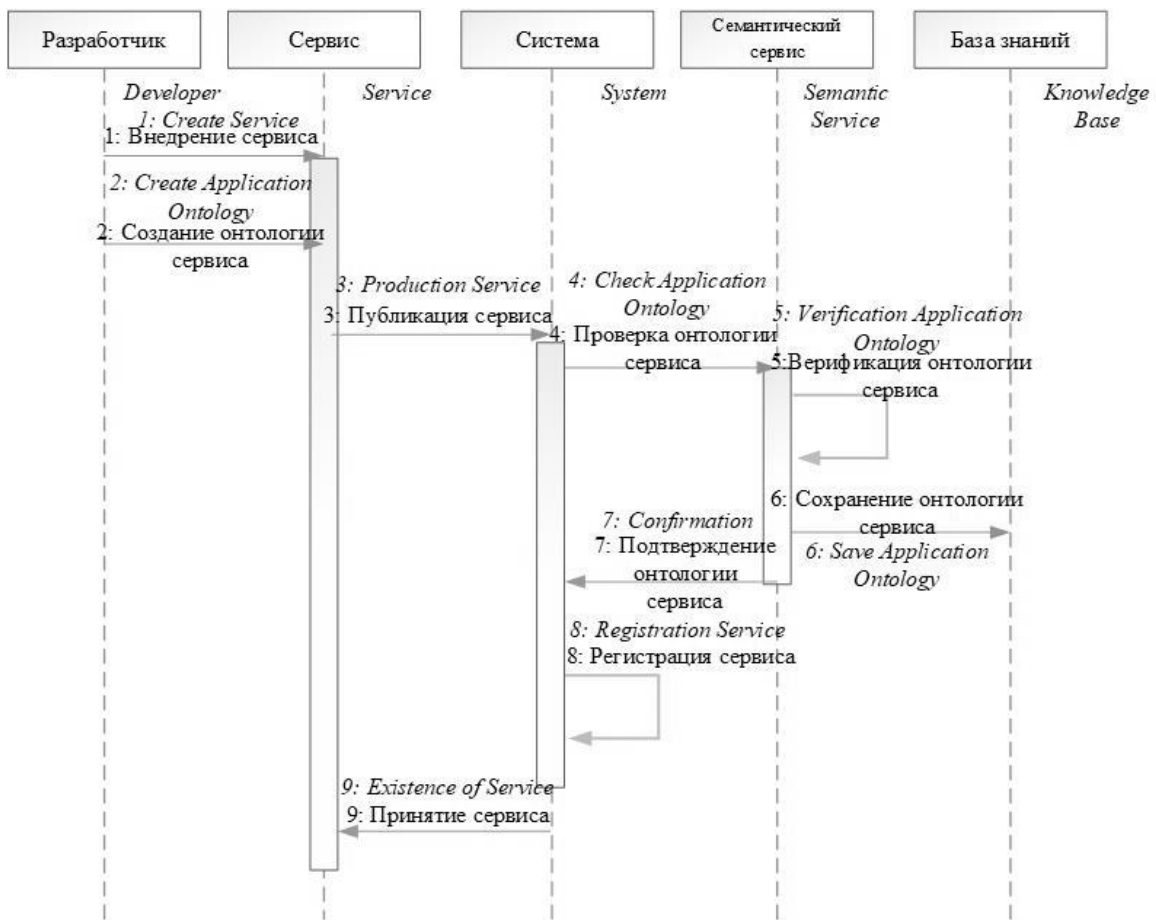


Рис. 5. Восходящий метод PCOЗ. ⁵

В каждом методе используется семантический сервис. Функции семантического сервиса заключаются в работе с онтологиями. Каждая функция характеризуется входными (11) и выходными аргументами (12). Соответственно, каждое действие семантического сервиса (13) характеризуется функцией работы с онтологией, значения ее аргумента при

вызове и результатом завершения для системы.

$$f_{SS} \rightarrow FA_k^{IN} = (fa_1^{IN}, \dots, fa_k^{IN}); k = 1, \dots, K. \quad (11)$$

$$f_{SS} \rightarrow FA_n^{OUT} = (fa_1^{OUT}, \dots, fa_K^{OUT}); k = 1, \dots, K. \quad (12)$$

$$a_{ij} = \{f_k, FAval_i^{IN}, FAval_j^{OUT}\}; \quad (13)$$

$$i = 1, \dots, K; j = 1, \dots, K; f_k \in F.$$

⁵ Appendix 1

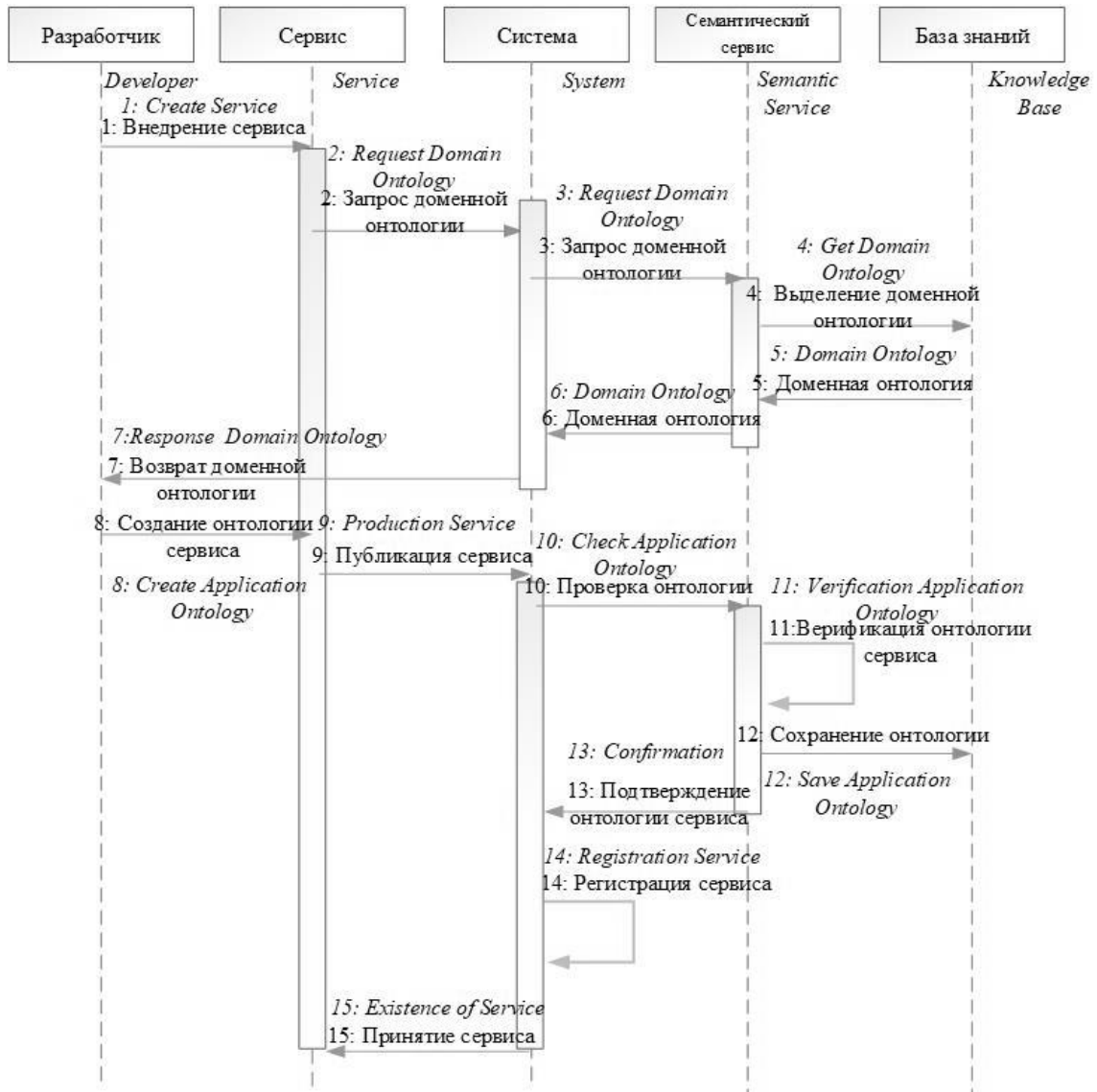


Рис. 6. Смешанный метод РСОЗ.⁶

IV. ЗАКЛЮЧЕНИЕ

В РСОЗ функционируют сервисы. Их роль - выполнять некие задачи и возвращать результат. РСОЗ стремится к автономной работе без участия пользователя. Особенностью РСОЗ является возможность сгенерировать и объяснить, как был получен результат, ссылаясь на использованные для его получения знания. Для реализации такой возможности реализован семантический сервис, который используется как пользователем системы, так и сервисами. Таким образом, главным отличием РСОЗ от распределенных систем и систем, основанных на знаниях является ис-

пользование сервисного подхода и онтологий в инженерии знаний.

APPENDIX 1 (ПРИЛОЖЕНИЕ 1)

¹Fig. 1. Service-oriented architecture.

²Fig. 2. Structure of systems based on knowledge.

³Fig. 3. The diagram of distributed knowledge based system.

⁴Fig. 4. Top-down method distributed knowledge based system.

⁵Fig. 5. Bottom-up method distributed knowledge based system.

⁶Fig. 6. Mixed method distributed knowledge based system.

⁶ Appendix 1

References

- [1] Burns B. *Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services*. O'Reilly Media, Inc, 2018. 166 p.
- [2] Tsvetkov V.Ya. *Bazy dannykh. Ekspluatatsiya informatsionnykh sistem s raspredelennymi bazami dannykh*. [Database. Operation of information systems with distributed databases]. Moskva, 2009. 88 p. (In Russian).
- [3] Wolfgang Dvořák, Sarah Alice Gaggl. Stage semantics and the SCC-recursive schema for argumentation semantics *Journal of Logic and Computation*, 2016, vol. 26(4), pp.1149-1202.
- [4] Shokin YU.I., Zhizhimov O.L., Pestunov I.A., Sinyavskiy YU.N., Smirnov V.V. Raspredelennaya informatsionno-analiticheskaya sistema dlya poiska, obrabotki i analiza prostranstvennykh dannykh [Distributed information and analytical system for searching, processing and analyzing spatial data]. *Vychislitel'nyye tekhnologii*, 2007, vol. 12, no. 3, pp. 108-115. (In Russian).
- [5] *WEB 3.0 budet pitat'sya stekom tekhnologii blokcheyn* [WEB 3.0 will be powered by blockchain technology stack]. Available at: <https://101blockchains.com/ru/web-3-0-%D0%B1%D0%BB%D0%BE%D0%BA%D1%87%D0%B5%D0%B9%D0%BD/> (accessed 01.03.2019)
- [6] *Deloitte projects by 2025 that 10% of global GDP to be built on blockchain applications*. Available at: <https://www.freightwaves.com/news/2017/9/2/deloitte> (accessed 02.09.2017)
- [7] Allemang D, Hendler J. *Semantic web for the working ontologist modeling in RDF, RDFS and OWL*. Elsevier Inc., 2008. 330 p.
- [8] Yelena G.U. Tekhnologii opisaniya semanticheskikh veb-servisov [Technologies for describing semantic web services]. *Yestestvennyye i matematicheskiye nauki v sovremennom mire*. Novosibirsk, 2016, no. 9 (44), pp. 29-34. (In Russian).
- [9] Kulikovskaya N.A. Semanticheskoye opisaniye veb-servisa [Semantic description of web service]. *Nauchnyye trudy Donetskogo natsional'nogo universiteta: vseukrainskiy nauchnyy zbornik. Seriya: Informatika, kibernetika, i vychislitel'naya tekhnika*. Donetsk, 2015, no. 2(21), pp.100-105. (In Russian).
- [10] Srimathi H. Knowledge Representation of LMS using Ontology. *International Journal of Computer Applications*, 2010, vol. 6, no. 3, pp. 35 – 38.
- [11] Wu Z., Palmer M. Verb semantics and lexical selection. *Proceedings of the 32nd Annual meeting of the Associations for Computational Linguistics*. New Mexico, 1994, pp. 133-138.
- [12] Resnik P. Semantic similarity in a taxonomy: An information based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 1999, vol. 11, pp. 95- 130.
- [13] Dzhekson P. *Vvedeniye v ekspertnyye sistemy: per. s angl. V. T. Tertysnogo* [Introduction to expert systems]. Moskva, Vil'yams, 2001. 624 p. (In Russian).
- [14] Negnevitsky M. *Artificial Intelligence: A Guide to Intelligent Systems*. Pearson Education Limited, 2011. 504 p.
- [15] Dzharratano Dzh. *Ekspertnyye sistemy: printsipy razrabotki i programmirovaniye: per. s angl. K. A. Ptitsyna. – 4-ye izd.* [Expert systems: principles of development and programming]. Moskva, Vil'yams, 2007. 1152 p. (In Russian).
- [16] Brogi A., Corfini S. SAM: A Semantic Web Service Discovery System. *Knowledge-Based Intelligent Information and Engineering Systems*, 2008, vol. 4694, pp. 703-710.
- [17] Czajkowski K., Ferguson D., Foster I., Frey J., Graham St., Maguire T., Snelling D., Tuecke St. *From Open Grid Services Infrastructure to WS-Resource Framework: Refactoring & Evolution. Version 1.1. 2004*. Available at: http://toolkit.globus.org/wsrfl/specs/ogsi_to_wsrfl_1.0.pdf (accessed 02.02.2019).
- [18] Velez R., Zhang D., Kho J. An intelligent tool for UNIX performance tuning. *Fifth International Conference on Tools with Artificial Intelligence*, Boston, 1993, pp. 118-122.
- [19] Wolf F., Mohr B. EARL – A Programmable and Extensible Toolkit for Analyzing Event Traces of Message Passing Programs. *Proceedings of the 7th International Conference on High-Performance Computing and Networking*, 2010, pp. 503-512.
- [20] Wolf F., Wylie B. J. N., Abraham E. Usage of the SCALASCA toolset for scalable performance analysis of large-scale parallel applications. *Proceedings of the 2nd International Workshop on Parallel Tools for High Performance Computing*. Stuttgart, 2008, pp. 157-167.
- [21] *Obzor sredstv avtomatizirovannogo refaktoringa v Java IDE* [An overview of automated refactoring in Java IDE]. Available at: http://www.javaportal.ru/java/ide/review_refactoring.html (accessed 01.03.2019)
- [22] Veronika Thost. *Using Ontology-Based Data Access to Enable Context Recognition in the Presence of Incomplete Information (Extended Abstract)* KI, 2017, vol. 31(4), pp. 377—380.
- [23] Zhaohui Wu, Huajun Chen. *Semantic Grid: Model, Methodology, and Applications*. Springer Science & Business Media, 2008. 230 p.
- [24] Foster I., Kesselman C., Nick J. M., Tuecke St. Grid Services for Distributed System Integration. *Computer*, 2002, vol. 35, issue 6, pp. 37-46.

- [25] *The Open Group COA Reference Architecture*. Available at: <http://www.opengroup.org/soa/source-book/soa/index.htm> (accessed 01.03.2019)
- [26] *The Fast Guide to Model Driven Architecture. The Basics of Model Driven Architecture*. Available at: www.omg.org/docs/omg/03-06-01.pdf (accessed 01.03.2019)
- [27] Smolin D. V. *Vvedeniye v iskusstvennyy intellekt: konspekt lektsiy* [Introduction to Artificial Intelligence]. Moskva, 2004. 208 p. (In Russian).
- [28] Behnam Azvine, Nader Azarmi, Detlef D. Nauck. *Intelligent Systems and Soft Computing: Prospects, Tools and Applications*. Springer, 2006, 358 p.
- [29] Gandon F. *Ontology engineering: A Survey and a Return of experience*. INSTITUT DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE, March 2002, 181 p.
- [30] Noy N., McGuinness D. L. *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, march 2001. 25 p.
- [31] Gruber Thomas R. A translation approach to portable ontology specifications. *Appeared in Knowledge Acquisition*, 1993, vol. 5(2), pp. 199-220.
- [32] Yu Liyang. *Introduction to Semantic Web and Semantic Web services*, CRC Press, 2007, 368 p.
- [33] John Davies, Rudi Studer, Paul Warren. *Semantic Web technologies: trends and research in ontology-based systems*. John Wiley & Sons, 2006, 327 p.
- [34] Dean Allemang, James Hendler. *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Elsevier, 2011, 384 p.
- [35] Leuf, Bo. *The Semantic Web: crafting infrastructure for agency*. John Wiley & Sons, 2006, 380 p.
- [36] Dieter Fensel, Holger Lausen, Axel Polleres, Jos de Bruijn, Michael Stollberg, Dumitru Roman, John Domingue. *Enabling semantic web services. The Web Service Modeling Ontology*. Springer Science & Business Media, 2007, 188 p.
- [37] *Semantic Web services architecture (SWSA)*. Available at: <http://www.daml.org/services/swsa/note/> (accessed 01.03.2019)

Сведения об авторах.



Куликовская Наталья А.
 Ассистент кафедры
 Компьютерных систем и
 сетей, Запорожского
 национального
 технического университета.
 Научные интересы:
 распределенные системы,
 семантический веб.
 E-mail:
natalya.gontar@gmail.com



Ильяшенко Матвей Б.
 к.т.н., доцент кафедры
 Компьютерных систем и
 сетей, Запорожского
 национального технического
 университета.
 Научные интересы: графы,
 машинное обучение,
 большие данные.
 E-mail:
matviy.ilyashenko@gmail.com



Тименко Артур В.
 Ассистент кафедры
 Компьютерных систем и
 сетей, Запорожского
 национального
 технического университета.
 Научные интересы: IoT;
 компьютерные сети.
 E-mail:
timenko.artur@gmail.com



Киричек Галина Г.
 к.т.н., доцент кафедры
 Компьютерных систем и
 сетей, Запорожского
 национального технического
 университета.
 Научные интересы:
 компьютерные сети, веб-
 технологии.
 E-mail: kirgal08@gmail.com