

УДК 004.89

И.Ю. Балашова, Е.А. Дзюба, Е.Н. Прошкина
**РАЗРАБОТКА ИНТЕГРИРОВАННЫХ СРЕДСТВ
ПРЕДСТАВЛЕНИЯ ЗНАНИЙ В ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЕ
ПОДДЕРЖКИ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНЫХ
ПРОДУКТОВ**

Пензенский государственный университет, Пенза, Россия

В статье обоснована актуальность автоматизации задачи выбора инструментальных средств в процессе управления жизненным циклом программного продукта. Выделены особенности данной задачи, позволяющие отнести ее к задачам искусственного интеллекта. Обосновано использование онтологического подхода при построении модели представления знаний для решения данной задачи в интеллектуальной информационной системе. Проведено функциональное моделирование процесса выбора инструментальных средств разработки программного продукта, и выделены основные функции интеллектуальной системы, реализующей данный процесс. Проведен анализ требований к построению и реализации интеллектуальной системы подбора инструментальных средств разработки программного продукта. Предложена архитектура системы, включающая базу знаний, сервер приложения и приложение пользователя. Разработана онтологическая модель базы знаний интеллектуальной системы, состоящая из множества классов предметной области и их экземпляров, множества отношений и аксиом онтологии. Построена объектная модель системы и ее подсистем. Описан формат представления знаний, с которыми работает сервер приложения интеллектуальной системы. Разработаны интегрированные средства представления знаний в интеллектуальной системе подбора инструментальных средств разработки программного продукта. Разработанная интеллектуальная система позволяет обеспечить сокращение материальных и временных затрат на поиск согласованного решения по выбору инструментальных средств разработки программного продукта.

Ключевые слова: представление знаний, онтологии, интеллектуальная система, инструментальные средства, программный продукт

Введение

Жизненный цикл программного продукта (ПП) определяется как период времени, который начинается с момента принятия решения о необходимости создания ПС и заканчивается в момент ее полного изъятия из эксплуатации [1]. В процессе управления жизненным циклом программного обеспечения возникает задача выбора инструментальных средств разработки, удовлетворяющих текущим и прогнозируемым требованиям при минимальном уровне затрат [2]. Обоснованный выбор связан с необходимостью проведения исследования потребностей и возможностей организации, специфики выполняемых проектов, требований заказчиков, обеспечения согласованности принятых решений. Задача выбора пакета инструментальных средств разработки ПП осложняется следующими аспектами [3]:

- исходная информация неполна, недостоверна и слабо структурирована;
- критерии оценки инструментальных средств имеют разные приоритеты;
- критерии оценки инструментальных средств разработки программного обеспечения по своей структуре имеют различный тип (существуют как количественные, так и качественные критерии оценки);
- набор критериев оценки может быть различным;
- все решения, принимаемые в течение жизненного цикла программного обеспечения, должны быть согласованными.
- при выборе пакета инструментальных средств могут быть выявлены альтернативы, которые в значительной мере, но не полностью, удовлетворяют поставленным требованиям и которые должны быть учтены.

Автоматизация задачи выбора инструментальных средств разработки ПП относится к задачам искусственного интеллекта. Ее решение в интеллектуальной информационной системе позволит обеспечить сокращение материальных и временных затрат на поиск согласованного решения по выбору инструментальных средств применительно к конкретным условиям разработки.

Материалы и методы

Выбор инструментальных средств разработки ПП осуществляется на основе множества критериев. Набор основных критериев приведен в Таблице 1.

Таблица 1 – Критерии выбора инструментальных средств разработки ПП

№ п/п	Критерий	Описание
1	Стоимость	Затраты на приобретение инструментального средства
2	Поддержка	Критерий определяет, поддерживается ли услуга обслуживания инструментального средства, кем и на каких условиях
3	Выполняемые функции	Набор функций, поддерживаемых инструментальным средством
4	Этапы жизненного цикла	Этапы жизненного цикла ПП, которым соответствует набор выполняемых инструментальным средством функций.
5	Поддерживаемые нотации, стандарты и методологии	Нотации, стандарты и методологии, поддерживаемые инструментальным средством на соответствующих этапах жизненного цикла ПП

№ п/п	Критерий	Описание
6	Интеграция с другими средствами разработки	Критерий определяет, с какими другими инструментальными средствами разработки ПП поддерживает интеграцию рассматриваемое инструментальное средство
7	Используемые платформы	Операционные системы, на базе которых может функционировать инструментальное средство разработки ПП
8	Программно-аппаратные требования	Набор технических и программных средств, необходимый для функционирования инструментального средства разработки ПП
9	Поддерживаемые форматы представления данных	Типы форматов представления данных, поддерживаемые инструментальным средством разработки ПП

Обоснованный выбор инструментальных средств для разработки ПП возможен и целесообразен на основе применения экспертных оценок степени приоритетности критериев применительно к конкретным условиям разработки [4].

Для решения задачи выбора инструментальных средств разработки по выделенным критериям интеллектуальная система должна обеспечивать возможность накопления знаний о средствах разработки ПП, их характеристиках, требованиях к применению и т.д. Знания в системе должны быть представлены таким образом, чтобы пользователь мог общаться с системой на привычном для него профессиональном языке и не испытывать при этом каких-либо сложностей и неудобств, связанных с взаимодействием с системой. Выбор той или иной модели представления знаний зависит от класса задач, решаемых интеллектуальной системой. Использование онтологического подхода имеет следующие преимущества [5]:

- масштабируемость относительно количества источников данных и размера онтологии;
- обеспечение возможности коллективной работы с базой знаний путем широкого использования полисинонимии и ретранслируемости;
- минимизация затрат, связанных с адаптацией приложения при расширении схемы базы знаний, и облегчение интеграции онтобазированных приложений благодаря наличию метауровня и универсальной стандартизированной модели представления знаний, не зависящей от конкретного приложения;
- минимизация влияния кодирования путем спецификации онтологии на уровне представления, а не символического кодирования.

Таким образом, в качестве метода представления знаний интеллектуальной системы подбора инструментальных средств

целесообразно использовать онтологический подход. На основе накопленных знаний интеллектуальная система должна обеспечивать принятие согласованных решений по выбору средств автоматизации разработки программного обеспечения на основании множества критериев.

Результаты

В общем случае процесс выбора инструментальных средств разработки ПП включает следующие этапы (Рисунок 1):

- анализ доступной информации;
- выявление альтернативных вариантов решения;
- выбор пакета инструментальных средств из числа альтернатив.

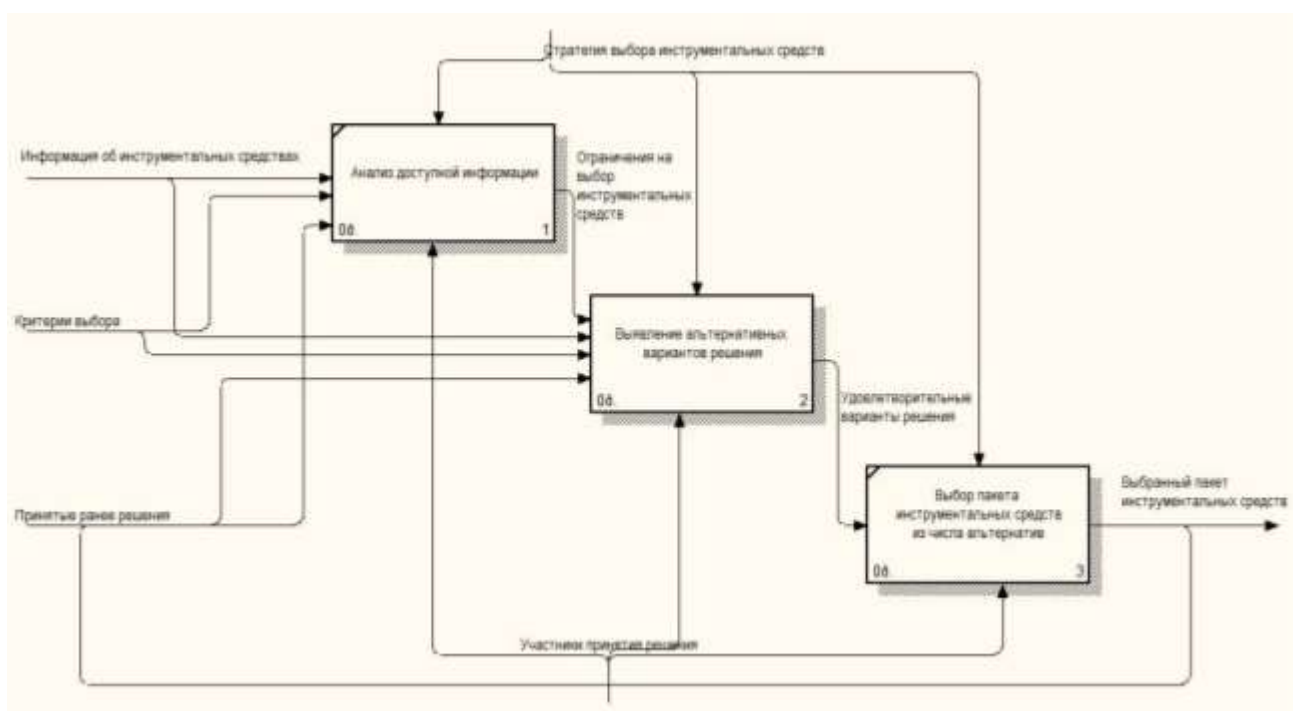


Рисунок 1 – IDEF0 диаграмма процесса выбора инструментальных средств разработки ПП

На первом этапе осуществляются анализ и структуризация информации, влияющей на выбор пакета инструментальных средств, выявляются ограничения, налагаемые на выбор пакета инструментальных средств. На следующем этапе формируются возможные пакеты инструментальных средств разработки ПП, которые в той или иной мере удовлетворяют выдвинутым требованиям. На третьем этапе производится выбор пакета инструментальных средств из числа альтернативных вариантов, выявленных на предыдущем этапе.

Проведен анализ требований и разработана архитектура интеллектуальной системы выбора инструментальных средств разработки ПП, включающая следующие уровни:

- база знаний;
- сервер приложения;
- приложение пользователя.

Уровень базы знаний включает в себя ряд онтологий различных уровней (Рисунок 2). Для поддержки ведения онтологии и решения на ее основе задачи выбора инструментальных средств разработки ПП определены операции добавления и модификации основных концептов онтологии.

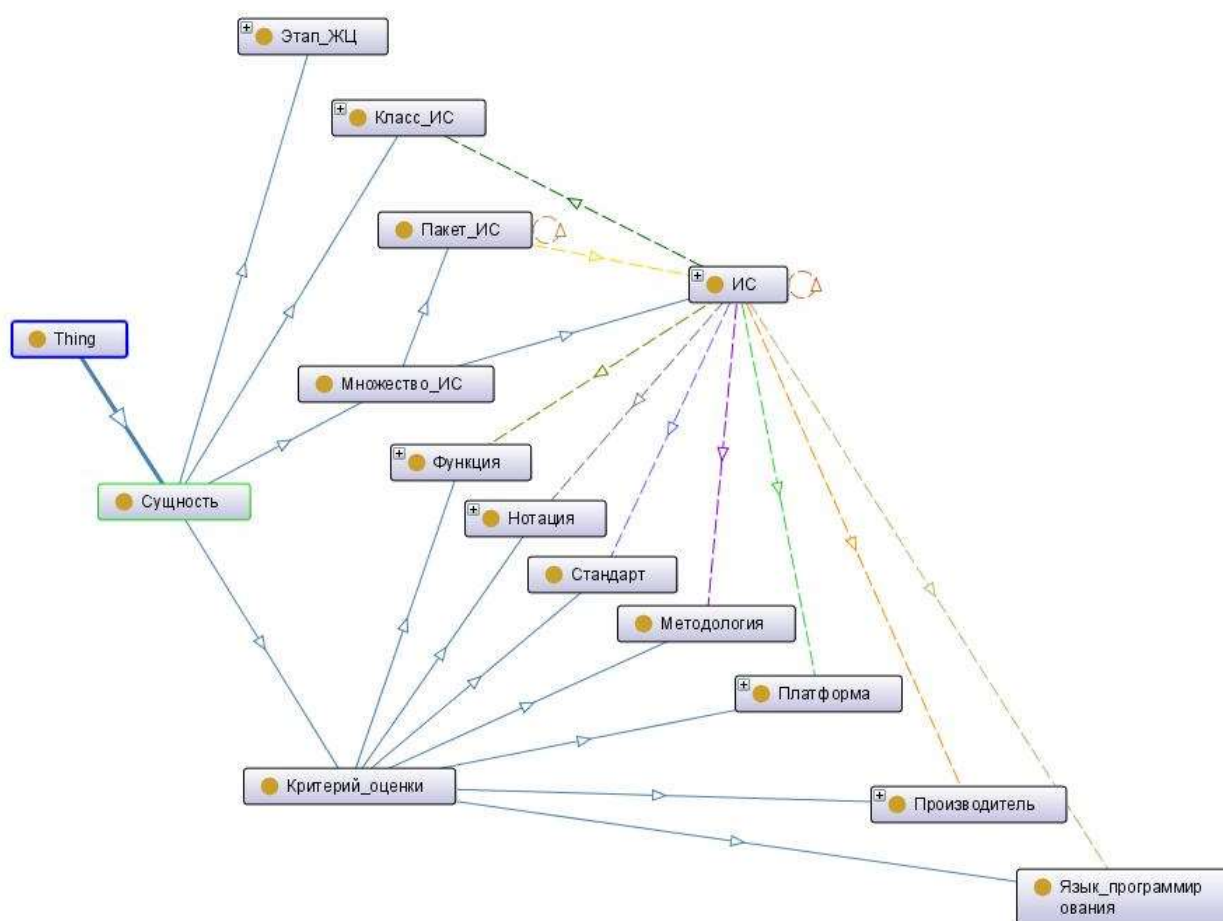


Рисунок 2 – Фрагмент онтографа базы знаний интеллектуальной информационной системы выбора инструментальных средств разработки ПП

Уровень сервера приложений содержит набор функций для работы с онтологией и является связующим звеном между пользовательским приложением и базой знаний.

Данный уровень включает в себя следующие подсистемы:

- подсистему работы с онтологией;
- подсистему ведения знаний;
- подсистему поиска решений.

Подсистема работы с онтологией содержит набор функций для чтения и записи аксиом и утверждений онтологии (Рисунок 3). Данная подсистема имеет непосредственный доступ к базе знаний через соответствующие средства представления знаний.

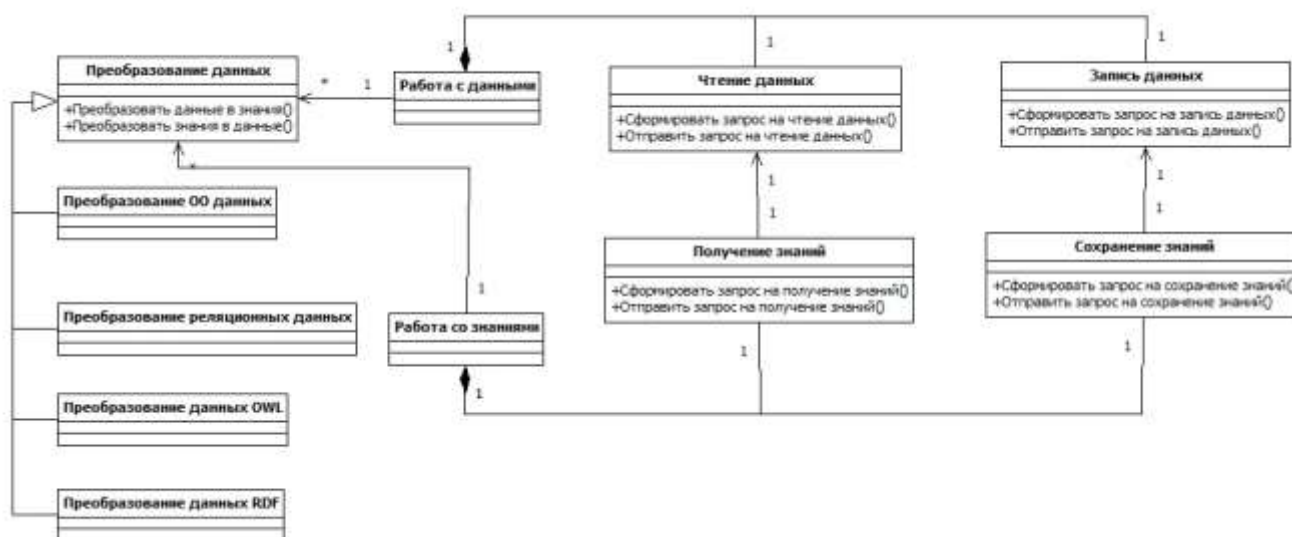


Рисунок 3 – Объектная модель подсистемы работы с онтологией

Подсистема ведения знаний содержит набор функций для добавления, модификации и удаления знаний онтологии (Рисунок 4). Доступ к базе знаний данная подсистема получает, используя соответствующие средства подсистемы работы с онтологией сервера приложения. В рамках подсистемы ведения знаний выделены два набора функций для двух различных типов конечных пользователей: инженера знаний и эксперта. Доступ к одному из этих наборов функций уровень приложения пользователя получает через соответствующий интерфейс.

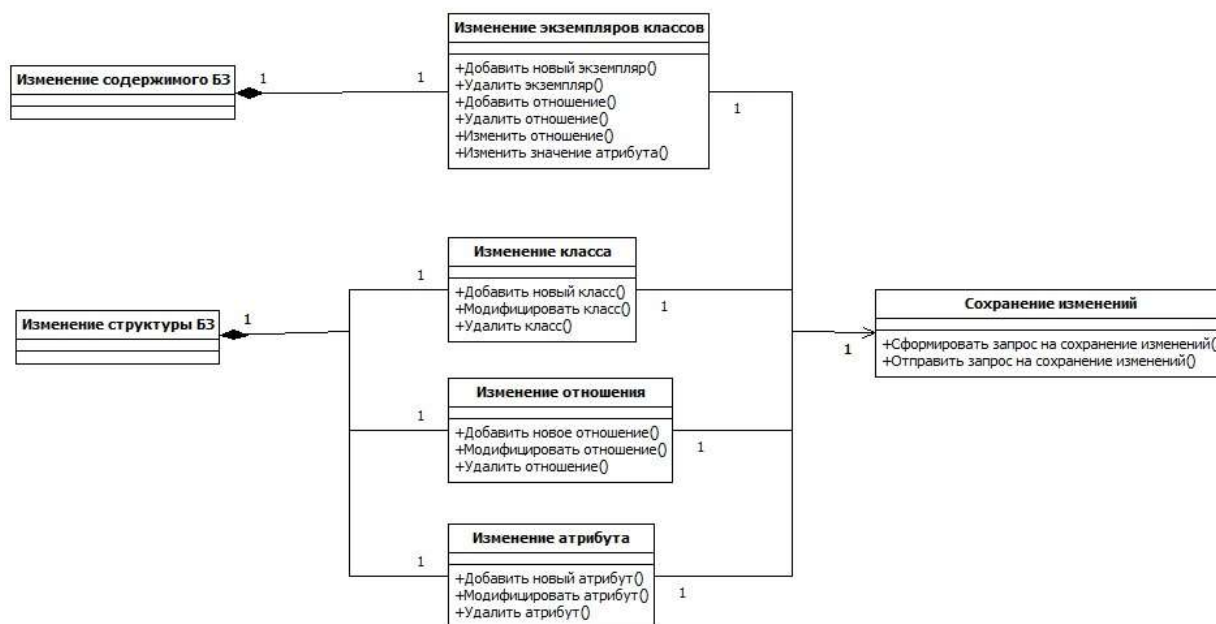


Рисунок 4 – Объектная модель подсистемы ведения знаний

Подсистема поиска решений содержит набор функций для осуществления поиска и выбора инструментальных средств разработки программного обеспечения по разработанному алгоритму, основанному на методе оценки подобия объектов в интеллектуальном пространстве знаний. (Рисунок 5). Данная подсистема, так же как и подсистема ведения знаний, получает доступ к знаниям и фактам базы знаний при помощи соответствующих средств подсистемы работы с онтологией.

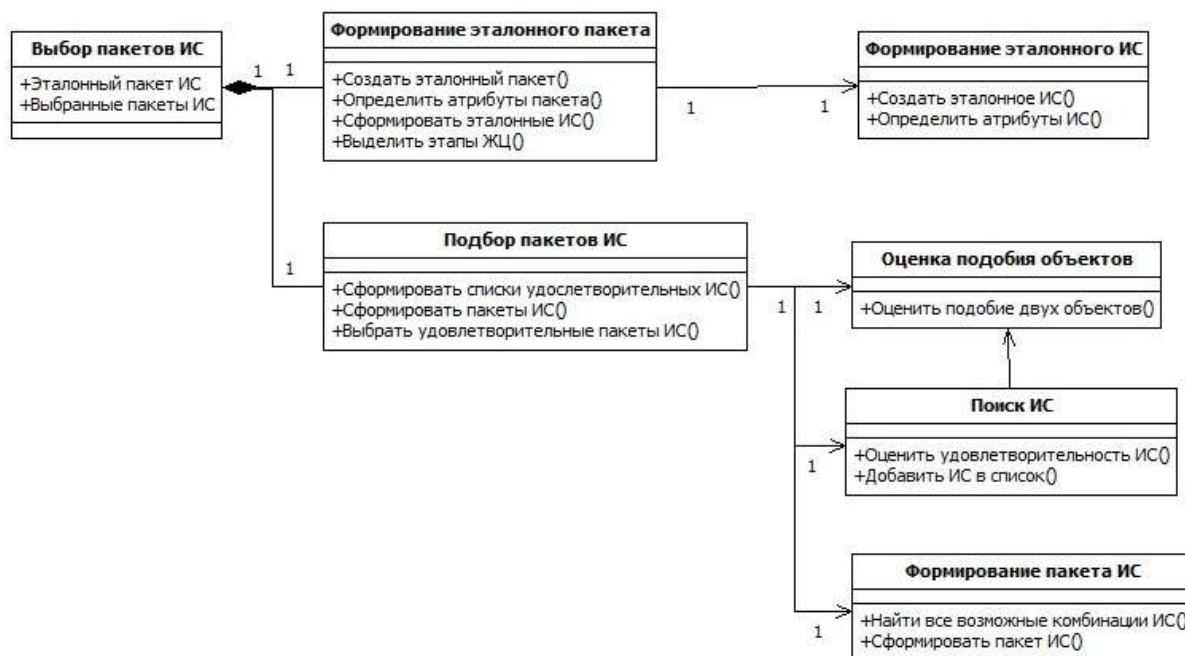


Рисунок 5 – Объектная модель подсистемы поиска решений

Все три подсистемы уровня сервера приложения интеллектуальной системы подбора инструментальных средств разработки ПП работают со знаниями, имеющими определенное представление (Рисунок 6).



Рисунок 6 – Объектная модель представления знаний на уровне сервера приложений интеллектуальной системы

Уровень приложения пользователя – это верхний уровень интеллектуальной системы, с которым непосредственно взаимодействует конечный пользователь. Уровень приложения пользователя включает в свой состав следующие компоненты:

- представление знаний онтологии;
- ведение знаний;
- поиск решений.

Каждый из компонентов уровня приложения пользователя связан через соответствующий интерфейс с одной из подсистем уровня сервера приложения и имеет доступ к определенному набору функций.

Компонент для представления знаний онтологии включает средства отображения всей структуры онтологии и хранящихся в ней знаний в понятной для конечного пользователя форме (Рисунок 7). Доступ к этой функции имеет любой пользователь.

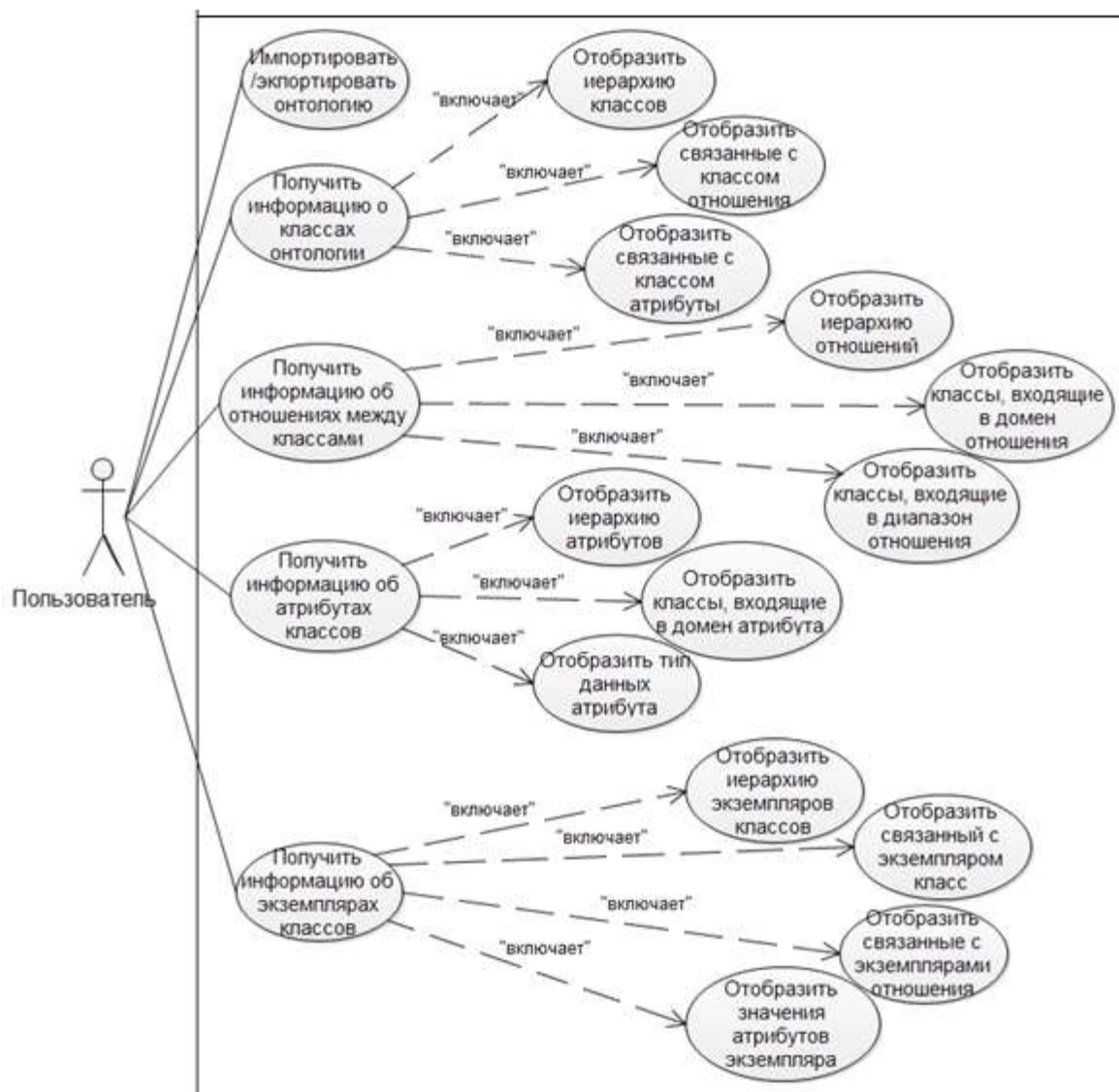


Рисунок 7 – Диаграмма вариантов использования для компонента представления знаний онтологии

Компонент для ведения знаний включает средства для предоставления возможности конечному пользователю осуществлять операции со знаниями онтологии. Этот компонент доступен в двух режимах работы: в режиме инженера знаний; в режиме эксперта. Для каждого режима определен свой функциональный набор, исходя из прав пользователя.

Компонент для поиска решений включает средства, обеспечивающие следующие возможности для конечного пользователя:

- определение набора критериев оценки инструментальных средств разработки ПП и их весовых значений;
- запуск процесса поиска инструментальных средств разработки ПП;

- отображение найденных вариантов решений;
- сохранение выбранных пакетов инструментальных средств.

Компонент для поиска решений доступен для любого конечного пользователя.

Заключение

Разработанная интеллектуальная система позволяет обеспечить

- ведение онтологий для накопления и обработки знаний и фактов, необходимых для принятия управленческих решений о выборе инструментальных средств разработки ПП;
- принятие согласованного решения о выборе инструментальных средств разработки ПП на основе накопленных знаний и фактов;
- сокращение материальных и временных затрат на поиск согласованного решения по выбору инструментальных средств разработки ПП.

Построенная онтологическая модель и алгоритм принятия решений могут быть расширены и применены для задач принятия любых управленческих решений на этапах жизненного цикла программных продуктов.

ЛИТЕРАТУРА

1. Назаров С.В. Архитектура и проектирование программных систем: монография. – М.: ИНФРА-М, 2013. – 413 с.
2. Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработки программного обеспечения: учебное пособие / под ред. Л.Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М., 2008. – 400 с.
3. Дзюба Е.А. Сравнительный анализ современных средств поддержки жизненного цикла информационных систем / Дзюба Е.А., Шибанов С.В., Герасина А.И. // Труды международного симпозиума «Надежность и качество' 2012». – т. 1. – Пенза, 2012. – с. 420-427.
4. Клименко И.В. Метод выбора инструментального средства для разработки программного обеспечения // Известия высших учебных заведений. Северо-Кавказский регион. Технические науки. – 2011. – №4. – С. 6-9.
5. Антониоу Г. Семантический веб / Г. Антониоу, П. Грос, Ф. ван Хармелен, Р. Хоекстра. – М: ДМК Пресс, 2016. – 240 с.

I.Y. Balashova, E.A. Dzyuba, E.N. Proshkina
**DEVELOPMENT OF INTEGRATED MEANS OF
REPRESENTATION OF KNOWLEDGE IN THE INTELLECTUAL
SYSTEM OF SUPPORT OF THE SOFTWARE LIFE CYCLE**

Penza State University, Penza, Russia

The article proves the relevance of automation of the task of selecting tools in the process of managing the life cycle of a software product. The peculiarities of this problem are singled out, which allow attributing it to the problems of artificial intelligence. The use of the ontological approach in the construction of the knowledge representation model for solving this problem in an intelligent information system is substantiated. Functional modeling of the process of selecting software development tools was carried out, and the main functions of the intellectual system implementing this process were identified. The analysis of the requirements for the construction and implementation of an intellectual system for selecting software development tools was carried out. A system architecture is proposed that includes a knowledge base, an application server, and a user application. An ontological model of knowledge base of an intellectual system is developed, consisting of many classes of domain and their instances, set of relations and axioms of ontology. The object model of the system and its subsystems is constructed. The format of knowledge representation with which the smart system application server works is described. Integrated means of representation of knowledge in the intellectual system of selection of tools for software product development have been developed. The developed intellectual system allows to provide reduction of material and time expenses for search of the coordinated decision on a choice of tools of development of a software product.

Keywords: knowledge representation, ontologies, intellectual system, software development tools, software product

REFERENCES

1. Nazarov S.V. Architecture and design of software systems: monograph. - Moscow: INFRA-M, 2013. - 413 p.
2. Gagarina LG, Kokoreva EV, Visnadul BD Technology for the development of software: a tutorial / ed. L.G. Gagarin. - Moscow: ID FORUM: INFRA-M., 2008. - 400 p.
3. Dzyuba E.A. Comparative analysis of modern means of supporting the life cycle of information systems / Dzyuba E.A, Shibanov S.V, Gerasina A.I // Proceedings of the International Symposium "Reliability and Quality '2012". - t. 1. - Penza, 2012. - p. 420-427.
4. Klymenko I.V. Method of choosing an instrument for software development // Izvestiya Vysshikh Uchebnykh Zavedenii. North-Caucasian region. Technical science. - 2011. - №4. - P. 6-9.
5. Antonioou G. Semantic Web / G. Antonioou, P. Gros, F. van Harmelen, R. Hoekstra. - M: DMK Press, 2016. - 240 p.