

МЕТОДЫ И ТЕХНОЛОГИИ ПРИНЯТИЯ РЕШЕНИЙ

УДК 004.89

DOI: 10.18287/2223-9537-2020-10-1-121-140

Поддержка принятия решений при управлении программными проектами на основе нечёткой онтологии

В.В. Антонов¹, О.В. Бармина², Н.О. Никулина¹

¹Уфимский государственный авиационный технический университет, Уфа, Россия

²ООО Нордиджи, Санкт-Петербург, Россия

Аннотация

Рассмотрены вопросы управления знаниями при выполнении программных проектов и организации поддержки принятия решений для их участников. Актуальность исследований обоснована необходимостью снизить степень неопределённости при принятии решений и подтверждена статистикой выполнения проектов. Поддержку принятия решений предложено организовать на основе анализа прецедентов ранее происходивших проблемных ситуаций и поиска оптимального решения в текущей ситуации, для чего предложено использовать методы инженерии знаний, в частности, онтологического анализа предметной области. Разработана методика формирования нечёткой онтологии поддержки принятия решений при управлении программными проектами. Новизна модели поиска прецедентов заключается в сочетании различных механизмов логического вывода при принятии решений на базе всего комплекса знаний о предметной области. Большой объём информации не позволяет использовать простые поисковые запросы для оперативного поиска необходимой информации, созданной в процессе выполнения проекта. Предложен алгоритм формирования поискового запроса, учитывающего нечёткие свойства, нечёткие таксономические и ассоциативные отношения между классами объектов в онтологии. Эффективность предложенных решений подтверждена экспериментом на прототипе интеллектуальной системы поддержки принятия решений.

Ключевые слова: управление проектами, нечёткая онтология, управление знаниями, поисковый запрос, поддержка принятия решений, прецедент проблемной ситуации.

Цитирование: Антонов, В.В. Поддержка принятия решений при управлении программными проектами на основе нечёткой онтологии / В.В. Антонов, О.В. Бармина, Н.О. Никулина // Онтология проектирования. – 2020. – Т.10, №1(35). – С.121-140. – DOI: 10.18287/2223-9537-2020-10-1-121-140.

Введение

Одним из направлений развития проектно-ориентированных ИТ-компаний является сосредоточение на извлечении уроков из успешных отраслевых проектов [1, 2]. Так, в Руководстве к своду знаний по управлению проектами активами организации, использование которых оказывает благоприятное влияние на достижение целей проекта, считаются: политики, правила и процедуры, а также корпоративная база знаний (БЗ) [3]. При этом «репозитории знаний организации, предназначенные для хранения и извлечения информации, включают в себя, среди прочего, репозитории данных по управлению проблемами, содержащие сведения о статусе проблем, информацию о контроле, данные о разрешении проблем, а также результаты предпринятых действий» [3, с.41]. Совокупность этих данных можно назвать прецеден-

тами проблемных ситуаций (ПС) [4]. Такая информация должна быть зафиксирована и доступна всем участникам проекта в рамках их компетенции и полномочий.

Программные проекты (ПП) обладают рядом особенностей, среди которых – высокая степень уникальности конечного продукта, многообразие методов и средств, применяемых при разработке программного обеспечения (ПО), затрудняющее их выбор для реализации конкретного проекта, плохо формализуемые взаимодействия между участниками проекта, большая доля творческой составляющей в деятельности человека. Всё это диктует необходимость применения инструментов, позволяющих не только управлять разработкой ПО (своевременно отслеживать статус решаемых задач, процент их выполнения и объём затраченных ресурсов, вести журнал выявленных и исправленных ошибок), но и накапливать информацию для устранения ПС. Возникновение ПС в ПП часто связывают с [5, 6]:

- неточным определением сроков и стоимости проекта – 62%;
- нарушением взаимодействия между различными участниками проекта – 47%;
- отсутствием интеграции между различными системами, используемыми при управлении ПП (системами календарно-ресурсного планирования, системами отслеживания ошибок, системами бюджетирования) – 38%;
- отсутствием единого централизованного источника информации о проекте – 35%;
- неудовлетворительным управлением ресурсами – 31%;
- отсутствием видимости результатов в ещё не завершённых задачах – 21%.

Возможность использовать знания и опыт, накопленные при управлении ПП в различных ситуациях, позволяет быстрее и более качественно решать возникающие проблемы. Однако всего лишь 14% компаний применяют этот подход на практике, что связано со сложностью представления и формализации знаний в этой области, высокой неопределённостью в принятии решений [1, 5, 7].

1 Нечёткая онтология как модель представления знаний

Неопределённость и неточность могут рассматриваться как две противоположные точки зрения на неполноту информации. Информация, как правило, выражается в виде логического высказывания, содержащего предикаты, квантификаторы. Под БЗ понимается множество сведений, имеющих у субъекта (группы субъектов) или содержащихся в информационной системе и относящихся к одной и той же предметной области (ПрО). Любое высказывание может рассматриваться как утверждение, относящееся к появлению некоторого события. Известны три эквивалентных способа анализа множества данных в зависимости от того, делается ли акцент на структуре (логическая точка зрения), содержании (теоретико-множественная точка зрения) этой информации или на её отношении к действительным фактам (событийная точка зрения) [8].

Информационную единицу знания можно представить в виде четвёрки:

$$(1) \quad K = \langle O, A, V, C \rangle,$$

где K – информационная единица знания (*knowledge*), O – объект (*object*), A – признак (*attribute*), V – значение (*value*), C – уверенность (*certainty*).

Под признаком понимается функция, задающая значение (множество значений) объекта или предмета, название которого упоминается в информационной единице знания. Значение соответствует некоторому предикату. Уверенность есть показатель надёжности информационной единицы знания. Принятие решения в большинстве случаев заключается в генерации возможных альтернативных решений, их оценке и выборе наилучшего варианта, иногда с привлечением экспертов или других способов информационной поддержки.

При выборе варианта приходится учитывать большое число неопределённых и противоречивых факторов [9, 10]. Неопределённость является неотъемлемой частью процессов принятия решений и может быть связана с:

- неполнотой знаний о проблеме, по которой должно быть принято решение;
- невозможностью полного учёта реакции окружающей среды на принимаемые решения;
- непониманием лицом, принимающим решение (ЛПР), своих целей.

Противоречивость возникает из-за неоднозначности оценки ситуаций, ошибки в выборе приоритетов, что осложняет принятие решений. Исследования показывают, что ЛПР без дополнительной аналитической поддержки, как правило, использует упрощённые, а иногда и противоречивые правила выбора решения [9]. Кроме того, наличие разнородных источников информации также затрудняет процесс принятия решений. Основная проблема, возникающая при попытках повторного использования знаний и передового опыта, заключается в сложности выбора успешных решений, опыт которых применим к конкретной решаемой задаче. Эта проблема может быть решена путём систематизации знаний и моделирования ПрО с позиций методов, учитывающих нечёткость описаний модели исследуемого объекта [11-13]. Снижению неопределённости способствует систематизация знаний о деятельности ЛПР, повышение интеграции источников знаний и создание единого информационного пространства, использование онтологий [14].

Онтологию можно представить в виде неструктурированного описания на естественном языке (*not formal ontology*), в виде полуструктурированного описания как графической схемы (*semi-formal ontology*), в виде структурированного описания на языке *OWL*, *RDF* (*formal ontology*). Онтология предназначена для решения проблем семантического различия информации, поступающей из различных источников, предоставления возможностей автоматического вывода знаний и уменьшения неопределённости знаний в ПрО. Примеры применения онтологий в различных ПрО приведены в [14-16]. Онтологическая модель описания прецедентов ПС, встречающихся в ПП, подробно описана в [17]. Чаще всего онтологии используют для уменьшения неопределённости легко структурируемых знаний, примерами могут служить таксономии с чёткими концептами, а также знания, основанные на правилах. В случае трудно структурируемых знаний применяются онтологии, основанные на нечёткой логике, теории вероятностей и теории возможностей [8, 18].

Нечёткие знания характерны для многих ПрО, в том числе таких как интеллектуальный анализ информации, машинное обучение, которые оперируют неточными или неполными знаниями. Нечёткая онтология может быть представлена в виде:

$$(2) \quad O = \langle C, I, P, T, N, A \rangle,$$

где C – множество нечётких понятий (классов), каждое понятие является нечётким множеством в области экземпляров, такое, что $C : I \rightarrow [0, 1]$; I – экземпляры (элементы классов). Множество объектов нечёткой онтологии определяется как $E = CUI$; P – множество нечётких свойств класса C ; T – множество нечётких таксономических отношений, такое что $T : E \rightarrow [0, 1]$; N – множество нечётких не таксономических ассоциативных отношений $N : E \rightarrow [0, 1]$; A – множество нечётких аксиом.

Нечёткая онтология может быть создана в редакторе онтологий *Protege 5.2.0* [19] с использованием плагина *FuzzyOWL*, основные элементы которого представлены в таблице 1.

Синтаксис *FuzzyOWL* предполагает три формата алфавита: для представления нечётких классов (*fuzzy concepts*), нечётких отношений (*fuzzy roles*) и нечётких объектов классов онтологии (*fuzzy individuals*), которые определяются свойством аннотации *fuzzyLabel*, хранящей параметры и значения функции принадлежности, соотносимые с конкретным классом, объектом, отношением внутри заданного нечёткого множества данных [18, 20].

Таблица 1– Основные элементы *FuzzyOWL*

Элемент <i>FuzzyOWL</i>	Формат	Пример представления в нечёткой онтологии <i>FuzzyOWL</i> в виде <i>OWL/XML</i> -разметки
<i>Fuzzy ontologies</i> - множество операторов нечётких множеств	<pre><FUZZY_LOGIC> := "zadeh" <FUZZY_LOGIC> := "lukasiewicz"</pre>	<pre><Annotation> <AnnotationProperty IRI="#fuzzyLabel"/> <Literal datatypeIRI="http://www.w3.org/1999/02/22- rdf-syntax-ns#PlainLiteral"><fuzzyOwl2 fuzzyType="ontology";>; <FuzzyLogic logic="zadeh"; />; </fuzzyOwl2>; </Literal> </Annotation></pre>
<i>Fuzzy datatypes</i> - множество функций степени принадлежности	<pre><DATATYPE>:= <Datatype type="leftshoulder" a="<DOUBLE>" b="<DOUBLE>" /> <Datatype type="rightshoulder" a="<DOUBLE>" b="<DOUBLE>" /> <Datatype type="triangular" a="<DOUBLE>" b="<DOUBLE>" c="<DOUBLE>" /> <Datatype type="trapezoidal" a="<DOUBLE>" b="<DOUBLE>" c="<DOUBLE>" d="<DOUBLE>" /></pre>	<pre><AnnotationAssertion > <AnnotationProperty IRI=' #fuzzyLabel' /> <IRI>#YoungAge</IRI> <Literal datatypeIRI=' &rdf;PlainLiteral' > <fuzzyOwl2 fuzzyType="datatype"> <Datatype type="leftshoulder" a="10" b="30" /> </fuzzyOwl2> </Literal> </AnnotationAssertion></pre>
<i>Fuzzy concepts</i> - множество нечётких классов	<pre><WEIGHTED_CONCEPT> := <Concept type="weighted" val- ue="<DOUBLE>" base="<STRING>" /></pre>	<pre><AnnotationAssertion > <AnnotationProperty IRI=' #fuzzyLabel' /> <IRI>#Weight0.8C</IRI> <Literal datatypeIRI=' &rdf;PlainLiteral' > <fuzzyOwl2 fuzzyType="concept"> <Concept type="weighted" value="0.8" base="C" /> </fuzzyOwl2> </Literal> </AnnotationAssertion></pre>
<i>Fuzzy axioms</i> - множество нечётких аксиом	<pre><FUZZY AXIOM> := <fuzzyType="axiom"> <Degree value="<DOUBLE>" /></pre>	<pre><ClassAssertion > <Class IRI=' #Tall' /> <NamedIndividual IRI=' #paul' /> <Annotation> <AnnotationProperty IRI=' #fuzzyLabel' /> <Literal datatypeIRI=' &rdf;PlainLiteral' > <fuzzyOwl2 fuzzyType="axiom"> <Degree value="0.5" /> </fuzzyOwl2> </Literal> </Annotation> </ClassAssertion></pre>
<i>Fuzzy roles</i> - множество нечётких отношений	<pre><FUZZY ROLE> := <fuzzyType="role"> <Role type="modified" modifier = "<modifier>" base="<STRING>" /></pre>	<pre><AnnotationAssertion> <AnnotationProperty IRI=" #fuzzyLabel" /> <AbbreviatedRI> generations:hasProject </AbbreviatedIRI> <Literal datatypeIRI="&rdf;PlainLiteral"> <fuzzyOwl2 fuzzyType="role"> <Role type="modified" modifier="low"; base="topObjectProperty" /> </fuzzyOwl2> </Literal> </AnnotationAssertion></pre>
<i>Fuzzy individuals</i> - множество нечётких экземпляров	<pre><FUZZY INDIVIDUALS> := <Concept type="nominal" val- ue="<DOUBLE>" individual="<STRING>" /></pre>	<pre><AnnotationAssertion> <AnnotationProperty IRI=" #fuzzyLabel" /> <IRI>#Project</IRI> <Literal datatypeIRI="&rdf;PlainLiteral"> <fuzzyOwl2 fuzzyType="concept"> <Concept type="nominal" value="0.1" individ- ual="Внедрение ИС" /> </fuzzyOwl2></Literal> </AnnotationAssertion></pre>

Онтологические модели, в том числе нечёткие, позволяют выявлять взаимосвязи в процессах управления проектом. Нечёткая онтология ПрО разработана таким образом, чтобы отобразить множество классов ПС, связанных с решением различных задач, возникающих во время выполнения основных процессов управления ПП, а также структурировать БЗ. Разра-

ботка нечёткой онтологии соответствует принципам *FuzzyOWL* и включает последовательность шагов, представленных на рисунке 1.

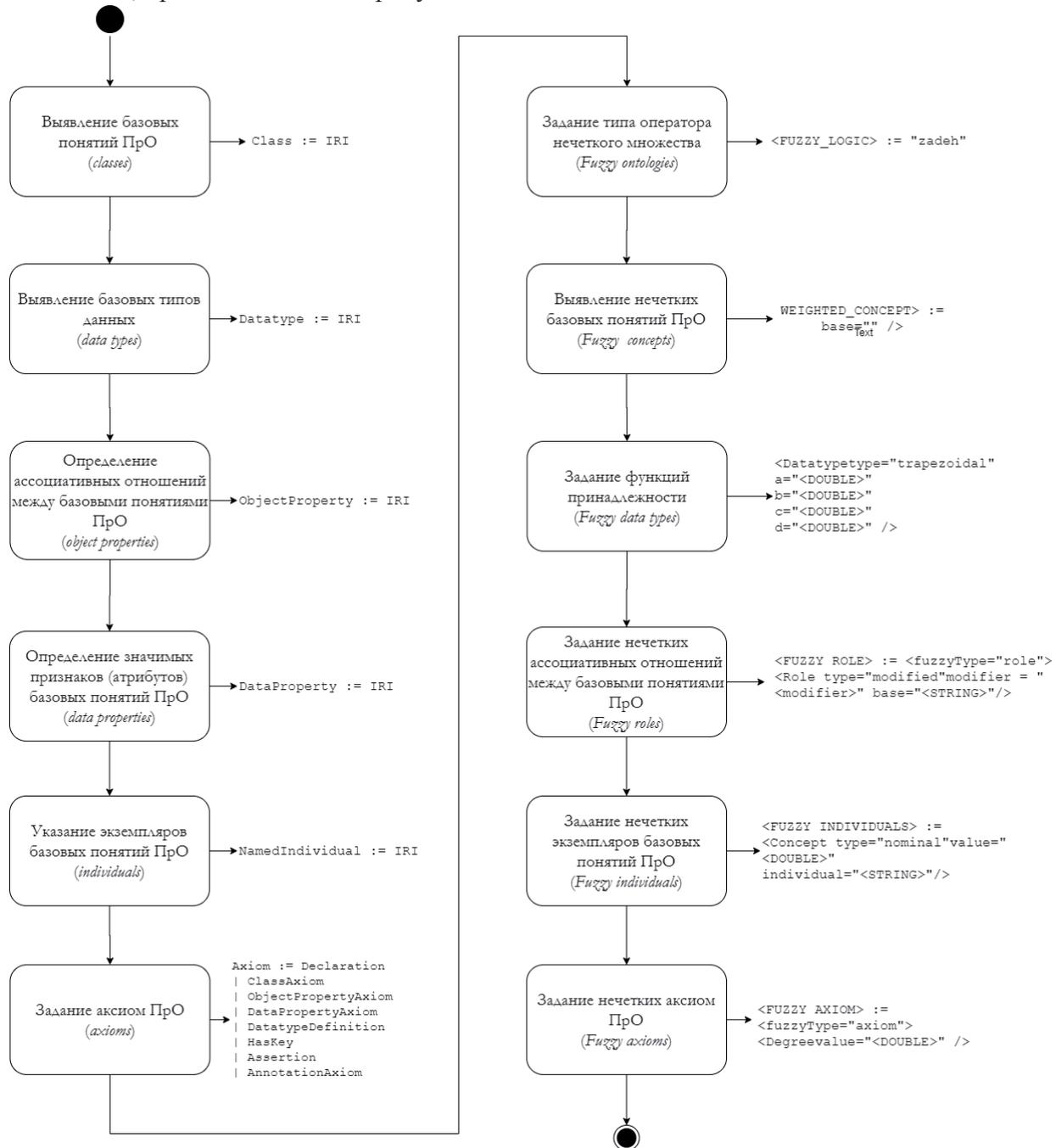


Рисунок 1 – Последовательность построения нечёткой онтологии

После разработки онтологии для обмена и использования знаниями необходимо обеспечить поддержание в актуальном состоянии базы (репозитория) прецедентов. Данная деятельность представляет собой замкнутый цикл, состоящий из шести этапов:

- 1) поиск прецедентов под возникшую проблему;
- 2) ранжирование похожих прецедентов;
- 3) выбор наиболее похожих прецедентов;

- 4) повторное использование информации и знаний из найденного прецедента для решения возникшей проблемы;
- 5) оценка эффективности предложенного решения проблемы;
- 6) сохранение вновь возникшего успешного прецедента в БЗ для решения подобных проблем в будущем.

2 Алгоритм поиска ближайших прецедентов в нечёткой онтологии

Под прецедентом понимается описание ПС в совокупности с подробным указанием действий, предпринимаемых в данной ситуации для решения данной проблемы. Обобщённая модель прецедента имеет следующий вид [4, 21]:

$$(3) \quad \textit{Case: Problem} \rightarrow \textit{Solution; Result},$$

где *Case* – прецедент; *Problem* – ПС, описывающая состояние процесса управления ПП, когда произошел прецедент; *Solution* – решение этой проблемы; *Result* – результат, который описывает состояние процесса управления ПП после произошедшей ПС.

Подробное описание признаков прецедента ПС, возникающих при управлении ПП, приведено в [17]. Одной из наиболее часто встречающихся проблем в проектно-ориентированных компаниях является распределение ресурсов (в том числе трудовых) между параллельно выполняющимися ПП, а успешность выполнения отдельного проекта может зависеть от того, насколько удачным окажется выбор участников команды проекта. Поэтому процесс поиска прецедента может быть ориентирован на достижение различных целей, например:

- найти такой прецедент, чтобы распределение ресурсов между проектами оказалось наиболее надёжным, обеспечивающим достижение конечной цели проекта;
- найти прецедент с таким распределением ресурсов в проекте, которое обеспечивает минимальное время исполнения;
- найти прецедент, для которого время получения нового решения окажется минимальным (минимум модификаций);
- найти прецедент, отражающий наиболее современный (поздний) опыт.

Онтологическая модель ПрО может быть представлена в виде графа. Такое представление семантической модели определяет способ построения запросов к ней, реализованный в языке *SPARQL*. Существующие алгоритмы поиска по прецедентам [22] не поддерживают нечёткие онтологии, поэтому разработан алгоритм поиска, позволяющий учитывать нечёткие свойства, нечёткие таксономические и ассоциативные отношения.

Поисковый запрос Q_i определяется как совокупность множества типов атрибутов t_i , наименований атрибутов n_i , значений атрибутов v_i и типов операций сравнения o_i :

$$(4) \quad Q_i = \langle t_i, n_i, v_i, o_i \rangle$$

Так как в онтологии задана модель данных прецедента ПС, то поисковый запрос Q_i содержит конечное множество кортежей, соответствующих модели данных (рисунок 2).

Поисковый запрос пользователя к БЗ, как правило, не полностью отражает ПС, так как пользователь может не знать всех терминов и структуры данных. Использование меры близости для оценки найденных прецедентов в БЗ позволяет расширять запросы и ранжировать результаты поиска. При осуществлении поиска выполняется сравнение заданных параметров в поисковом запросе со значениями прецедентов – экземпляров в онтологии. На основании таксономической близости поисковый запрос расширяется схожими классами ПС, так что предварительная кластеризация прецедентов становится необязательной.

Схема алгоритма поиска прецедента в БЗ в случае использования нечёткой онтологии приведена на рисунке 3.

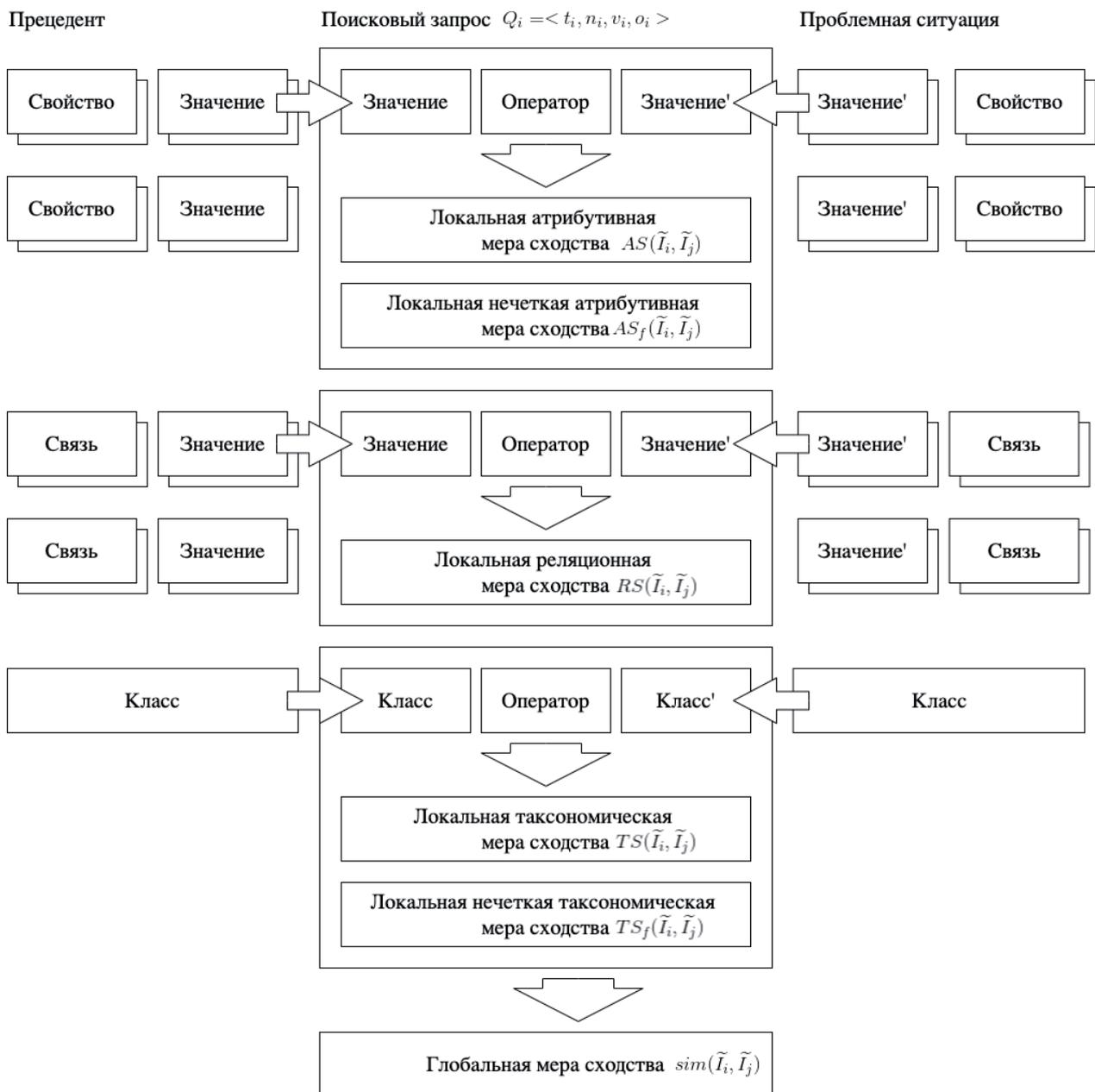


Рисунок 2 – Модель данных поискового запроса Q_i

Определение меры сходства ПС и прецедента в БЗ состоит из следующих шагов.

Шаг 1. Парное сравнение атрибутов прецедента и ПС на основе выбранного оператора сравнения для определения атрибутивной близости $AS(\tilde{I}_i, \tilde{I}_j)$. Атрибутивной мерой сходства $AS(\tilde{I}_i, \tilde{I}_j)$, где $\mu_{\tilde{I}_i}(x) = 1$ и $\mu_{\tilde{I}_j}(x) = 1$, считается определение подобия признаков описания ПС, где, в зависимости от типа данных атрибута, применяется соответствующий оператор сравнения. Так, для строковых типов может использоваться мера сходства – расстояние Левенштейна [23] или строгое посимвольное сравнение.

Шаг 2. Парное сравнение прецедента и ПС для определения нечёткой атрибутивной близости $AS_f(\tilde{I}_i, \tilde{I}_j)$. В качестве нечёткой меры сходства используется мера сходства Заде для нечётких признаков $\tilde{I}_i(x_k)$ и $\tilde{I}_j(x_k)$, $x \in E$ [24]:

$$(5) \quad AS_f(\tilde{I}_i, \tilde{I}_j) = \min \left(\max \left(1 - \mu_{\tilde{I}_i}(x_k), \mu_{\tilde{I}_j}(x_k) \right), \max \left(\mu_{\tilde{I}_i}(x_k), 1 - \mu_{\tilde{I}_j}(x_k) \right) \right), k = \overline{1, n}$$

где $\mu_{\tilde{I}_i}(x_k)$ – значение функции принадлежности нечёткого признака $\tilde{I}_i(x_k)$,
 $\mu_{\tilde{I}_j}(x_k)$ – значение функции принадлежности нечёткого признака $\tilde{I}_j(x_k)$

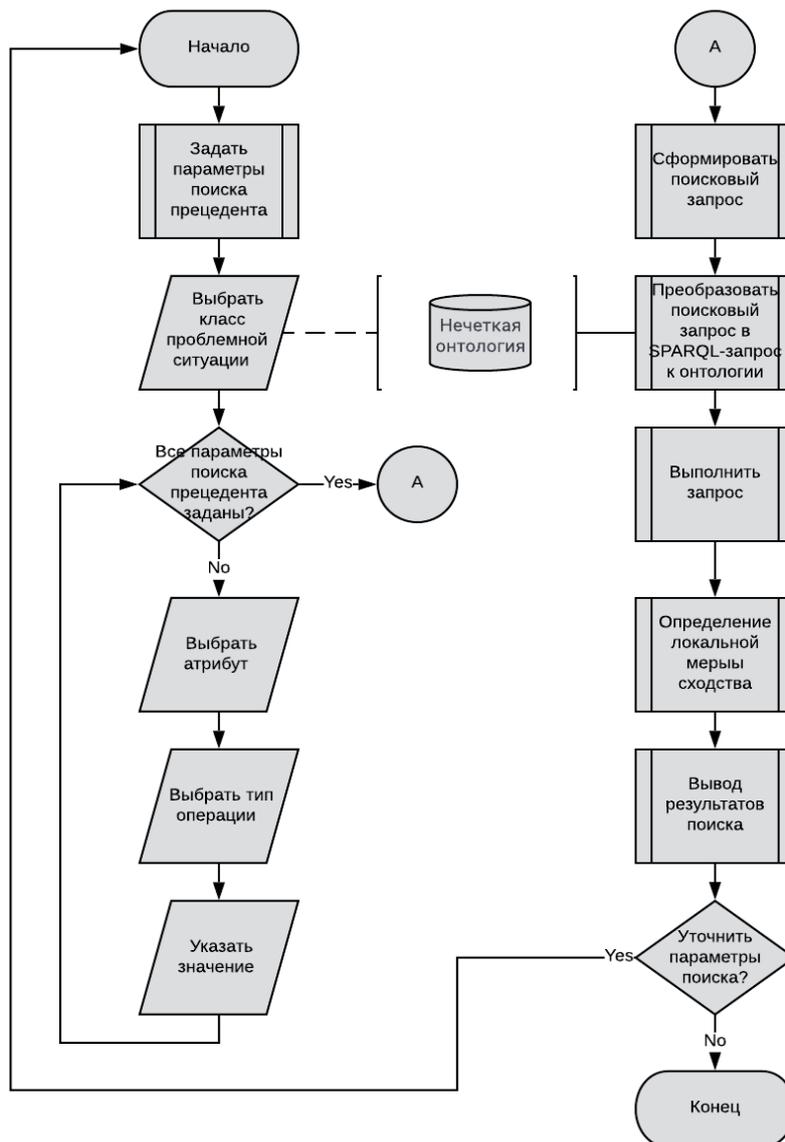


Рисунок 3 – Схема алгоритма поиска прецедента в нечёткой онтологии поддержки принятия решений при управлении программными проектами

Термы лингвистической переменной и тип функции принадлежности задаются в виде аннотации *fuzzyLabel* к пользовательскому типу данных в онтологии (рисунок 4). Описание работы с редактором онтологий *Protege 5.2.0* приведено в [19]. Определение нечёткого типа данных *fuzzy data type* в редакторе онтологий *Protege 5.2.0* приведено на рисунке 4, а определение типа данных для свойства *data property* на рисунке 5. *Domain range* свойства *data property* содержит значения термов лингвистической переменной (5).

В таблице 2 приведён пример определения нечёткой атрибутивной близости для двух проектов C_1 и C_2 к проекту P , в котором обнаружена ПС.

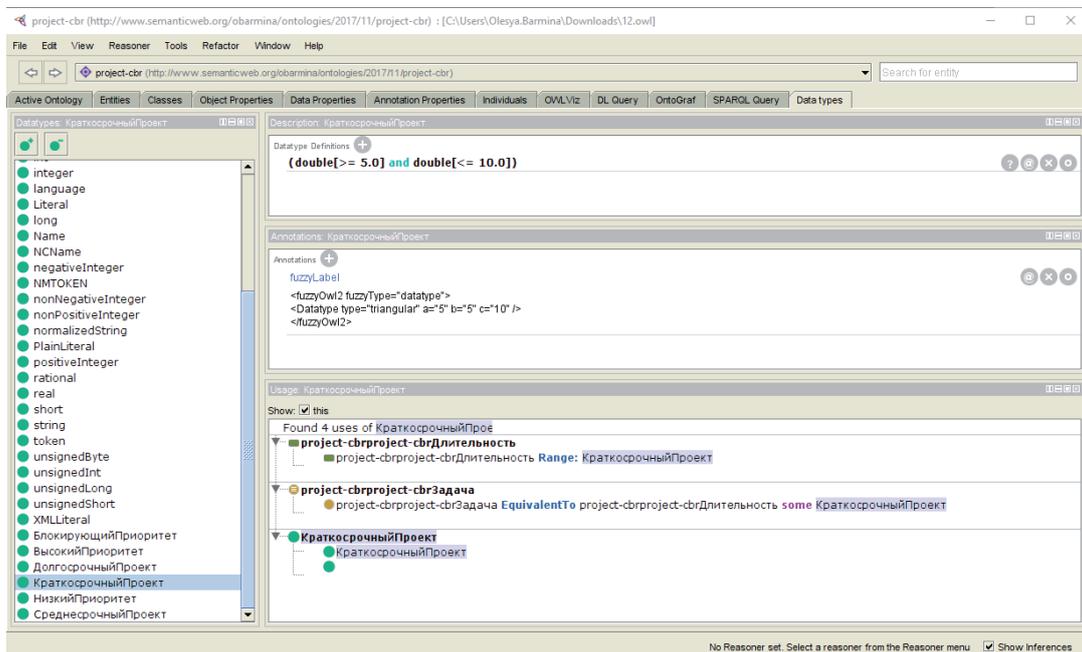


Рисунок 4 – Определение нечёткого типа данных *fuzzy data type* в редакторе онтологий *Protege 5.2.0*

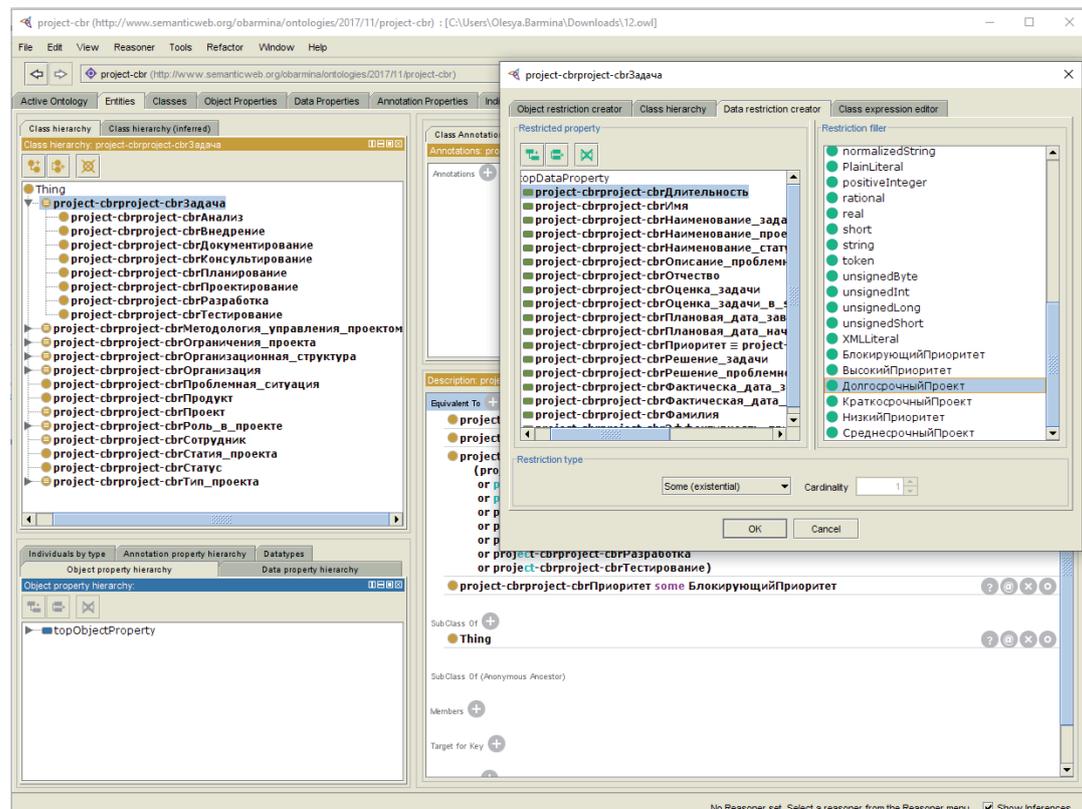


Рисунок 5 – Определение типа данных для свойства *data property* в редакторе онтологий *Protege 5.2.0*

Для определения нечёткой атрибутивной близости требуется провести фаззификацию (введение нечёткости), поскольку в онтологии все значения атрибутов хранятся в виде тер-

мов лингвистической переменной. Фаззификация заключается в установке соответствия между численным значением входной переменной, представленной в виде критериев поискового запроса, и значением функции принадлежности соответствующего ей термина лингвистической переменной. На этапе фаззификации критериям поискового запроса ставятся в соответствие конкретные значения функций принадлежности соответствующих лингвистических термов (рисунок 6).

Таблица 2 – Пример определения нечёткой атрибутивной близости

Признак I , выраженный в виде функции принадлежности $data\ type$	Проект P	Проект C_1	Проект C_2
Чёткое значение длительности проекта	6 лет	1 год	3 года
Нечёткое значение длительности проекта	Долгосрочный, $\mu = 1.0$	Краткосрочный $\mu = 0.1$	Среднесрочный $\mu = 0.5$
$AS_f(\tilde{I}_i, \tilde{I}_j)$ определяется по формуле (5)		$AS_f(\tilde{I}_i, \tilde{I}_j)=0.1$	$AS_f(\tilde{I}_i, \tilde{I}_j)=0.5$

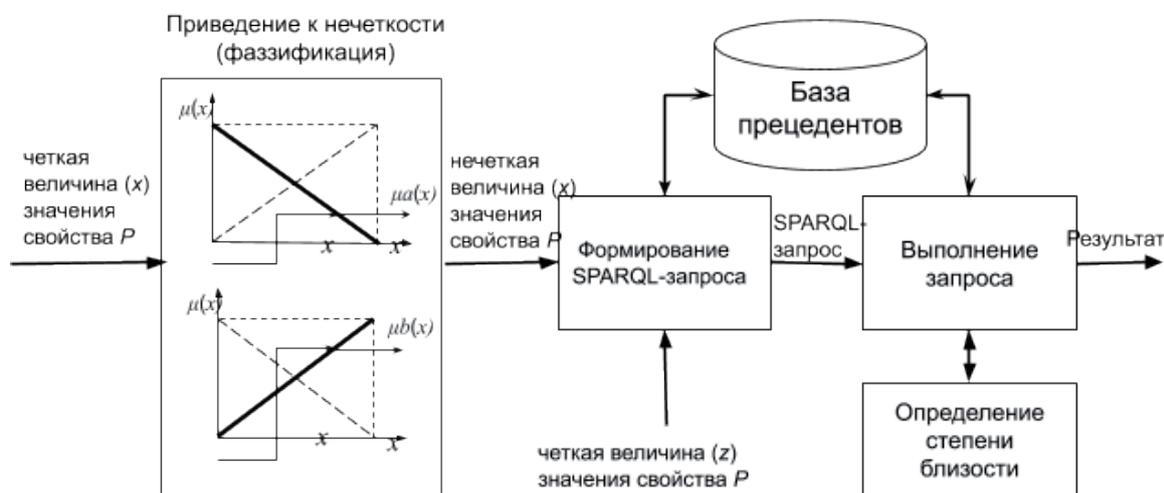


Рисунок 6 – Архитектурная схема поиска прецедента

Пусть переменная *<Длительность проекта>* может принимать любое значение из диапазона от нуля до бесконечности. Каждому значению длительности из указанного диапазона может быть поставлено в соответствие некоторое число от 0 до 1, которое определяет степень принадлежности данной длительности проекта к тому или иному терму лингвистической переменной *<Длительность проекта>*. Степень принадлежности определяется так называемой функцией принадлежности $\mu(x)$. Например, длительности в 1 год (рисунок 7, таблица 2) можно задать степень принадлежности к терму «краткосрочный» равную 0.1, а к терму «среднесрочный» – 0.0.

Шаг 3. Парное сравнение прецедента и ПС для определения реляционной близости $RS(\tilde{I}_i, \tilde{I}_j)$. Реляционная мера сходства $RS(\tilde{I}_i, \tilde{I}_j)$, где $\mu_{\tilde{I}_i}(x) = 1$ и $\mu_{\tilde{I}_j}(x) = 1$, основана на определении подобия ассоциативных отношений ПС.

Шаг 4. Парное сравнение прецедента и ПС для определения таксономической близости $TS(\tilde{I}_i, \tilde{I}_j)$, где $\mu_{\tilde{I}_i}(x) = 1$ и $\mu_{\tilde{I}_j}(x) = 1$. Таксономическая близость между экземплярами \tilde{I}_i и \tilde{I}_j вычисляется с учётом положения соответствующих им понятий C_i и C_j в таксономии

$$(6) \quad H^c: UC(C_i, H^c) = \{H^c(C_i, C_j) \vee C_j = C_i\},$$

где H^c – таксономическая иерархия; UC – вершина, так называемая «верхняя котопия» (*upwards cotopy* – UC) [25]. Таким образом, можно определить таксономическую близость $TS(\tilde{I}_i, \tilde{I}_j)$:

$$(7) \quad TS(\tilde{I}_i, \tilde{I}_j) = \frac{UC(C_i, H^c) \cap UC(C_j, H^c)}{UC(C_i, H^c) \cup UC(C_j, H^c)}$$

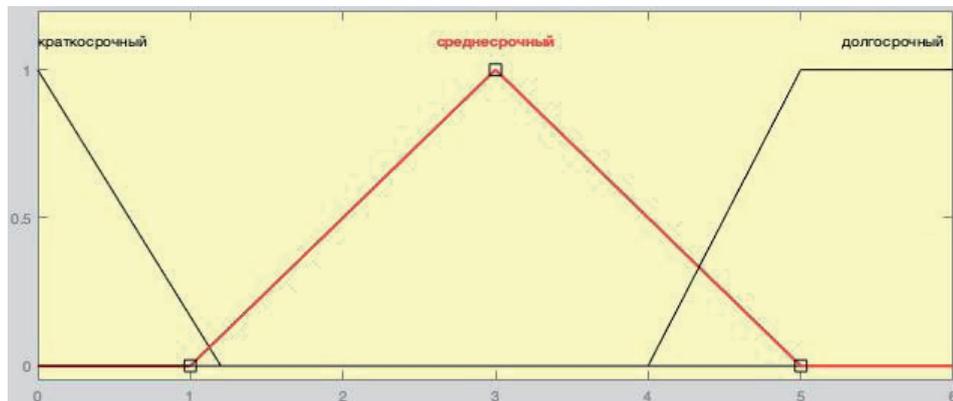


Рисунок 7 – Пример описания лингвистической переменной <Длительность проекта>

Шаг 5. Парное сравнение прецедента и ПС для определения нечёткой таксономической близости $TS_f(\tilde{I}_i, \tilde{I}_j)$

В таблице 3 приведён пример определения нечёткой таксономической близости для двух проектов C_1 и C_2 к проекту P , в котором обнаружена ПС.

Таблица 3 – Пример определения нечёткой таксономической близости

Класс проекта I	Проект P	Проект C_1	Проект C_2	$TS_f(\tilde{I}_i, \tilde{I}_j)$	$TS_f(\tilde{I}_i, \tilde{I}_j)$
Вид проекта	Региональный	Международный	Региональный	<code><fuzzyOwl2 fuzzyType="concept"> <Concept type="weighted" value="0.7" base="C" /></code>	<code><fuzzyOwl2 fuzzyType="concept"> <Concept type="weighted" value="0.77" base="C" /></code>
				$TS_f(\tilde{I}_i, \tilde{I}_j) = 0.7$	$TS_f(\tilde{I}_i, \tilde{I}_j) = 0.77$

Нечёткая таксономическая мера сходства $TS_f(\tilde{I}_i, \tilde{I}_j)$ определяется с использованием меры сходства Заде, где значение функции принадлежности нечёткого класса задаётся в аннотации *fuzzyLabel* в виде параметра *fuzzy classes* (рисунок 8).

Шаг 6. Вычисление глобальной метрики сходства после расчёта локальных контекстно-зависимых метрик сходства.

Общая величина подобия $sim(\tilde{I}_i, \tilde{I}_j)$ экземпляров \tilde{I}_i и \tilde{I}_j определяется формулой:

$$(8) \quad sim(\tilde{I}_i, \tilde{I}_j) = \frac{t \times TS(\tilde{I}_i, \tilde{I}_j) + r \times RS(\tilde{I}_i, \tilde{I}_j) + a \times AS(\tilde{I}_i, \tilde{I}_j) + t_f \times TS_f(\tilde{I}_i, \tilde{I}_j) + a_f \times AS_f(\tilde{I}_i, \tilde{I}_j)}{t + r + a + t_f + a_f}$$

где t, r, a, t_f, a_f – веса различных измерений сходства, которые могут быть подобраны с привлечением экспертных оценок в зависимости от важности учёта различных измерений.

Решение извлекается из найденного похожего прецедента, при необходимости адаптируется к текущей ПС по правилам, разрабатываемым специально для проектно-ориентированной компании в зависимости от условий ее функционирования или для определенных типов программных проектов. Онтологическая модель описания прецедентов ПС в ПП приведена в [17]. В случае, если похожий прецедент не найден, описание ПС сохраняется в онтологии. ЛПР находит решение самостоятельно или с помощью экспертов и вносит принятое решение в онтологию, формируя новый прецедент.

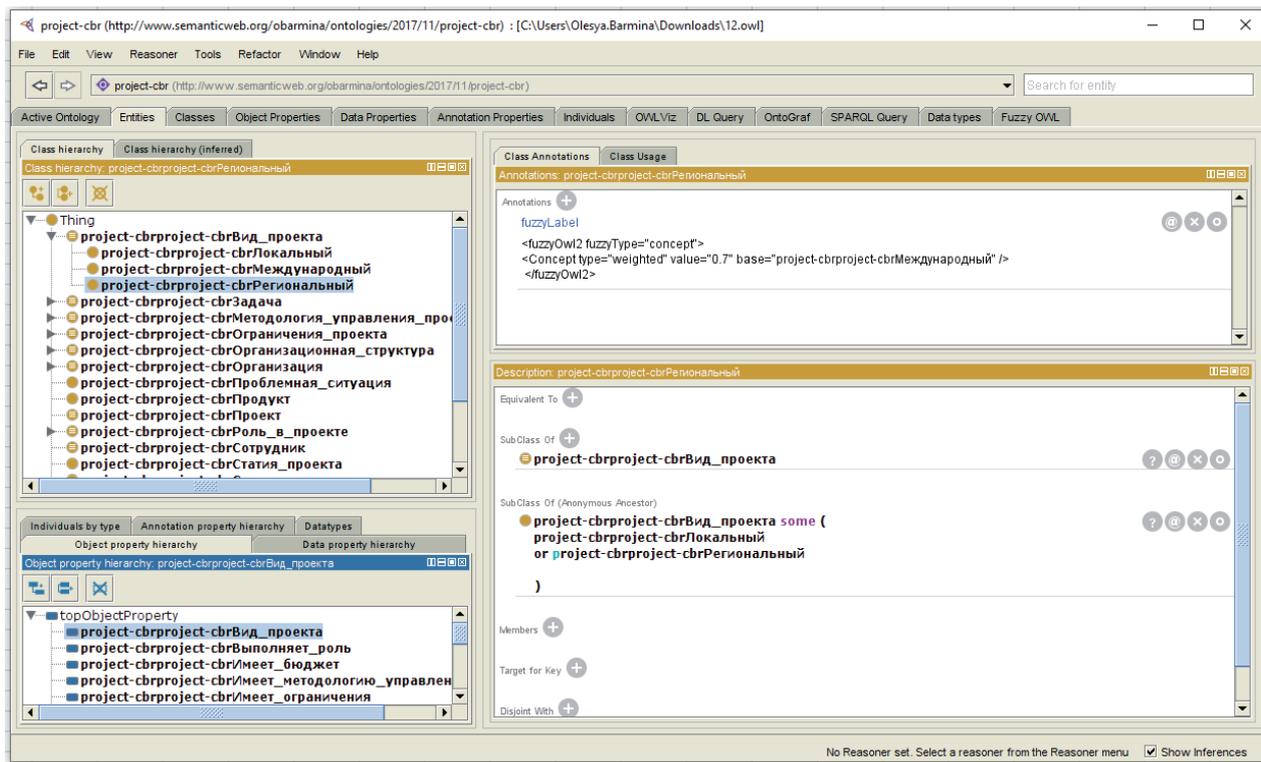


Рисунок 8 – Пример класса с указанной аннотацией *fuzzyLabel*

3 Программная реализация

Для интеллектуальной системы поддержки принятия решений в соответствии с предложенными моделями было разработано расширение к редактору онтологий *Protege 5.2.0 Search-plugin* [26]. Все расширения к редактору онтологий *Protege 5.2.0* типа *TabWidget* реализуются с использованием *OSGi*-спецификации динамической модульной системы и сервисной платформы для разработки *Java*-приложений (рисунок 9).

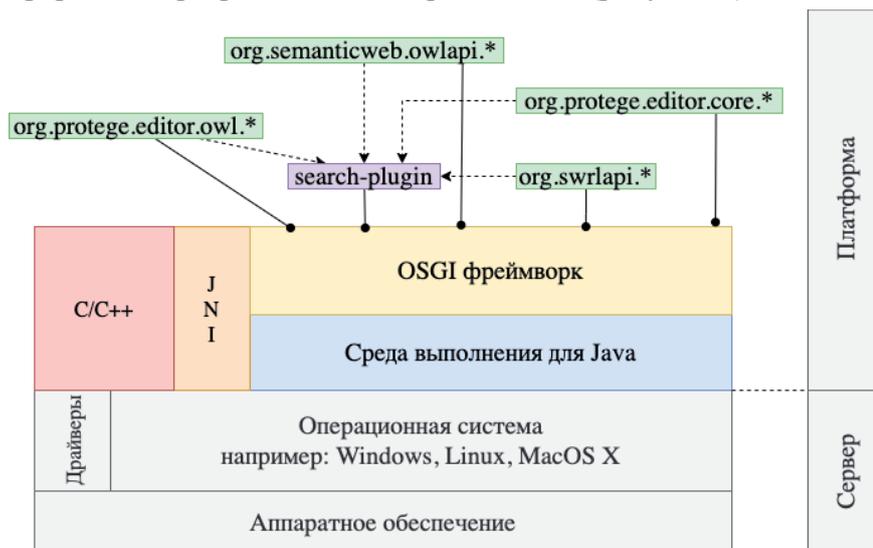


Рисунок 9 – Программная реализация плагина *Search-plugin* к *Protege 5.2.0* для поиска прецедентов проблемных ситуаций

Особенность использования *OSGi* заключается в возможности динамически переинсталлировать компоненты и составные части приложения без необходимости его останова и перезапуска. Круг применений данной спецификации довольно широк: она изначально разрабатывалась для создания встроенных систем (в частности, для автомобилей *BMW*, также в разработке спецификации активно участвует *Siemens*), но сейчас на базе *OSGi* строят многофункциональные автономные настольные приложения (например, *Eclipse SDK*) и корпоративные системы. Модули (плагины, бандлы (*bundle*)) взаимодействуют между собой посредством сервисов: объектов системы с заявленными реализованными интерфейсами. Модули регистрируют сервисы для предоставления определённой функциональности, механизма создания и обработки событий. Модуль *OSGi* (*OSGi bundle*) содержит *java*-классы и другие ресурсы, которые реализуют необходимые функции, а также предоставляют сервисы и пакеты другим модулям. На рисунке 10 представлена процедура запуска разработанного плагина.

```

*****
**                                     Protege
*****

----- Initialising and Starting the OSGi Framework -----
FrameworkFactory Class: org.apache.felix.framework.FrameworkFactory

The OSGi framework has been initialised
----- Starting Bundles -----
Starting bundle org.protege.common
Starting bundle org.eclipse.equinox.common
Starting bundle org.eclipse.equinox.supplement
Starting bundle org.protege.editor.core.application
Starting bundle org.eclipse.equinox.registry
Starting bundle log4j.over.slf4j
Starting bundle com.google.inject
Starting bundle jul.to.slf4j
Starting bundle ch.qos.logback.core
Starting bundle slf4j.api
Starting bundle org.apache.commons.io
Starting bundle search-plugin

```

Рисунок 10 – Запуск плагина *Search-plugin* в *Protege 5.2.0*

Под плагином понимается *Java™ Archive (JAR)* – самодостаточный и самоопределяемый модуль. Самодостаточность означает, что модуль содержит в себе код и ресурсы, необходимые ему для работы. Самоопределяемость означает наличие информации о своей сущности, описание требований модуля к внешней среде и описание собственных возможностей для внешней среды. Плагин (модуль *Search-plugin*) предназначен для поиска похожих прецедентов ПС, возникающих при разработке ПО, путём формирования *SPARQL*-запросов к нечёткой онтологии в диалоговом режиме [26]. Пример формирования запроса на поиск похожих прецедентов в нечёткой онтологии с использованием интерфейса разработанного плагина к редактору онтологии *Protege 5.2.0* и листинг запроса представлены на рисунках 11 и 12 соответственно.

На запрос поиска прецедентов по ПС, возникшей в программном проекте при выполнении задачи со следующими параметрами: затронуты процессы <Разработка документации>, влияет на работу специалиста <Аналитик>, возможная причина проблемной ситуации состоит в <Неправильная оценка проекта> найдено 13 прецедентов (рисунок 11).

4 Оценка эффективности поиска ближайших прецедентов

Для оценки адекватности разработанных моделей, алгоритмов и ПО были проведены вычислительные эксперименты на основании данных о выполнении ПП (разработка системы

электронного документооборота, системы управления финансами, аналитической системы) в различных ПрО для различных заказчиков. Программные проекты выполнялись в одной и той же ИТ-компании, данные для эксперимента собирались в течение 6 месяцев. План экспериментов приведён в таблице 4.

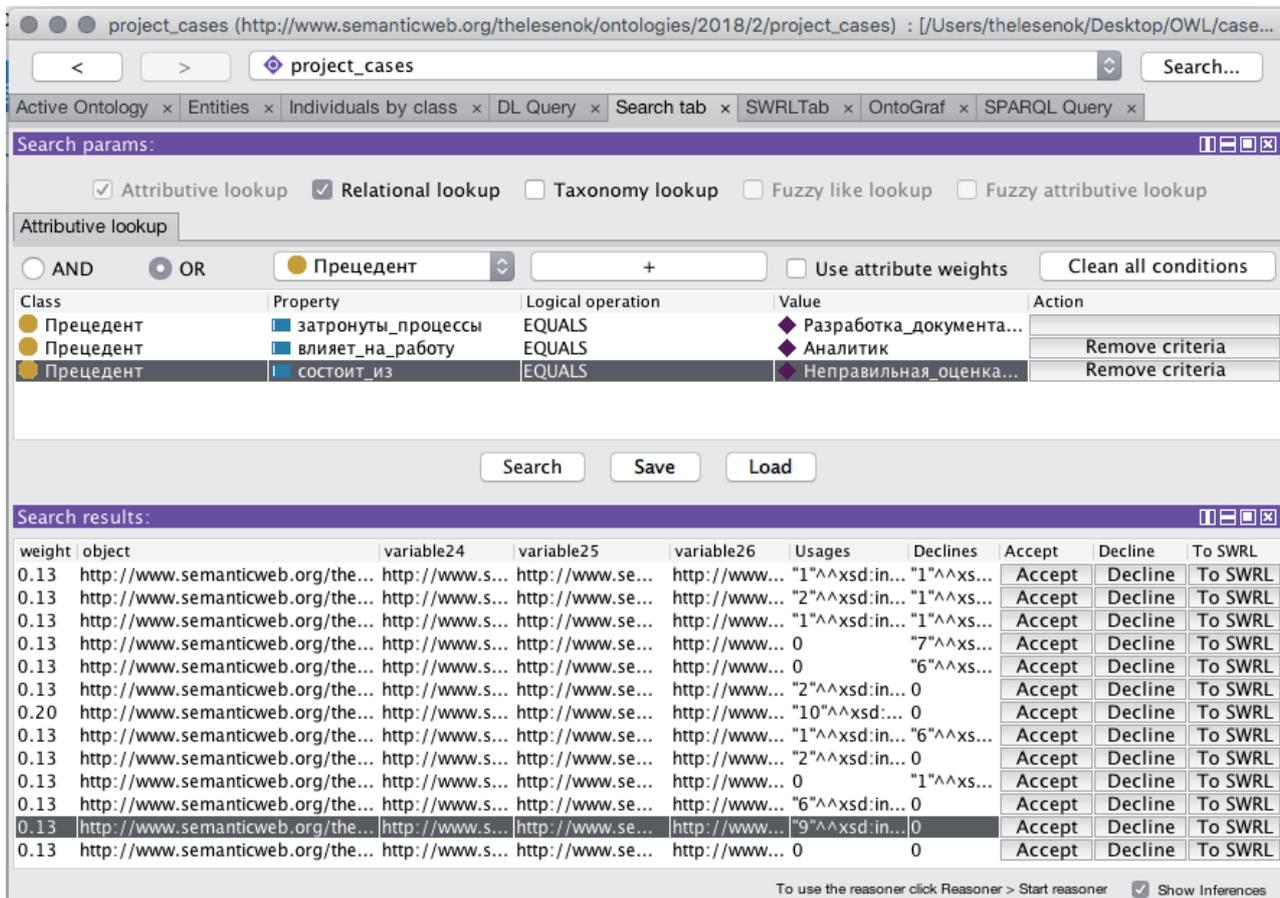


Рисунок 11 – Плагин Search-plugin в редактору онтологий Protege 5.2.0

```
SELECT * WHERE {
?object a <http://www.semanticweb.org/ontologies/2018/2/project_cases#Прецедент>.
?object prfx3:влияет_на_работу ?variable1.
?object prfx3:затронуты_процессы ?variable2.
?object prfx3:состоит_из ?variable3
FILTER ((STR(?variable1) = "http://www.semanticweb.org/ontologies/2018/2/project_cases#Аналитик")
&& (STR(?variable2) =
"http://www.semanticweb.org/ontologies/2018/2/project_cases#Разработка_документации") &&
(STR(?variable3) =
"http://www.semanticweb.org/ontologies/2018/2/project_cases#Неправильная_оценка_проекта"))}
```

Рисунок 12 – Листинг SPARQL-запроса на поиск прецедентов

Таблица 4 – План проведения экспериментов

№ п/п	Наименование этапа
1	Анализ инструментов поиска прецедентов ПС при выполнении ПП
2	Выявление часто встречающихся поисковых запросов
3	Создание и наполнение БЗ (разработка онтологий <i>FuzzyOWL</i> , <i>Jcolibri</i> , разработка базы данных <i>MySQL</i>)
4	Поиск прецедентов в БЗ, которая представлена в виде онтологии <i>FuzzyOWL</i> в <i>Protege 5.2.0</i>
5	Поиск прецедентов в БЗ, которая представлена в виде базы данных <i>MySQL</i>
6	Поиск прецедентов в БЗ, которая представлена в виде онтологии <i>OWL</i> в <i>JColibri</i>
7	Анализ выполненных экспериментов

При проведении исследований оценивались: инструмент собственной разработки (*Search-plugin*), редактор построения SQL-запросов и готовое решение *JColibri* [27] – платформа для построения решений по прецедентам, основывающаяся на онтологиях с дескрипционной логикой. Результаты сравнения инструментов поиска ближайших прецедентов сведены в таблицу 5.

Таблица 5 – Результаты сравнения инструментов поиска прецедентов

№ п/п	Параметр	<i>Search-plugin</i>	<i>SQL</i>	<i>JColibri</i>
Используемые меры сходства				
1	атрибутивная мера сходства	+	+	-
2	реляционная мера сходства	+	+	+
3	таксономическая мера сходства	+	-	+
4	нечёткая атрибутивная мера сходства	+	-	-
5	нечёткая таксономическая мера сходства	+	-	-
6	атрибутивная мера сходства нечёткого значения атрибута	+	-	-
База знаний				
7	<i>OWL</i>	+	-	+
8	<i>FuzzyOWL</i>	+	-	-
9	<i>MySQL</i>	-	+	+
Принятие решения в использовании прецедента				
10	Учёт количества принятия/отказа в использовании прецедента	+	-	-

Эксперимент заключается в последовательном выполнении поисковых запросов, ранжированных по частоте использования (выявленные на шаге 2 плана проведения эксперимента).

Для оценки эффективности выполнения поиска прецедентов использовались параметры: точность поиска, полнота поиска, *F*-мера Ван Ризбергена [28]. Показатель полноты поиска (*R*) характеризует способность системы находить нужные прецеденты ПС для ЛПП, но не учитывает количество нерелевантных прецедентов, выданных в результатах поиска. Показатель точности поиска (*P*) характеризует способность системы находить только релевантные прецеденты ПС. *F*-мера является интегральной метрикой, объединяющей полноту и точность поиска. Результаты проведённого эксперимента по сравнению эффективности поиска прецедентов приведены в таблице 6 и на рисунке 13. Использование предложенного алгоритма повышает интегральный показатель более чем на 40%.

Таблица 6 – Результаты проведения эксперимента

Параметр	<i>Search-plugin</i>	<i>SQL</i>	<i>JColibri</i>
Точность <i>P</i>	0,78	0,33	0,51
Полнота <i>R</i>	0,77	0,45	0,33
<i>F</i> -мера	0,78	0,38	0,40

Для проведения экспериментов были выбраны типовые запросы (всего 10 видов), что позволило рассчитать средние значения величин, приведённых в таблице 6. Номер эксперимента – это факт выполнения типового запроса вида: «ПС произошла в ПП <Наименование проекта> при выполнении задачи типа <Разработка новой функциональности>, связанной с доработкой модуля <Название модуля> исполнителем <Разработчик>, что повлияло на процесс <Проведение приемочного тестирования> и затронуло изменение сроков проекта <Да>». Количество запросов в каждом ПП, которое позволило считать их типовыми, варьировалось от 20 до 224.

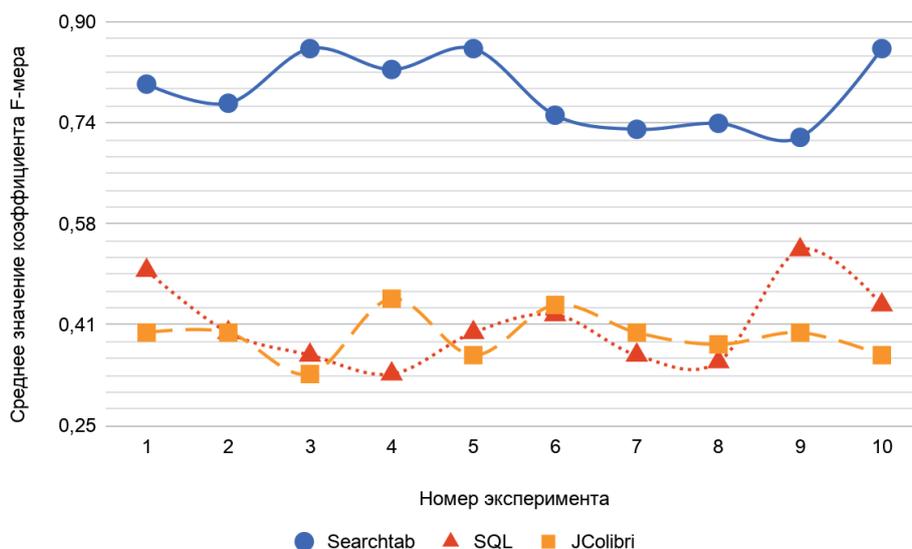


Рисунок 13 – Результаты оценки F -меры поиска похожих прецедентов ПС (фрагмент эксперимента)

Применение предложенных методов и алгоритмов для управления ресурсами проекта в ПС позволило осуществить при выполнении вышеуказанных ПП оперативный поиск прецедентов ПС и снабжение ЛПР необходимой информацией. Использование нечётких свойств в онтологии в качестве критериев позволяет значительно повысить точность и полноту поиска за счёт последовательного сужения области поиска заранее заданными параметрами.

Заключение

Обоснована необходимость разработки нечёткой онтологии для описания процедур поддержки принятия решений в ходе управления ПП. Применение прецедентного подхода в процессе принятия решений позволяет использовать накопленный опыт решения схожих проблем. Разработка новых методов, повышающих эффективность повторного использования прецедентов, остаётся актуальной, особенно для проектов по созданию ПО, характеризующихся высокой степенью уникальности. Предложенный метод использования нечёткой онтологии основан на принципах *FuzzyOWL* и позволяет построить онтологию, отображающую множество классов ПС, требующих решения в ходе выполнения ПП, и структурировать БЗ.

Разработан алгоритм поиска, позволяющий учитывать нечёткие свойства, нечёткие таксономические и ассоциативные отношения. Эффективность алгоритма поиска ближайшего прецедента в нечёткой онтологии подтверждена на основании экспериментальных данных, полученных при тестировании прототипа интеллектуальной системы поддержки принятия решений на проектах по разработке ПО.

Благодарности

Исследования выполнены при финансовой поддержке Российского фонда фундаментальных исследований (грант № 19-08-00937 «Методы и модели поддержки принятия решений при управлении программными проектами в среде производственных предприятий»).

СПИСОК ИСТОЧНИКОВ

- [1] Тренды проектного управления // GANTBPM Управление проектами. - <https://gantbpm.ru/trendy-proektnogo-upravleniya/>.
- [2] **Черняховская, Л.Р.** Разработка моделей и методов интеллектуальной поддержки принятия решений на основе онтологии организационного управления программными проектами / Л.Р. Черняховская, А.И. Малахова // *Онтология проектирования*. 2013. №4 (10). - С.42-52.
- [3] A guide to the project management body of knowledge (PMBOK), Sixth ed. / Project Management Institute, Inc., 14 Campus Boulevard, Newton Square, Pennsylvania 19073-3299 USA, 2017, 756 p.
- [4] **Aamodt, A.** Case-based reasoning: foundational issues, methodological variations, and system approaches / A. Aamodt, E. Plaza // *AI Communications*. IOS Press. Vol. 7: 1. 1994. – P.39-59.
- [5] The Access Group: Inbox Insight Survey. - http://www.theaccessgroup.com/media/250643/5_project_lessons_to_take_into_2014_whitepaper_v2.pdf.
- [6] Complete collection project management statistics. - <https://www.wrike.com/blog/complete-collection-project-management-statistics-2015/>.
- [7] Top project management trends. - http://www.esi-intl.co.uk/resource_centre/white_papers/top-project-management-trends-2016.pdf.
- [8] **Дюбуа, Д.** Теория возможностей. Приложения к представлению знаний в информатике / Д. Дюбуа, А. Прад // Пер. с фр. - М.: Радио и связь, 1990. - 288 с.
- [9] **Ларичев, О. И.** Теория и методы принятия решений / О.И. Ларичев. - М.: Логос, 2006. - 392 с.
- [10] **Мадера, А.Г.** Бизнес-процессы и процессное управление в условиях неопределенности: количественное моделирование и оптимизация / А.Г. Мадера. - М.: ЛЕНАНД, 2019. 160 с.
- [11] **Антонов, В.В.** Формальная модель предметной области на основе нечётких отношений / В.В. Антонов, Г.Г. Куликов // *Программные продукты и системы*. 2011. №2(94). - С.48-51.
- [12] **Антонов, В.В.** Семантико-математический язык описания структуры интеллектуальной системы на основе нечёткой логики / В.В. Антонов, Г.Г. Куликов // *Программные продукты и системы*. 2011. №3(95). - С.33-35.
- [13] **Антонов, В.В.** Построение формальной модели предметной области с применением нечёткой кластеризации / В.В. Антонов, Г.Г. Куликов, Д.В. Антонов // *Вестник УГАТУ: науч. журн. Уфимск. гос. авиац. техн. ун-та*. 2011. Т.15, № 5(45). - С.3-11.
- [14] **Гаврилова, Т.А.** Инженерия знаний. Модели и методы / Т.А. Гаврилова, Д.В. Кудрявцев, Д.И. Муромцев. СПб.: Издательство «Лань», 2016. 324 с.
- [15] **Болотова, Л.С.** Системы искусственного интеллекта: модели и технологии, основанные на знаниях / Л.С. Болотова. - М.: Финансы и статистика, 2012. 664 с.
- [16] **Массель, Л.В.** Онтологический инжиниринг для поддержки принятия стратегических решений в энергетике / Л.В. Массель, Т.Н. Ворожцова, Н.И. Пяткова // *Онтология проектирования*. 2017. Т.7, №1 (23). - С.66-76. - DOI: 10.18287/2223-9537-2017-7-1-66-76.
- [17] **Бармина, О.В.** Интеллектуальная система управления взаимодействием бизнес-процессов в проектно-ориентированных организациях / О.В. Бармина, Н.О. Никулина // *Онтология проектирования*. 2017. Т.7, №1 (23). - С.48-65. - DOI: 10.18287/2223-9537-2017-7-1-48-65.
- [18] **Bobillo, F.** Fuzzy description logics with general t-norms and datatypes / F. Bobillo, U. Straccia // *Fuzzy Sets and Systems*. 2009. 160 (23). P.3382–3402.
- [19] Official documentation on Protégé. - <http://protrgrwiki.stanford.edu/index.php/ProtegeUserDocs>.
- [20] **Ярушкіна, Н.Г.** Применение способа интеграции нечётких временных рядов и нечётких онтологий в задачах диагностики технических систем / Н.Г. Ярушкіна, В.С. Мошкин, Г.Р. Ишмуратова, И.А. Андреев, И.А. Мошкина // *Онтология проектирования*. 2018. Т.8, №4 (30). - С.594-604. - DOI: 10.18287/2223-9537-2018-8-4-594-604.
- [21] **Althof, Klaus-Dieter.** A Review of Industrial Case-Based Reasoning Tools / Klaus-Dieter Althof, Eric Auriol, Ralph Barlette, Michel Manago // *AI Intelligence*, 1995. – p. 159.
- [22] **Люгер, Д.Ф.** Искусственный интеллект: стратегии и методы решения сложных проблем / Д.Ф. Люгер // Пер. с англ. – 4-е изд. – М.: Издательский дом «Вильямс», 2003. 864 с.
- [23] **Левенштейн, В.И.** Двоичные коды с исправлением выпадений, вставок и замещений символов. Доклады Академии Наук СССР, 1965. Т. 163, №4. С.845-848.
- [24] **Zadeh I.** Fuzzy Logic, Neural Networks and Soft Computing / I. Zadeh // *Communication on the ACM*, 1994. – Vol. 37, No 3. – p.77–84.
- [25] **Крюков, К.В.** Меры семантической близости в онтологии / К.В. Крюков, Л.А. Панкова, В.А. Пронина, В.С. Суховеров, Л.Б. Шипилина // *Проблемы управления*. 2010, № 5. С.2–14

- [26] **Черняховская, Л.Р.** Программа подбора прецедентов проблемных ситуаций при управлении проектами / Л.Р. Черняховская, О.В. Бармина, Н.О. Никулина // Свидетельство о государственной регистрации программы для ЭВМ №2018615125 от 26.04.2018.
- [27] Official documentation on JColibri. - <http://gaia.fdi.ucm.es/research/colibri/jcolibri/>.
- [28] **Van Rijsbergen, C.J.** Information Retrieval (2nd ed.). Butterworth-Heinemann. 1979. <http://www.dcs.gla.ac.uk/Keith/Preface.html>.
-

Сведения об авторах



Антонов Вячеслав Викторович, 1956 г. рождения. Окончил Башкирский государственный университет (1979), кандидат технических наук (2007), доктор технических наук (2015). Профессор кафедры автоматизированных систем управления Уфимского государственного авиационного технического университета. В списке научных трудов около 30 работ в области построения интеллектуальных систем поддержки принятия решений. AuthorID (РИНЦ): 530537. Author ID (Scopus): 57200254522; ORCID 0000-0002-5402-9525; Researcher ID (WoS): AАН-5121-2019. Antonov.V@bashkortostan.ru.

Бармина Олеся Владимировна, 1988 г. рождения. Окончила Уфимский государственный авиационный технический университет (2011), магистратуру (2014), а также аспирантуру по направлению «Информатика и вычислительная техника» (2018). Ведущий аналитик ООО "Нордиджи", г. Санкт-Петербург. Научные интересы: реинжиниринг бизнес-процессов, управление программными проектами, интеллектуальные системы поддержки принятия решений. В списке трудов более 20 работ в указанных областях. ORCID 0000-0003-4404-1376; obarmina@outlook.com.



Никулина Наталья Олеговна, 1971 г. рождения. Окончила Уфимский государственный авиационный технический университет (1994), кандидат технических наук (1998). Доцент кафедры автоматизированных систем управления. Научные интересы: исследования в области управления проектами, системного моделирования и проектирования информационных систем в организационном управлении. В списке трудов более 40 работ в указанных областях. AuthorID (РИНЦ): 396597; ORCID 0000-0002-8594-2093. nikulina@outlook.com.

Поступила в редакцию 08.12.2019, после рецензирования 25.02.2020. Принята к публикации 05.03.2020.

Decision-making support in software project management based on fuzzy ontology

V.V. Antonov¹, O.V. Barmina², N.O. Nikulina¹

¹Ufa State Technical Aviation University, Ufa, Russia

²Nordigy LCC, Saint-Petersburg, Russia

Abstract

The article deals with knowledge management in software projects and decision-making support of its participants. The novelty of research is justified by the need to reduce the degree of uncertainty in decision-making and is confirmed by statistics indicating a sufficiently large number of unsuccessful projects. It is proposed to organize decision-making support based on the cases analysis of previously occurring problem situations. Also, it is proposed to use knowledge engineering methods, in particular, ontological analysis of the subject area. A formation of a fuzzy ontology method is developed for decision-making support in the software development. The novelty of the mathematical model lies in the combination of various inference mechanisms when making decisions on the basis of the entire complex of knowledge about the subject area. Increasing amount of information does not allow using simple searches for quick retrieval of the

necessary information. Therefore, an algorithm is proposed for generating a search query that takes into account fuzzy properties, fuzzy taxonomic and associative relations between classes of objects in ontology. The effectiveness of the proposed solutions is confirmed by an experiment on the prototype of an intelligent decision-making support system.

Keywords: *project management, fuzzy ontology, knowledge management, search, retrieval, decision support, incidents, case management.*

Citation: *Antonov VV, Barmina OV, Nikulina NO. Decision-making support in software project management based on fuzzy ontology [In Russian]. *Ontology of designing*. 2020; 10(1): 121-140. DOI: 10.18287/2223-9537-2020-10-1-121-140.*

Acknowledgments: The work is supported by RFBR grant № 19-08-00937 «Methods and models of the intellectual decision-making support in software project management realized on manufacturing enterprises»

List of figures and tables

- Figure 1 – The sequence of building a fuzzy ontology
 Figure 2 – Q_i search query data model
 Figure 3 – The precedent search algorithm scheme in the fuzzy ontology of decision-making support in software project management
 Figure 4 – Defining a fuzzy data type in the Protégé 5.2.0 ontology editor
 Figure 5 – Defining a data type for the data property in the Protege 5.2.0 ontology editor
 Figure 6 – Problem situation case search architecture
 Figure 7 – An example of the description of the linguistic variable «Project Duration»
 Figure 8 – An example class with the specified annotation *fuzzyLabel*
 Figure 9 – Software implementation of the Search-plugin for Protege 5.2.0 for finding precedents for a problem situation
 Figure 10 – Plugin Search-plugin launch
 Figure 11 – Plugin Search-plugin to Protege 5.2.0
 Figure 12 – Listing a SPARQL query for use cases
 Figure 13 – Evaluation results of the F-measure for searching for similar precedents for a problem situation (experiment fragment)
 Table 1 – Key elements of FuzzyOWL
 Table 2 – An example of determining fuzzy attributive proximity
 Table 3 – An example of determining fuzzy taxonomic proximity
 Table 4 – Experiment plan
 Table 5 – Results comparing tools precedents search
 Table 6 – Experiment results

References

- [1] Project Management Trends [In Russian]. <https://gantbpm.ru/trendy-proektnogo-upravleniya/>.
- [2] **Chernyakhovskaya LR, Malakhova AI.** Development of intellectual decision support models and methods based on ontology of software projects organization management [In Russian]. *Ontology of designing*. 2013; 4: 42–52.
- [3] A guide to the project management body of knowledge (PMBOK Guide), Sixth ed. / Project Management Institute, Inc., 14 Campus Boulevard, Newton Square, Pennsylvania 19073-3299 USA, 2017, 756 p.
- [4] **Aamodt A, Plaza E.** Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communications*. IOS Press. 1994; 7(1): 39-59.
- [5] The Access Group: Inbox Insight Survey. http://www.theaccessgroup.com/media/250643/5_project_lessons_to_take_into_2014_whitepaper_v2.pdf.
- [6] Complete collection project management statistics. <https://www.wrike.com/blog/complete-collection-project-management-statistics-2015/>.
- [7] Top project management trends. http://www.esi-intl.co.uk/resource_centre/white_papers/top-project-management-trends-2016.pdf.
- [8] **Dyubua D, Prad A.** Theory of opportunity. Applications to the representation of knowledge in computer science [In Russian]. Moscow: Radio and communication Pubs.; 1990. 288 p.
- [9] **Larichev OI.** Theory and Decision Making Methods [In Russian]. Moscow: Logos Pubs.; 2006. 392 p.
- [10] **Madera AG.** Business processes and process management in the face of uncertainty: quantitative modeling and optimization [In Russian]. Moscow: LENAND Pubs.; 2019. 160 p.

- [11] **Antonov VV, Kulikov GG.** Formal domain model based on fuzzy relations [In Russian]. *Software and systems*. 2011; 2(94): 48-51.
- [12] **Antonov VV, Kulikov GG.** Semantic and mathematical language of intellectual system structure description based on fuzzy logic [In Russian]. *Software and systems*. 2011; 3(95): 33-35.
- [13] **Antonov VV, Kulikov GG, Antonov DV.** Development a formal domain model using fuzzy clustering [In Russian]. *Vestnik UGATU*. 2011; 15(5): 3-11.
- [14] **Gavrilova TA, Kudryavtsev DV, Muromtsev DI.** Knowledge engineering. Models and methods [In Russian]. SPb: Lan' Publs; 2016. 324 p.
- [15] **Bolotova LS.** Artificial intelligence system: models and technologies based on knowledge [In Russian]. Moscow: Finance and statistics Publs. 2012. 664 p.
- [16] **Massel LV, Vorozhtsova TN, Pjatkova NI.** Ontology engineering to support strategic decision-making in the energy sector [In Russian]. *Ontology of designing*. 2017; 7(1): 66-76. DOI: 10.18287/2223-9537-2017-7-1-66-76.
- [17] **Barmina OV, Nikulina NO.** Intellectual system for managing the interaction of business processes in project-oriented organizations [In Russian]. *Ontology of designing*. 2017; 7(1): 48-65. DOI: 10.18287/2223-9537-2017-7-1-48-65.
- [18] **Bobillo F, Straccia U.** Fuzzy description logics with general t-norms and datatypes. *Fuzzy Sets and Systems*. 2009; 160 (23): 3382-3402.
- [19] Official documentation on Protégé. <http://protrgrwiki.stanford.edu/index.php/ProtegeUserDocs>.
- [20] **Yarushkina NG, Moshkin VS, Ishmuratova GR, Andreev IA, Moshkina IA.** Application of fuzzy time series and fuzzy ontology integration in diagnostics of technical systems [In Russian]. *Ontology of designing*. 2018; 8(4): 594-604. DOI: 10.18287/2223-9537-2018-8-4-594-604.
- [21] **Althof K-D, Auriol E, Barlette R, Manago M.** A Review of Industrial Case-Based Reasoning Tools. *AI Intelligence*. 1995, 159 p.
- [22] **Luger DF.** Artificial Intelligence: Strategies and Methods for Solving Complex Problems [In Russian]. Moscow: Williams Publishing House; 2003. 864 p.
- [23] **Levenshtein VI.** Binary codes capable of correcting deletions, insertions, and reversals, DAN SSSR, 1965, 163, No. 4, 845–848 [in Russian] (English translation: *Soviet Physics Doklady*, 1965; 10(8): 707–710).
- [24] **Zadeh I.** Fuzzy Logic, Neural Networks and Soft Computing // *Communication on the ACM*, 1994; 37(3): 77–84.
- [26] **Kryukov KV, Pankova LA, Pronina VA, Suhoverov VS, SHipilina LB.** Measures of semantic proximity in ontology [In Russian]. *Management issues.*, 2010; 5: 2–14.
- [27] **Chernyakhovskaya LR, Nikulina NO, Barmina OV.** The program of selection of precedents of problem situations in project management. Certificate of state registration of computer programs №2018615125; 26.04.2018.
- [28] Official documentation on JColibri. <http://gaia.fdi.ucm.es/research/colibri/jcolibri/>.
- [29] **Van Rijsbergen CJ.** Information Retrieval (2nd ed.). Butterworth-Heinemann. 1979. <http://www.dcs.gla.ac.uk/Keith/Preface.html>.
-

About the authors

Vyacheslav Viktorovich Aatonov (b. 1956) graduated from the Bashkir State University in 1979. PhD in Technical Science (2007), Doctor of Technical Science (2015). Professor of Automated management systems department of the Ufa State Aviation Technical University. He is the co-author about 30 scientific publications in the field of intellectual decision-making support systems development. AuthorID (RCI): 530537. Author ID (Scopus): 57200254522; ORCID 0000-0002-5402-9525; Researcher ID (WoS): AAH-5121-2019. Antonov.V@bashkortostan.ru.

Olesya Vladimirovna Barmina (b. 1988) graduated from the Ufa State Aviation Technical University in 2011, master's degree (2014), post-graduate course (2018) in computer science (Department of technical cybernetics). Now she is working as Senior Data Analyst at Nordigy LLC. Research fields are project management, AI, ontology engineering. She is the co-author more than 20 scientific articles and abstracts. ORCID 0000-0003-4404-1376. obarmina@outlook.com.

Natalya Olegovna Nikulina (b. 1971) graduated from the Ufa State Aviation Technical University in 1994, PhD in Technical Science (1998). She is an associate professor of Automated management systems department of the Ufa State Aviation Technical University. Research fields are project management, system modeling and designing information-systems in organizational management. She is the co-author more than 40 scientific articles and abstracts. AuthorID (RCI): 396597; ORCID 0000-0002-8594-2093. nikulino@outlook.com.

Received December 8, 2019. Revised February 25, 2020. Accepted March 05, 2020.
