

ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ

ONTOLOGICAL MODEL
OF PROJECT ACTIVITYD. Kuzin
I. Danilenko

Summary: The article deals with methodological issues of research of human project activity as one of the types of complex activity. Based on the given methodology, the development of an ontological model of project activity is described. Provides information about the structure and composition of the model, including the model class hierarchy, class properties, and integrity constraints. As a practical result, a link to the source code of the ontological model of project activity in the OWL/RDF format is provided. The model is implemented in the free-distributed ontological editor Protégé version 5.5.3. Hermit 1.3.8.413 is used as a logical inference machine.

Keywords: project activity, ontology, semantics, semantic model, knowledge base, intelligent system, descriptive logic, semantic web, protégé, OWL, RDF.

Кузин Дмитрий Александрович

к.т.н., доцент, Сургутский государственный
университет
kuzin_da@surgu.ru

Даниленко Иван Николаевич

к.т.н., Сургутский государственный университет
vice1@surgu.ru

Аннотация: В статье затронуты методологические вопросы исследования проектной деятельности человека, как одного из видов комплексной деятельности. На основе приведенной методологии описывается разработка онтологической модели проектной деятельности. Приводятся сведения о структуре и составе модели, включая иерархию классов модели, свойства классов и ограничения целостности. В качестве практического результата приводится ссылка на исходный код онтологической модели проектной деятельности в формате OWL/RDF. Модель реализована в свободнораспространяемом онтологическом редакторе Protégé версии 5.5.3, в качестве машины для логического вывода использован Hermit 1.3.8.413.

Ключевые слова: проектная деятельность, онтология, семантика, семантическая модель, база знаний, интеллектуальная система, дескрипционная логика, semantic web, protégé, OWL, RDF.

Введение

В условиях перехода к постиндустриальной экономике информационные технологии начинают играть ключевую роль во всех сферах деятельности человека. В результате уже созданы и продолжают с ускорением темпов создаваться огромные массивы информации. Значительная часть этой информации является слабоструктурированной, то есть малопригодной для эффективной машинной обработки. К подобной информации относится большинство документов, существующих в среде World Wide Web. Данная проблема является следствием того, что документы в среде web создавались, прежде всего, как визуальные представления, удобные для их восприятия человеком. Их внутренняя структура слабо приспособлена для машинного чтения, то есть извлечения из документа *семантики* – смыслового содержания. HTML-разметка не несет в себе почти никакой семантической нагрузки. Современные поисковые системы работают по принципу контекстного поиска, поэтому результаты поиска оказываются, зачастую, нерелевантны. Даже абсолютно бессмысленный, бессвязный текст будет проиндексирован поисковой машиной и займет свое место в результатах поисковой выдачи, если будет содержать достаточное количество ключевых слов.

Но и с данными, хранящимися в базах в хорошо структурированном виде, тоже возникают проблемы. С

такими данными можно эффективно и просто работать в пределах той системы, для которой проектировалась их структура, однако использование данных вне «родной» системы сопровождается теми же проблемами. Семантика таких данных обычно содержится в неявном виде в коде программного обеспечения, которое их обрабатывает. Эффективного способа извлечь и передать эту семантику за пределы системы не существует. В результате организация любого внешнего взаимодействия информационной системы требует согласования форматов данных и написание сложных процедур интеграции. Отчасти проблема решается разработкой единых спецификаций на структуру данных и использованием единых механизмов представления на основе языка XML, но такие решения, как правило, являются отраслевыми и используются в частных случаях.

В самом общем виде решение описанной проблемы может заключаться в переходе к «знаниевой» модели, когда в информационных системах, в том числе в web, будут определены универсальные способы представления семантики, а сами данные будут от этой семантики неотделимы. *Семантические технологии* представляют собой следующий шаг в развитии машинно-читаемых способов представления информации. Они дают возможность реализовать в электронном виде концептуальные модели, логика которых схожа с логикой человеческого мышления. Помимо хранения информации семантические модели позволяют вести машинное

чение информации, то есть «понимание» ее смысла, а также осуществлять логический вывод на основании правил.

Системы, основанные на знаниях, известны достаточно давно [1]. Существуют различные подходы к построению подобных систем, относящихся к классу интеллектуальных. Наиболее перспективным и современным из них может считаться т.н. «онтологический» подход, в основе которого лежит представление любой информации в виде *онтологии* – семантического графа произвольной структуры, узлами которого являются концепты, а ребра – отношениями. На формальном уровне онтология – это система, состоящая из набора понятий и набора утверждений об этих понятиях, на основе которых можно описывать классы, отношения, функции и индивиды. Неформально онтология представляет собой некоторое описание взгляда на мир применительно к конкретной области интересов. Это описание состоит из терминов и правил использования этих терминов, ограничивающих их значения в рамках конкретной области.

Базовым способом представления семантической информации является *триплет* – синтаксическая структура, состоящая из трех элементов: субъекта, предиката и объекта (рисунк 1). Триплет может выражать связь между двумя концептами, семантика (смысл) которой выражается предикатом, например, «Кошка – Является – Млекопитающим». Также в качестве предиката может выступать свойство субъекта, а в качестве объекта – значение этого свойства, например, «Кошка – Имеет_имя – Мурка». То есть в качестве объекта может выступать либо сущность, либо литерал (числовое или строковое значение).

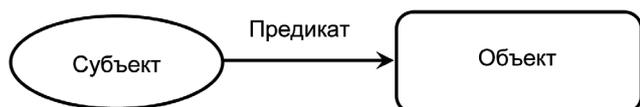


Рис. 1. Триплет – базовая единица представления семантической информации

Важной частью семантической модели является принцип уникальности имен объектов. В концепции Semantic Web, которая описывается ниже, принято, чтобы эти имена имели вид URI (Uniform Resource Identifier – унифицированный (единообразный) идентификатор ресурса).

Наиболее известным применением онтологического подхода является проект Semantic Web – семантический web. Это концепция развития web-среды по внедрению в информацию метаданных, которые позволят хранить и обрабатывать семантику документов. Кроме того, использование специальных инструментов для извлечения данных позволит осуществлять не контекстный, а

«семантический поиск», т.е. фактически искать ответ на вопрос и использованием логического вывода. Технологии Semantic Web могут быть использованы и в других областях – для интеграции систем, описания и каталогизации информационных ресурсов, создания интеллектуальных программных агентов.

Онтологический подход также имеет ряд преимуществ при реализации информационных моделей сложных формализуемых предметных областей [2]:

1. «Открытость» модели данных, предусматривающая ее расширение путем добавления новых концептов и отношений в течение всего жизненного цикла системы.
2. Возможность моделирования сложных связей и отношений и применение логического вывода.
3. Использование согласованной (разделяемой всеми) терминологии с точно определенной семантикой.

Методология проектной деятельности

Проектный подход в управлении последнее время очень популярен в самых различных областях промышленности, бизнеса, государственного управления и общественных некоммерческих организаций. Однако, несмотря на большую популярность проектного управления, все инструменты менеджмента в этой сфере остаются «плоскими» и не используют возможности интеллектуальных информационных систем. Разработка, апробация и стандартизация онтологической модели проектной деятельности может дать старт разработке и внедрению таких систем.

Комплексная деятельность – целенаправленная активность человека, обладающая нетривиальной внутренней структурой, с множественными и/или изменяющимися субъектом, технологией, ролью предмета деятельности в его целевом контексте [3].

Одной из форм комплексной деятельности человека или организации является *проектная деятельность* – комплексная деятельность, которая разбивается на отдельные завершённые циклы, называемые проектами.

Проект – это ограниченное во времени целенаправленное изменение отдельной системы с установленными требованиями к качеству результатов, возможными рамками расхода средств и ресурсов и специфической организацией [4].

Характерными признаками проекта являются:

1. направленность на достижение конечных целей, определенных уникальных результатов;
2. координированное выполнение многочисленных

взаимосвязанных работ с поуровневой детализацией по видам деятельности, ответственности, объемам и ресурсам;

3. ограниченная протяженность во времени, с определенными моментами начала и завершения;
4. ограниченность требуемых ресурсов;
5. специфическая организация управления.

Таким образом, ключевым отличием проектной деятельности от процессной деятельности (регулярной, повторяющейся, цикличной деятельности) является ее *однократность*, то есть нецикличность.

По основным сферам деятельности, в которых осуществляется проект можно выделить следующие типы проектов: технический, организационный, экономический, социальный, образовательный, инвестиционный, инновационный, научно-исследовательский, учебный, смешанный.

Обычно проект включает в себя несколько стадий выполнения, или *мероприятий* – отдельных комплексов работ или операций, направленных на достижение целей проекта. В свою очередь, мероприятие также может состоять из других мероприятий. Таким образом, для проектной деятельности характерна *фрактальность* – каждое из мероприятий может при необходимости рассматриваться как самостоятельный проект.

С учетом вышесказанного сформулируем требования к методологии проектной деятельности:

1. иерархичность, вложенность и фрактальность;
2. отражение причинно-следственных связей между элементами;
3. описание структуры целей проектной деятельности, а также характеристики степени их достижения;
4. описание исполнителей как по отдельным мероприятиям, так и по проекту в целом;
5. учет использования ресурсов.

Общий алгоритм или последовательность действий, которые необходимо выполнить для создания онтологии можно описать следующим образом:

1. перечисление категорий, обозначающих сущности или явления в моделируемой области;
2. связывание категорий определенными отношениями;
3. соотнесение категориям наборов конкретных экземпляров.

Разработка онтологической модели

Ввиду отсутствия распространенных русскоязычных проектов создания онтологий верхнего уровня, принятых в качестве стандарта, разработанная онтологическая модель включает в себя не только модель предметной области, но и базовые концепты, описывающие фундаментальные понятия. Разработка онтологии осуществлялась в наиболее популярном свободно распространяемом онтологическом редакторе с открытым исходным кодом Protégé [5]. Разработанная иерархия классов, отображаемая в окне редактора, приведена на рисунке 2.

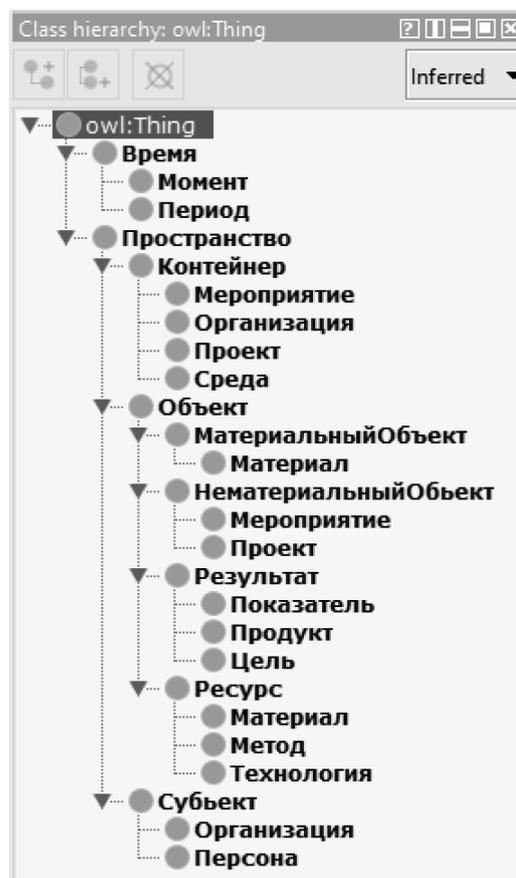


Рис. 2. Иерархия классов онтологии проектной деятельности

Важнейшей частью онтологической модели является набор терминологических аксиом, образующих иерархию свойств объектов (object properties). Они определяют набор и характеристики *ролей* – семантических отношений между концептами. Все эти отношения в модели проектной деятельности можно описать, разделив логику модели на несколько «областей», включающих связанные по смыслу отношения. Эти области реализованы в виде четырех отношений верхнего уровня – «Связан во времени», «Связан структурно», «Связан организационно», «Связан причинно-следственно» (рисунк 3).

Основной структурной сущностью онтологии проектной деятельности является *Проект*, который может включать в себя произвольное количество *Мероприятий*. Мероприятия, в свою очередь, могут включать в себя другие мероприятия с неограниченным уровнем

вложенности. Данная структура реализуется транзитивными отношениями *Предусматривает_выполнение* и инверсным ему *Выполняется_в* (рисунок 4).



Рис. 3. Иерархия object properties онтологии проектной деятельности

Для описания отношений между концептами во времени предусмотрены два класса – *Момент* и *Период*. При этом для момента определено свойство данных (data properties) типа `xsd:TimeStamp`. Для класса *Период*, в свою очередь, определены отношения *Начинается* и *Заканчивается*, связывающие его с классом *Момент*, а также обратные отношения *Является_началом* и *Является_окончанием*. Также между моментами времени определены отношения *Раньше* и *Позднее*.

Любое мероприятие или проект может выполняться в течение периода времени. Для этого определены отношения *Выполняется_в_течение* и обратное ему *Включает_выполнение*. Рассмотрим пример определения свойства *Выполняется_в_течение*. Данное свойство связывает класс *Проект* или *Мероприятие* с классом *Период*. Семантически данное свойство показывает, что проект или мероприятие выполнялись в течение определенного временного периода. Пусть «Мероприятие 1» выполняется в течение временного периода «Период 1». При этом необходимо, чтобы «Мероприятие 2», которое выполняется в рамках «Мероприятия 1» (свойство *Выполняется_в*) также определялось как мероприятие, выполняемое в течение временного периода «Период 1». Для этого создается цепочка свойств (chain) *Выполняется_в* -> *Включен_в*, которая делается подсвойством (SubProperty) свойства *Выполняется_в_течение* (рисунок 5). В силу транзитивности свойства *Включен_в* свойство *Выполняется_в_течение* будет распространяться на всю иерархию вложенных мероприятий с любы числом уровней.

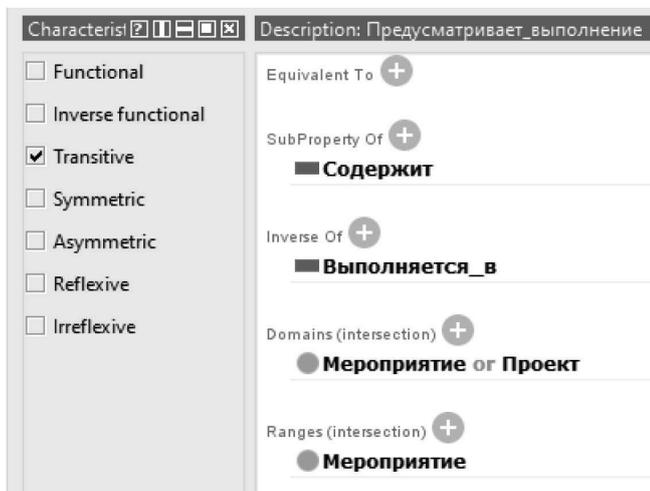


Рис. 4. Определение свойства *Предусматривает_выполнение*

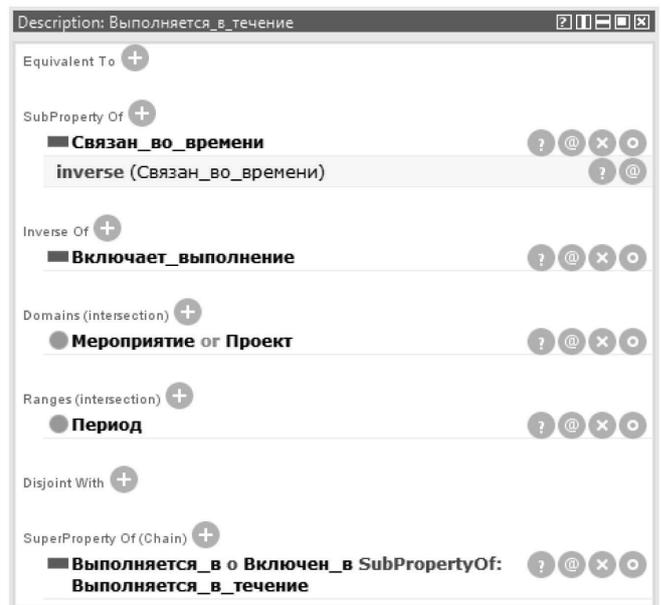


Рис. 5. Определение вложенной цепочки свойств для *Выполняется_в_течение*

Организационные отношения обеспечивают семантику участия субъекта (организации или персоны) в про-

екте или мероприятии. Для этих целей предусмотрены два свойства – *ВыполняетсяСУчастием* и инверсное ему *Участствует*. Как видно на рисунке 6, свойство связывает мероприятие или проект с субъектом. Для того, чтобы отразить семантику участия субъекта в мероприятии 1, в том случае, если он явно участвует в мероприятии 2, выполнение которого предусматривает мероприятие 1, необходимо задание цепочки свойств *Предусматривает_выполнение* о *ВыполняетсяСУчастием* *SubPropertyOf*: *ВыполняетсяСУчастием* (см. рисунок). Это обеспечивает то, что субъект, участвующий во «вложенном» мероприятии автоматически становится участником мероприятия «верхнего уровня». Причем за счет транзитивности свойства *Предусматривает_выполнение* это распространяется на неограниченное число уровней вложенности мероприятий (рисунок). Аналогичную функцию выполняет цепочка свойств *Участствует* о *Выполняется_в* *SubPropertyOf*: *Участствует*.

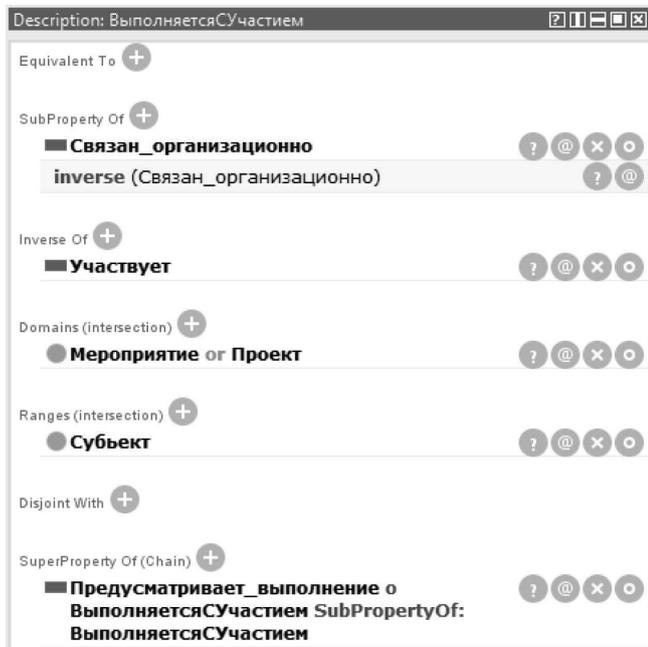
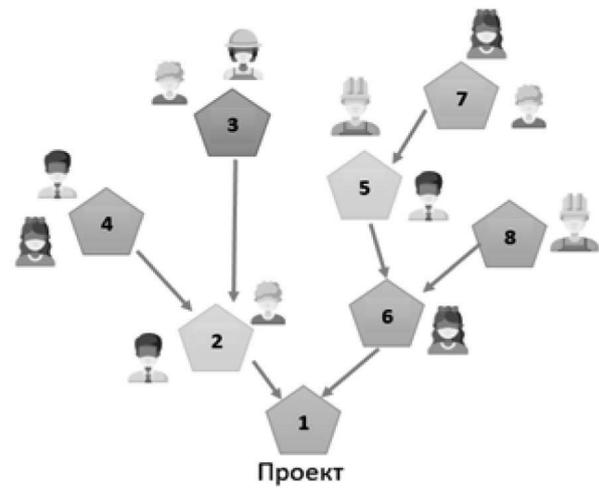


Рис. 6. Определение вложенной цепочки свойств для *ВыполняетсяСУчастием*

Транзитивность отношений и возможность определения цепочек отношений позволяют эффективно обрабатывать иерархические структуры семантических данных – деревья. Примером такой структуры является иерархия мероприятий проекта, каждое из которых может иметь произвольное число исполнителей. Предположим, что требуется получить список всех исполнителей проекта. Если такая структура данных была бы представлена в традиционной табличной форме, то для решения этой задачи потребовались бы реализовать достаточно сложный рекурсивный алгоритм. Использование онтологической модели позволяет решить такую задачу с помощью простого DL-запроса (рисунок 7).

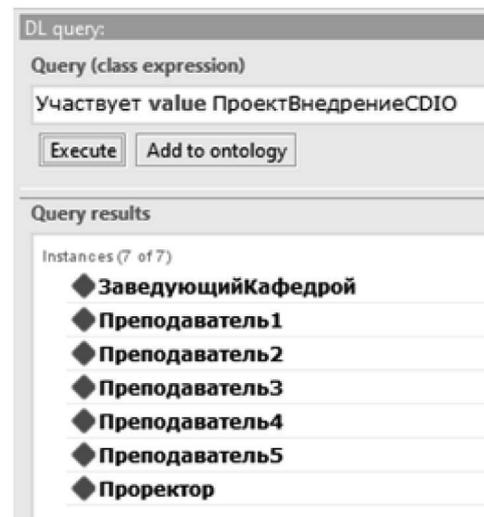


Рис. 7. Обработка иерархии с помощью DL-запроса

Группа причинно-следственных отношений обеспечивает связи между материальными и нематериальными объектами, отражающие связи сущностей «показатель», «результат», «цель», а также «метод», «ресурс», «технология» с проектом или мероприятием с использованием свойств «достигается» и «используется» соответственно.

После проектирования терминологии онтологической модели (определения интенциональных знаний) модель была наполнена экстенциональными знаниями, то есть созданы экземпляры классов (индивидуумы) и определены их свойства. В качестве примера при реализации онтологии проектной деятельности были созданы экземпляры классов и их свойства, описывающие проект «Внедрение проектно-ориентированного обучения в формате CDIO», реализуемый на кафедре Автоматики и компьютерных систем СурГУ. В онтологии были созданы индивидуумы, соответствующие мероприятиям, ресурсам проекта, а также персонам, которые их реализуют (рисунок 8).

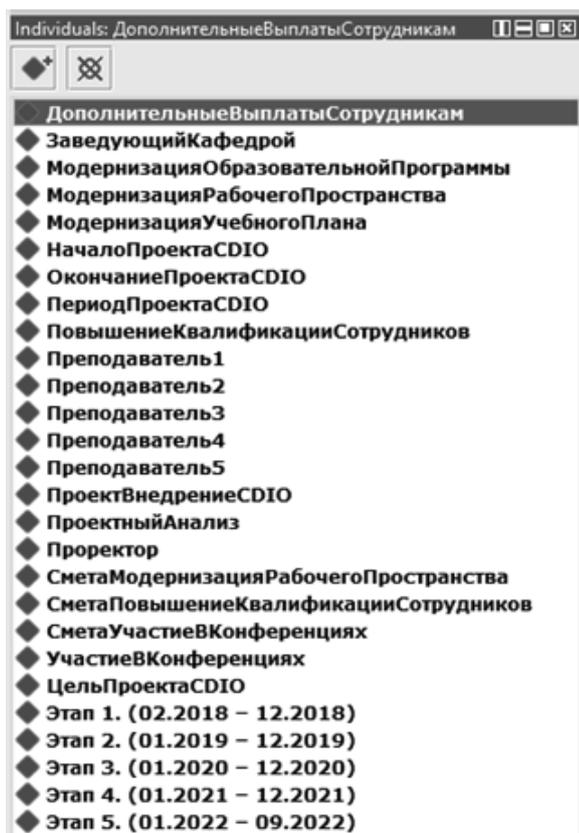


Рис. 8. Экземпляры классов

Статистические данные онтологической модели, включающие сведения о количестве созданных объектов, экземпляров, свойств и т.д. приведены на рисунке 9. Разработанная онтологическая модель включает 23 класса, 39 свойств. Экстенциональная часть модели со-

стоит из 27 экземпляров классов (индивидов), для которых определены 40 значений свойств. Общее количество аксиом в модели – 398.

Metrics	
Axiom	398
Logical axiom count	223
Declaration axioms count	88
Class count	23
Object property count	34
Data property count	5
Individual count	27
Annotation Property count	2
DL expressivity	SROIF(D)

Рис. 9. Статистические данные онтологической модели

Заключение

В данной статье были сформулированы методологические требования и разработана онтологическая модель проектной деятельности. В ходе построения модели авторами были предложены, реализованы и апробированы типовые паттерны онтологического моделирования организационных, структурных, временных и причинно-следственных отношений. Исходный код онтологической модели проектной деятельности в формате OWL/RDF размещен по ссылке http://ontology.student.surgu.ru/project_activities/1.0.0. Разработанная модель обладает свойством практически неограниченного расширения как за счет усложнения структуры и семантики разработанной онтологии, так и за счет ее объединения с другими онтологиями.

ЛИТЕРАТУРА

1. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. Учебник. СПб: Питер, 2001.
2. Кучуганов В.Н. Онтология и анмация прецедентов // Онтология проектирования, 2016.
3. Белов М.В., Новиков А.М. Методология комплексной деятельности. Москва: Ленанд, 2018.
4. Новиков Д.А. Управление проектами: организационные механизмы. Москва: ПМСОФТ, 2007.
5. Stanford University. Protégé - a free, open-source ontology editor and framework for building intelligent systems [Электронный ресурс] URL: <https://protege.stanford.edu/>

© Кузин Дмитрий Александрович (kuzin_da@surgu.ru), Даниленко Иван Николаевич (vice1@surgu.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»