

УДК 62.50

В. П. Пермяков, О. И. Шеховцов

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

## Система визуализации онтологической модели предметной области

*Описывается система визуализации онтологической модели представления знаний. Система реализована при помощи объектно-ориентированного программирования на языке Java с использованием графической библиотеки Swing. Приводятся основные сведения о структуре онтологического формализма, ее скриншоты, отражающие представления компонентов модели. Проблемно-независимый и проблемно-зависимый уровни модели формируются посредством нового улучшенного интерфейса программы, который позволяет создавать и редактировать модель предметной области.*

*Представлены примеры окон для создания объектов предметной области, определения их атрибутов и их свойств и характеристик, а также определения связей между объектами. Связи определяются прямыми линиями различного цвета. С помощью графического интерфейса можно создавать неограниченное количество объектов системы, а также редактировать их атрибуты и связи согласно модели данных.*

### Система визуализации, онтологическая модель, проблемно-независимая модель, проблемно-ориентированная модель, концепты, связи

Целью создания системы было представление автоматизированной системы визуализации онтологической модели предметной области в графической форме. На первом этапе раскрывается формализм проблемно-независимой модели (ПНМ) первого уровня, в соответствии с которой в дальнейшем формируется так называемая проблемно-ориентированная модель второго уровня.

Одним из основных компонентов любой информационной системы является предметная область. Предметную область (ПО) можно определить как сферу человеческой деятельности, выделенную и описанную согласно установленным критериям. В описываемое понятие должны входить сведения об ее элементах, явлениях, отношениях и процессах, отражающих различные аспекты этой деятельности.

В данной статье в качестве объекта визуализации рассматривается онтология ПНМ (спецификационная модель) [1] предметной области «Животный мир», структура и основные характеристики которой приводятся далее.

В основу формализма структурной модели положены следующие принципы: принцип иерархической многоуровневости (последовательное раскрытие неопределенности – от абстрактного к конкретному); принцип многоаспектности описания проблемной области.

Согласно первому принципу модель содержит следующие уровни:

- модель нулевого уровня – абстрактную модель (АМ) ПО;
- модель первого уровня – ПНМ ПО;
- модель второго уровня – проблемно-зависимую или проблемно-ориентированную (ПЗМ) ПО.

Абстрактная модель включает минимально возможную совокупность предельно широких (абстрактных) понятий и отношений между ними. Модели последующих уровней должны являться ее конкретизацией, последовательно раскрывая описываемую проблемную область.

Проблемно-независимая модель рассматривается как экстенциональное представление (расширение) абстрактной модели и интенционал проблемно-ориентированной модели.

В соответствии со вторым принципом проблемно-ориентированная модель должна формироваться в понятийном, информационном и функциональном аспектах последовательным применением следующих правил:

- правила конкретизации: для каждого понятия ПНМ, рассматриваемого как интенционал, формируется из понятий ПО соответствующий

экстенционал. При этом данное правило последовательно применяется не только к понятиям ПНМ, но и к общим понятиям ПЗМ;

– правила интерпретации: для устранения семантической неоднозначности применяемых понятий и формирования тезауруса ПО каждому понятию ПЗМ должна быть поставлена в соответствие его однозначная интерпретация в данной ПО;

– правила структуризации: для каждого структурно-сложного понятия должно быть сформулировано его структурное описание. При этом в соответствии с положениями системного анализа структурное описание должно отражать как внутреннюю структуру объекта, представляемого данным понятием, так и его связи с окружением в данной ПО.

Формально АМ может быть описана следующим образом:

АМ ::= {ОБЪЕКТ, СВЯЗЬ, АТРИБУТ}

ОБЪЕКТ – некоторое понятие предметной области, имеющее самостоятельный смысл;

СВЯЗЬ – определяет существование отношений между двумя и более ОБЪЕКТАМИ;

АТРИБУТ – поименованная характеристика ОБЪЕКТА или СВЯЗИ.

Для наполнения данной модели конкретным содержанием воспользуемся результатами разработок в области спецификации программных систем.

Совокупность исходных (проблемно-независимых) понятий представим в следующем виде:

$T = \langle TA, T_{пн} \rangle$ ,

где  $TA$  – совокупность абстрактных понятий;  $T_{пн}$  – совокупность проблемно-независимых понятий.

Тогда построение структурной модели ПО есть проблемная ориентация проблемно-независимой модели, выполняемая в соответствии с введенными правилами.

Основные компоненты в данной версии ПНМ (задача их формирования является самостоятельной и определяется моделируемой предметной областью) в формальном представлении имеют вид

$\langle \text{ПНМ} = \{ \text{ЗАДАЧА, ДАННОЕ, МОДЕЛЬ, МЕТОД, ПРОЦЕСС, СИСТЕМА, СТРУКТУРА, КОМПОНЕНТА, ПРОЦЕДУРА, ДЕЙСТВИЕ, БАЗИС} \} \rangle$ .

Определим контекстно-зависимый тезаурус, соответствующий данной модели.

ЗАДАЧА – ЕСТЬ некоторый желаемый результат из множества результатов, для достижения которых предназначена СИСТЕМА;

ДАННОЕ (снм. ЭЛЕМЕНТ\_ДАННЫХ) – ЕСТЬ наименьшая единица поименованных сведений, используемых в данной предметной области в различных приложениях;

МОДЕЛЬ – ЕСТЬ формализованное описание постановки ЗАДАЧИ, обеспечивающее возможность ее автоматизированного (автоматического) решения;

МЕТОД – ЕСТЬ описание пути решения ЗАДАЧ определенного класса на концептуальном уровне (непроцедурное);

ПРОЦЕСС – ЕСТЬ последовательность ПРОЦЕДУР, выполняемых для решения определенной ЗАДАЧИ;

СИСТЕМА – ЕСТЬ определенным образом взаимосвязанная и взаимодействующая по определенной программе совокупность КОМПОНЕНТ, системные (интегративные) СВОЙСТВА которой отличаются от СВОЙСТВ образующих ее КОМПОНЕНТ; СИСТЕМА – ЕСТЬ средство решения ЗАДАЧИ;

СТРУКТУРА – ЕСТЬ способ организации КОМПОНЕНТ в ОБЪЕКТЕ;

КОМПОНЕНТА (снм. элемент) – ЕСТЬ\_ЧАСТЬ структурно-сложного ОБЪЕКТА;

ПРОЦЕДУРА – ЕСТЬ КОМПОНЕНТА\_ПРОЦЕССА – набор последовательно выполняемых ДЕЙСТВИЙ над одним и тем же набором ДАННЫХ;

ДЕЙСТВИЕ – ЕСТЬ\_ЧАСТЬ (элементарная) ПРОЦЕССА – ЕСТЬ КОМПОНЕНТА ПРОЦЕДУРЫ;

БАЗИС – ЕСТЬ совокупность структурно-неделимых в рамках определенного ОБЪЕКТА элементов, на которых реализуется данный ОБЪЕКТ; – ЕСТЬ среда реализации ОБЪЕКТА;

::= {} – определение области значений понятия (домен);

=> <> – определение математической структуры понятия.

Введем в рассмотрение связи и атрибуты:

СВЯЗИ := {ЕСТЬ, ЕСТЬ\_НЕК, ЕСТЬ\_ЧАСТЬ, СЛЕДОВАТЬ\_ЗА, СЛЕДОВАТЬ\_ДО, СЛЕДОВАТЬ\_ВМЕСТЕ\_С, ТРЕБОВАТЬ},

где ЕСТЬ – связь определительного типа: А есть (определение);

ЕСТЬ\_НЕК – СВЯЗЬ классификационного типа (родовидового); ЕСТЬ\_ЧАСТЬ – СВЯЗЬ структурного типа (внутриобъектная);

СЛЕДОВАТЬ\_ЗА – СВЯЗЬ типа взаимодействия;

СЛЕДОВАТЬ\_ДО – СВЯЗИ типа взаимодействия;

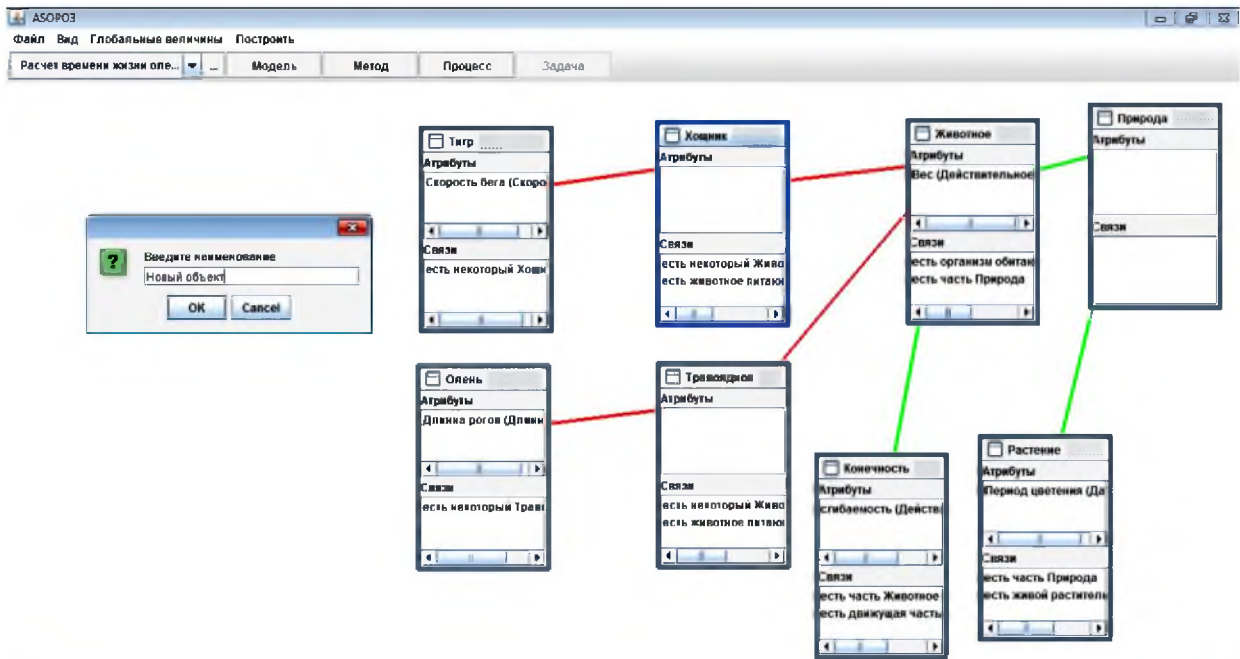


Рис. 1

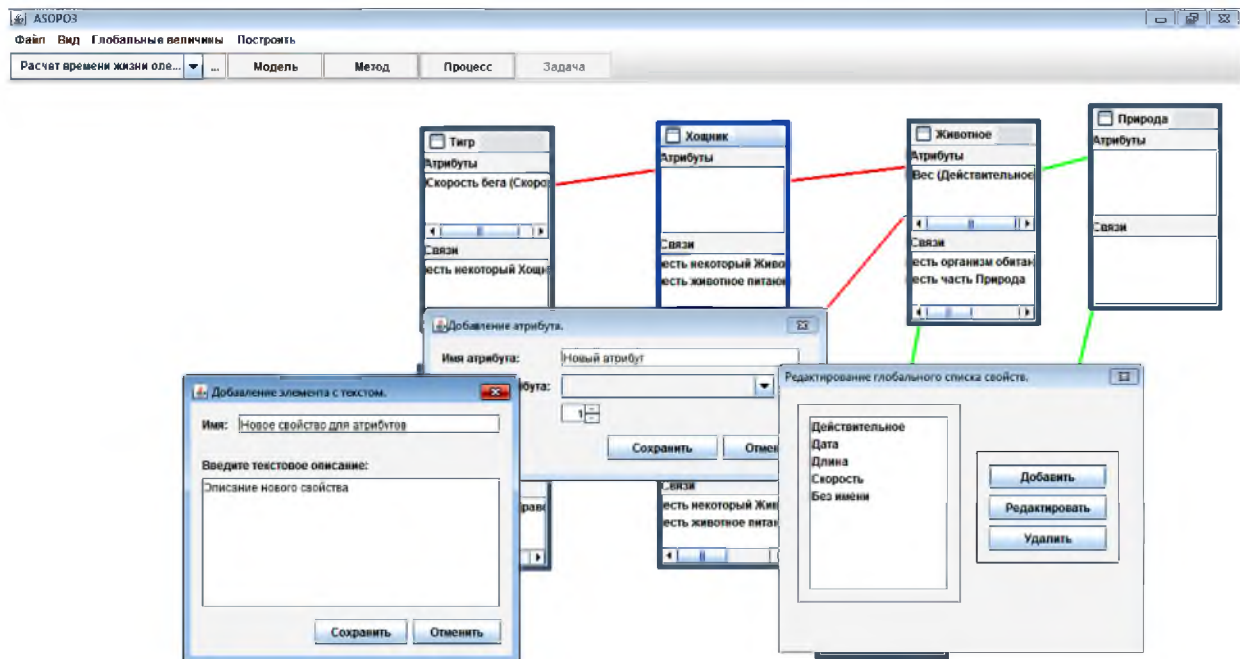


Рис. 2

СЛЕДОВАТЬ\_ВМЕСТЕ\_С – СВЯЗЬ типа взаимодействия;

ТРЕБОВАТЬ – СВЯЗЬ структурного типа (интерфейсная).

АТРИБУТ := {ИМЯ, СВОЙСТВО, ПАРАМЕТР, ХАРАКТЕРИСТИКА},

где ИМЯ – ЕСТЬ идентификатор, обеспечивающий возможность хранения и извлечения сведений об ОБЪЕКТЕ, хранящихся в базе ДАННЫХ (фактов);

СВОЙСТВО – ЕСТЬ тип АТРИБУТА, специфицирующий ОБЪЕКТ либо класс ОБЪЕКТОВ;

ПАРАМЕТР – ЕСТЬ количественная величина, характеризующая свойство ОБЪЕКТА либо класса ОБЪЕКТОВ;

ХАРАКТЕРИСТИКА (снм. показатель) – ЕСТЬ выходной ПАРАМЕТР СИСТЕМЫ.

ПЗМ строится как конкретизация и уточнение определений компонентов ПНМ.

Таким образом, ПНМ представляется тремя компонентами – классификационным, структурным и тезаурусом. Перейдем к рассмотрению системы визуализации элементов трехуровневой модели предметной области. ПН- и ПЗ-уровни

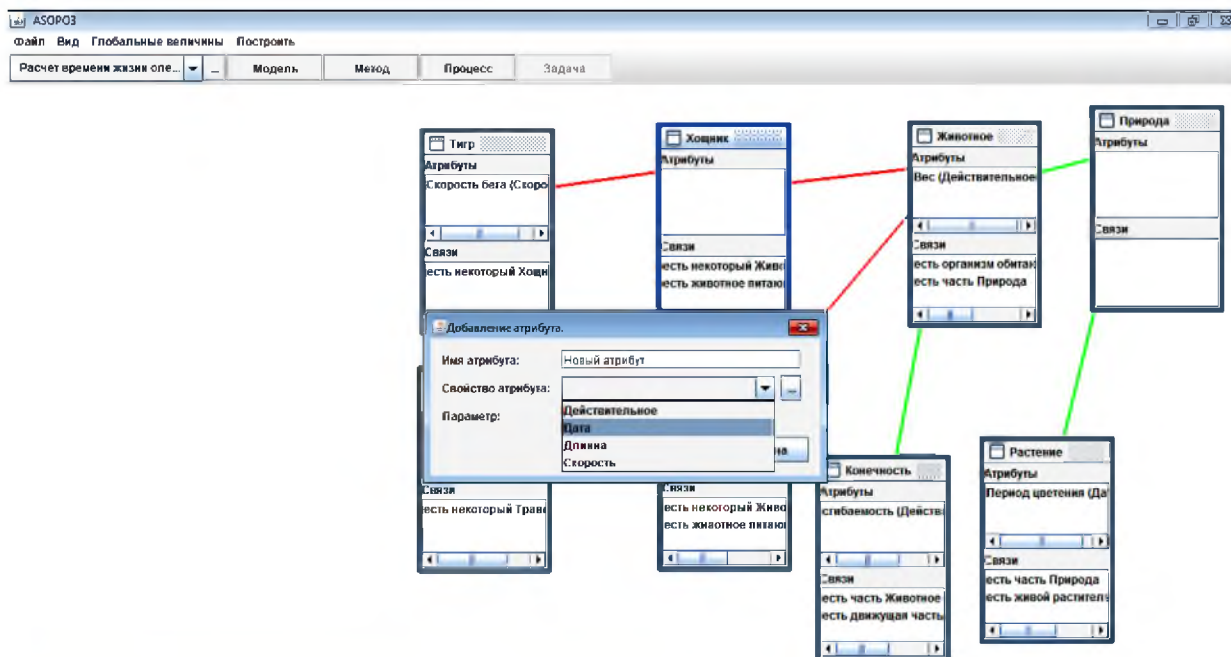


Рис. 3

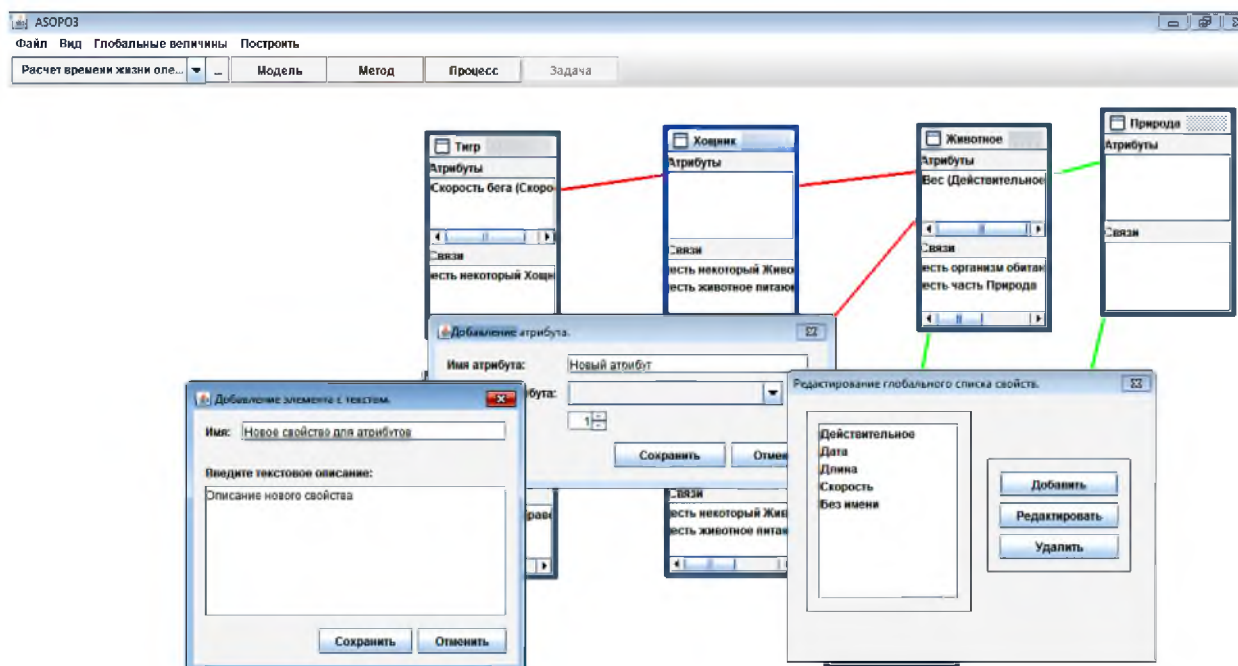


Рис. 4

модели формируются посредством нового улучшенного интерфейса программы [2], который позволяет создавать, редактировать и представлять модель предметной области.

Для создания новых объектов предметной области разработана форма редактирования объектов, позволяющая создавать, удалять и перемещать по рабочему пространству форму (окно) объекта.

Для добавления атрибута спроектировано окно (форма) объекта предметной области, которое состоит из трех частей (рис. 1):

- 1) заголовок – отображение названия объекта предметной области;
- 2) поле с атрибутами – поле, отображающее атрибуты объекта, позволяющее добавлять, удалять и редактировать их;
- 3) поле связей – поле, отображающее связи объекта. При добавлении атрибута есть возможность задать ему необходимые свойства, добавлять, удалять и редактировать связи.

Как видно из рис. 1, введен блок «новый объект», в который можно вносить необходимые ат-

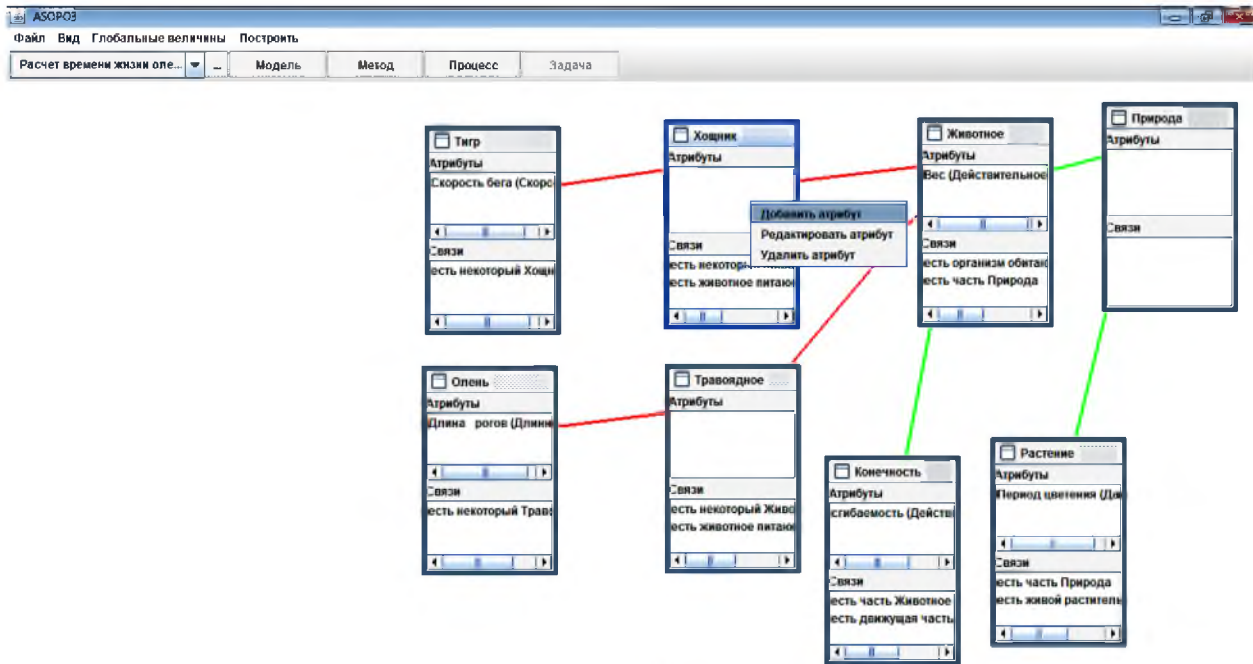


Рис. 5

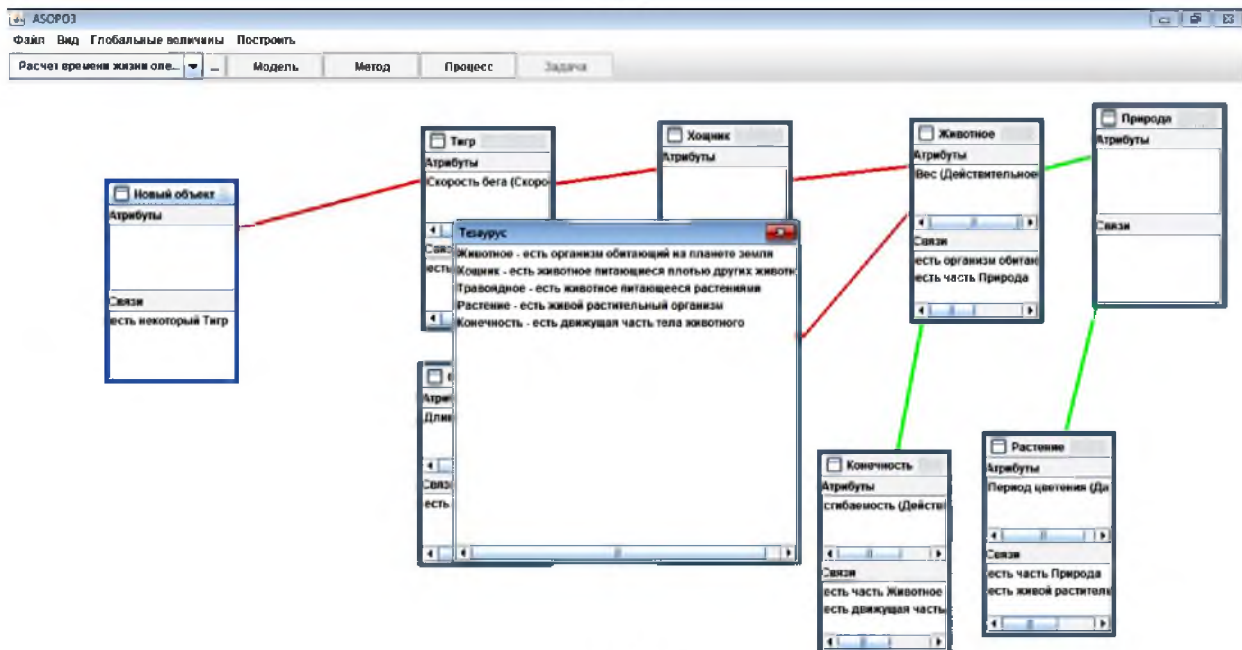


Рис. 6

рибуты и связи. Посредством графического интерфейса можно создавать неограниченное количество объектов системы, а также редактировать их атрибуты и связи согласно модели данных. При добавлении атрибута есть возможность задать ему необходимые свойства (рис. 2).

На рис. 2 приведена форма для редактирования атрибута. Можно выбрать одно из существующих свойств либо создать новое, причем можно задать описание нового свойства.

При создании нового объекта (рис. 3) можно сразу определить его связи, которые отображают-

ся цветом (классификационные → красным, структурные – зеленым).

Назначение свойств атрибутам приведено на рис. 3, а их описание – на рис. 4.

На рис. 5 приведен пример редактирования атрибутов.

Как видно из описания формализма, его структура содержит 3 основных компонента: ось классификационного типа, ось структурного типа и тезаурус. В системе предусмотрено окно для формирования тезауруса.



В автоматизированной системе реализована процедура формирования тезауруса, а также дерева связей. Подобные функции вызываются посредством графического интерфейса и выводятся в отдельных модальных диалогах для просмотра результата построения. Для вызова построителя

тезауруса необходимо выполнить команду Построить -> Тезаурус.> (рис. 6).

Кроме построения тезауруса можно вызвать одну из команд построения дерева связей для конкретного типа связей, например для связи типа «ЕСТЬ\_НЕКОТОРЫЙ».

Вывод тезауруса показан на экране.

## СПИСОК ЛИТЕРАТУРЫ

1. Шеховцов О. И. Информационная технология предпроектного обследования производственных систем // Изв. ГЭТУ. «Информационные технологии и

технических и организационных системах». 1997. Вып. 514. С. 95–103.

2. Давыдов С., Ефимов А. IntelliJ IDEA. Профессиональное программирование на Java. СПб., 2005.

V. P. Permyakov, O. I. Shekhovtsov  
Saint Petersburg Electrotechnical University «LETI»

## THE VISUALIZATION SYSTEM IS THE ONTOLOGICAL MODEL OF THE SUBJECT AREA

*Describes a system for the visualization of ontological models of knowledge representation: the System is implemented using object oriented programming in Java using the graphical library Swing. First provides basic information about the structure of the ontological formalism, and then the screenshots reflecting the view of the components of the model. The development of PI and PZ the levels of the model is made by new improved interface.*

*Examples Windows to create domain objects, their attributes, and their properties and characteristics, and define relationships between objects. Relationships are defined by straight lines of different colors. Through a graphical interface, you can create an unlimited number of system objects and edit their attributes and relationships according to the data model.*

**Visualization system, ontological model, domain-independent model, problem-oriented model, concepts, connection**

УДК 004.056

Е. С. Новикова, Я. А. Бекенева, А. В. Шоров  
Санкт-Петербургский государственный электротехнический  
университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

## Исследование методов корреляции событий безопасности для обеспечения безопасности облачных вычислительных сред

*Представлены результаты анализа процесса корреляции событий для решения задач информационной безопасности, представлено описание основных его этапов и операций, выполняемых над событиями. Рассмотрены возможные способы построения компонента корреляции событий, описаны их достоинства и недостатки.*

**Безопасность облачных технологий, корреляция событий безопасности, операции над событиями, архитектура модуля корреляции событий**

Системы облачных вычислений привлекательны для использования в бизнес-процессах благодаря ряду серьезных преимуществ, таких,

как легкое конфигурирование вычислительных устройств под необходимые нужды организации, гибкость и эластичность предоставляемых облач-