

УДК 004.822:514

СЕМАНТИЧЕСКАЯ ТЕХНОЛОГИЯ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ**В. В. Голенков, Н. А. Гулякина, И. Т. Давыденко, Д. Н. Корончик, Д. В. Шункевич**

Белорусский государственный университет информатики и радиоэлектроники

ул. П. Бровки, 6, г. Минск, 220013, Республика Беларусь. E-mail: golenkov.v.v@gmail.com, shu.dv@tut.by

Рассматриваются принципы построения технологии проектирования интеллектуальных систем, ориентированной на семантическое представление знаний, расширение контингента разработчиков и сокращение сроков проектирования. Особое внимание уделяется современным проблемам в области проектирования интеллектуальных систем, таких, как высокая степень зависимости технологий от конкретной платформы, закрытости и недоступности таких технологий, отсутствия возможности интеграции различных подходов к представлению и обработке знаний в рамках отдельной технологии и многим другим. С целью решения указанных проблем разрабатывается технология проектирования интеллектуальных систем, ориентированная на накопление и систематизацию многократно используемых компонентов различного рода.

Ключевые слова: искусственный интеллект, базы знаний, семантические технологии, открытый проект, многоагентные системы.

СЕМАНТИЧНА ТЕХНОЛОГІЯ ПРОЕКТУВАННЯ ІНТЕЛЛЕКТУАЛЬНИХ СИСТЕМ**В. В. Голенков, Н. О. Гулякіна, І. Т. Давиденко, Д. М. Корончик, Д. В. Шункевич**

Білоруський державний університет інформатики і радіоелектроніки

вул. П. Бровки, 6, м. Мінськ, 220013, Республіка Білорусь. E-mail: golenkov.v.v@gmail.com, shu.dv@tut.by

Розглядаються принципи побудови технології проектування інтелектуальних систем, орієнтованої на семантичне уявлення знань, розширення контингенту розробників і скорочення строків проектування. Особлива увага приділяється сучасним проблемам в області проектування інтелектуальних систем, таким як високий ступінь залежності технологій від конкретної платформи, замкнутості та недоступності таких технологій, відсутність можливості інтеграції різних підходів до представлення й обробки знань у рамках окремої технології та іншим. З метою вирішення вказаних проблем розробляється технологія проектування інтелектуальних систем, орієнтована на накопичення та систематизацію багатократно використовуваних компонентів різного роду.

Ключові слова: штучний інтелект, бази знань, семантичні технології, відкритий проект, багатоагентні системи.

АКТУАЛЬНОСТЬ РАБОТЫ. Основной тезис данной работы состоит в том, что результатом исследований в области искусственного интеллекта должна являться не столько разработка конкретных интеллектуальных систем, обладающих высокими интеллектуальными возможностями, сколько *разработка технологий*, позволяющих быстро и в большом количестве порождать самые разнообразные интеллектуальные системы, имеющие большую практическую ценность. Очевидно, что составляющими таких технологий являются:

- формальная теория интеллектуальных систем (формальное описание того, как они устроены);
- методы проектирования интеллектуальных систем;
- инструментальные средства (средства автоматизации проектирования интеллектуальных систем);
- средства информационной поддержки (информационного обслуживания) разработчиков интеллектуальных систем;
- средства компьютерной поддержки управления коллективной разработкой интеллектуальных систем.

Анализ современных технологий проектирования интеллектуальных систем показывает, что:

- технологии искусственного интеллекта *не ориентированы на широкий круг разработчиков* интеллектуальных систем и, следовательно, не получили массового распространения;
- *велики сроки разработки* интеллектуальных

систем и велика трудоемкость их сопровождения;

– *высока степень зависимости технологий искусственного интеллекта от платформ*, на которых они реализованы, что приводит к существенным изменениям технологий при переходе на новые платформы;

– для эффективной реализации даже существующих моделей представления знаний и моделей решения трудно формализуемых задач *современные компьютеры оказываются плохо приспособленными*, что требует разработки принципиально новых компьютеров;

– отсутствуют подходы, позволяющие на некоторой универсальной основе *интегрировать научные и практические результаты* в области искусственного интеллекта, что порождает высокую степень дублирования результатов. В частности, высокая *трудоемкость интеграции* различных моделей представления информации, моделей обработки информации, моделей решения задач и, следовательно, различных интеллектуальных компьютерных систем.

Искусственный интеллект является междисциплинарной научной дисциплиной. Этим обусловлен большой её потенциал, так как на стыках научных направлений рождаются сильные результаты. Но этим же обусловлены и большие трудности, т.к. развитие искусственного интеллекта требует глубокого взаимопонимания и сотрудничества исследователей, имеющих разный стиль мышления, разный подход к объекту и предмету исследования, разный

менталитет, разные целевые установки и традиции. Современный этап развития искусственного интеллекта остро нуждается в преодолении указанных трудностей.

Важнейшей задачей искусственного интеллекта в настоящее время является построение общей комплексной теории интеллектуальных систем, в рамках которой бы сочетались самые разные направления искусственного интеллекта:

- и теория представления знаний, и теория решения задач (в том числе различные исчисления, эвристики, стратегии),
- теория программ (процедурных, декларативных, параллельных, последовательных),
- архитектуры интеллектуальных систем, в том числе детализированные до уровня аппаратной поддержки,
- теория интеллектуальных пользовательских интерфейсов,
- компьютерная лингвистика.

Эпицентром развития искусственного интеллекта становится не столько разработка отдельных его направлений, сколько их глубокая *семантическая интеграция*, целью которой должна быть не только общая теория интеллектуальных систем, но и общая, доступная технология их комплексного проектирования.

Цель работы – предлагается подход к созданию технологии проектирования интеллектуальных систем, направленной на устранение рассмотренных выше проблем.

МАТЕРИАЛ И РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЙ. В первую очередь предполагается использовать опыт наиболее продвинутых в этом плане технологий, в частности, технологии проектирования микросхем.

Для того, чтобы аналогичным образом построить технологию проектирования интеллектуальных систем, необходимо

- 1) создать языковые средства полного унифицированного формального описания интеллектуальных систем;
- 2) четко отделить разработку полного унифицированного формального описания проектируемой интеллектуальной системы от разработки различных вариантов интерпретации таких формальных описаний интеллектуальных систем;
- 3) создать библиотеку формальных описаний типовых (многократно используемых) компонентов интеллектуальных систем. Но для того, чтобы такая библиотека была создана, необходимо обеспечить *интегрируемость* (семантическую совместимость) указанных компонентов интеллектуальных систем.

В качестве формальной основы проектируемых интеллектуальных систем предполагается использовать графодинамические модели специального вида, ориентированные на *параллельную и асинхронную* обработку информации.

Графодинамическая модель параллельной асинхронной обработки информации, которую будем

также называть *графодинамической параллельной асинхронной машиной*, трактуется нами как абстрактная *многоагентная система*, состоящая из

- абстрактной *графодинамической памяти*, в которой хранятся обрабатываемые графовые структуры;
- коллектива агентов*, работающих над *общей для них графодинамической памятью* и обменивающихся информацией только через эту память (в том числе и для координации своих действий).

Графодинамическая память носит реконфигурируемый, структурно перестраиваемый характер, поскольку процесс обработки графовых структур в конечном счете сводится к генерации и удалению различных элементов графовых структур, а также к генерации и удалению пар инцидентности между этими элементами. Другими словами, процесс обработки информации в графодинамической памяти сводится не только к изменению состояния элементов памяти, но и к изменению конфигурации связей между ними.

Агенты могут работать параллельно, если одновременно возникают условия инициирования агентов.

В качестве непосредственно основы абстрактных логико-семантических моделей интеллектуальных систем используются графодинамические модели специального вида – семантические модели представления и обработки знаний, в основе которых лежат *семантические сети* [1–6].

Фактически, речь идет о создании формальных средств описания семантики различных видов знаний и формальных средств описания обработки знаний на семантическом уровне.

Далее необходимо определить структуру унифицированных семантических сетей, обеспечивающих представление самых различных видов знаний.

Это предполагает разработку соответствующего стандарта, выделяющего из всего многообразия абстрактных языков семантических сетей определенный базовый универсальный язык семантических сетей, который мы назвали SC-кодом (Semantic Computer code). В основе данного языка лежит теоретико-множественная интерпретация семантических сетей, на базе которых строится сам указанный язык.

SC-код представляет собой ядро *универсального открытого языка семантических сетей*, являющегося результатом интеграции всевозможных языков семантических сетей, построенных на основе SC-кода, и задаваемого

- фиксированным алфавитом (алфавитом SC-кода);
- постоянно расширяемым (открытым) семейством ключевых узлов, в состав которого входят все ключевые узлы всех интегрируемых языков.

SC-код представляет собой *единство языка и метаязыка*. Так, например, в виде sc-конструкций можно описать синтаксис, семантику и онтологию SC-кода. С формальной точки зрения SC-код можно трактовать как метаязык базовой семантической спецификации sc-элементов с помощью специального набора ключевых узлов SC-кода.

Технология проектирования баз данных. Семантическая структура базы знаний каждой интеллектуальной системы трактуется как отражение иерархической системы взаимосвязанных друг с другом предметных областей, представляемых в базе знаний. Это предполагает

- уточнение понятия предметной области;
- разработку языковых средств описания структуры предметных областей с помощью унифицированных семантических сетей;
- разработку языковых средств описания типологии предметных областей и различных видов связей между ними.

Структуризация базы знаний, выделение в ней различных связанных между собой подструктур необходимы по целому ряду причин. В частности, это необходимо для дидактических целей (человеку, усваивающему некоторые знания, желательно иметь своего рода оглавление или «карту» этих знаний, что позволяет планировать их усвоение и рассматривать их с различной степенью детализации), а также для организации распределения работ по проектированию баз знаний (когда разным исполнителям поручается разработка разных фрагментов базы знаний, имеющих достаточно четкие границы).

Можно говорить о достаточно богатой типологии предметных областей.

Заметим, что некоторым из классов предметных областей может соответствовать одинаковый (унифицированный, фиксируемый) набор используемых в них ключевых понятий. Унификация (стандартизация) таких наборов понятий является важнейшим средством более глубокой семантической совместимости (интегрируемости) различных фрагментов базы знаний. Результатом такой унификации фактически является разработка средств SC-кода, ориентированных на представление предметных областей соответствующего класса. Такие языковые средства будем называть специализированным *sc-языком*.

Таким образом, *SC-код является ядром целого семейства самых различных sc-языков*, ориентированных на описание различных классов предметных областей, в каждый из которых входят предметные области с разными множествами исследуемых объектов, но с одинаковыми предметами исследования.

Каждому такому специализированному *sc-языку* ставится в соответствие множество ключевых узлов, обозначающих различные классы исследуемых объектов, различные отношения и алгебраические операции, заданные на множестве исследуемых объектов.

Технология проектирования машин обработки знаний. Для описания способов решения задач и поведения агентов над общей графодинамической памятью используются *графовые языки программирования*, которые ориентированы на обработку унифицированных семантических сетей и программы которых сами являются унифицированными семантическими сетями.

Если все используемые в интеллектуальной сис-

теме графовые языки программирования привести к общему унифицированному стандарту – к SC-коду (это требует представления в виде *sc-текстов* не только самих программ, но и обрабатываемых ими данных), то можно достаточно эффективно решать проблему формализации семантической совместимости программ, написанных не только на одном, но и на разных языках программирования.

Из всех *sc-языков* программирования выделен *базовый графовый язык программирования*, ориентированный на описание агентов, работающих над общей графодинамической памятью, в которой хранятся и обрабатываются *унифицированные* семантические сети.

Выделение базового *sc-языка* программирования предназначено для унификации формального описания поведения агентов, работающих над общей графодинамической памятью.

Такой базовый *sc-язык* программирования будем называть *языком SCP* (Semantic Code Programming), а написанные на нем программы будем называть *scp-программами*.

Перечислим основные особенности языка SCP:

- язык SCP относится к классу графовых языков программирования;
- язык SCP ориентирован на обработку *унифицированных семантических сетей* (*sc-текстов*), хранимых в семантической памяти;
- программы языка SCP представляются также в виде унифицированных семантических сетей (*sc-текстов*), т. е. язык SCP принадлежит классу *sc-языков*;
- язык SCP ориентирован на описание *параллельной асинхронной обработки* *sc-текстов*, хранимых в семантической памяти;
- язык SCP использует *ассоциативный доступ* к фрагментам обрабатываемых *sc-текстов*;
- язык SCP является процедурным языком программирования низкого уровня, предназначенным для описания поведения агентов, работающих над семантической памятью;
- уникальной особенностью языка SCP является то, что на нем можно писать *реконфигурируемые программы*, т.е. программы, которые в процессе своего выполнения могут изменять сами себя (удалять или порождать операторы, корректировать порядок их выполнения и т. п.). Такая особенность языка SCP обусловлена не только тем, что *scp-программы* и обрабатываемые ими данные хранятся в общей памяти, но и тем, что они принадлежат одному и тому же, базовому языку (SC-коду), имеющему четко заданную семантическую интерпретацию.

На основе унифицированных семантических сетей уточняется понятие *унифицированной модели обработки информации*, а также понятие унифицированной модели решения задач.

Все указанные абстрактные модели будем называть *sc-моделями обработки знаний* или *sc-машинами*, поскольку в основе их лежит использование SC-кода. Каждая такая модель (*sc-машина*)

представляет собой многоагентную систему, состоящую:

1) из графодинамической памяти, в которой хранятся и обрабатываются тексты SC-кода – такую память будем называть *sc-памятью*;

2) из коллектива агентов, работающих над общей для них *sc-памятью* и взаимодействующих между собой только через эту память – такие агенты будем называть *sc-агентами*.

Каждый *sc-агент* реагирует на соответствующий ему класс ситуаций и/или событий, происходящих в *sc-памяти*, и осуществляет определенное преобразование *sc-текста*, находящегося в семантической окрестности обрабатываемой ситуации и/или события. Типология *sc-агентов* достаточно богата, и в данной работе рассматриваться более подробно не будет.

В понятии *sc-машины* набор агентов не задается, т.е. могут существовать разные *sc-машины* с разным набором *sc-агентов*. Несколько разных *sc-машин* можно *интегрировать*. С формальной точки зрения это сделать не очень сложно:

1) интегрировать *sc-текст*, описывающий текущее состояние взаимодействия *sc-агентов* одной *sc-машины*, с аналогичным *sc-текстом* другой *sc-машины*;

2) полученный интегрированный *sc-текст* поместить в *sc-память* интегрированной *sc-машины*;

3) в интегрированную *sc-машину* включить все *sc-агенты* первой интегрируемой *sc-машины* и все *sc-агенты* второй интегрируемой *sc-машины*.

На основе унифицированных семантических сетей (т.е. на основе SC-кода) обеспечивается построение *унифицированных семантических моделей информационного поиска* (унифицированных семантических моделей ассоциативного доступа к информации, хранимой в семантической памяти).

Унифицированная семантическая модель информационного поиска, которую будем называть *sc-моделью информационного поиска*, включает в себя:

1) *SC-язык вопросов*;

2) *SC-язык оформления ответов*;

3) Семейство информационно-поисковых *sc-агентов*, каждый из которых реагирует на соответствующий ему тип *sc-вопроса* и выполняет соответствующую поисковую процедуру в *sc-памяти*.

На основе унифицированных семантических сетей строятся также *унифицированные семантические модели интеграции знаний*, которые предполагается использовать (1) как основу процесса приобретения интеллектуальной системой новых знаний, как со стороны конечных пользователей, так и со стороны разработчиков; (2) как основу интеграции программ и различных семантических моделей расширения задач; (3) как основу интеграции абстрактных логико-семантических моделей интеллектуальных систем.

При интеграции интеллектуальных систем происходит приобретение нового качества "на стыке" интегрируемых систем, когда для решения некоторых задач одна часть необходимых знаний и/или

умений находится в одной интегрируемой системе, а другая часть – в другой системе.

На основе предлагаемой модели в рамках проектируемой интеллектуальной системы обеспечивается использование не только самых различных видов знаний, но и самых *различных моделей и стратегий решения задач*, реализуемых в виде соответствующих коллективов *sc-агентов* над общей *sc-памятью*.

Технология проектирования пользовательских интерфейсов. В качестве основы организации графического пользовательского интерфейса используется язык унифицированного визуального представления абстрактных унифицированных семантических сетей в виде, близком к изоморфному.

Указанный язык графического изображения *sc-текстов* назван SCg-кодом (Semantic Code graphical). С помощью SCg-кода осуществляется отображение на экране не только пользовательских сообщений, адресуемых системе, и не только сообщений, адресуемых пользователю, но и всей остальной информации, необходимой для организации работы пользователя (прежде всего – это элементы управления интерфейсом). Такая унификация отображаемой пользователю информации дает возможность организовать взаимодействие пользователя с help-системой точно так же, как и его взаимодействие с основной (предметной) системой.

Заметим также, что кроме SCg-кода для внешнего представления абстрактных унифицированных семантических сетей используются также и другие языки:

– SCs-код, обеспечивающий представление унифицированных абстрактных семантических сетей (*sc-текстов*) в виде, близком к традиционным текстам;

– SCn-код, обеспечивающий гипертекстовое представление абстрактных *sc-текстов*, предназначенное для оформления исходных текстов баз знаний.

Более подробно различные языки внешнего представления абстрактных *sc-текстов* вместе с большим количеством примеров рассмотрены в работах [7].

Пользовательский интерфейс интеллектуальной системы, построенной на основе предлагаемой технологии, рассматривается как специализированная интеллектуальная система, построенная по той же технологии и предназначенная для трансляции адресуемых пользователю сообщений с внутреннего абстрактного семантического языка представления знаний (SC-кода) на тот или иной внешний язык, тексты которого отображаются пользователю в удобном для него виде, а также для трансляции пользовательских сообщений с внешнего языка внутренний семантический язык интеллектуальной системы (т.е. в SC-код).

Принципы проектирования интеллектуальных систем. В целях ускорения процесса проектирования интеллектуальных систем создать общую библиотеку многократно используемых семантически совместимых компонентов интеллектуальных сис-

тем, на основі котрої розробити методикy модульного (компонентного, сборочного) проектування інтелектуальних систем.

В указанній бібліотеці можна виділити наступні розділи (частинні бібліотеки):

- бібліотека багаторазово використовуваних компонентів баз знань;
- бібліотека компонентів семантичних моделей інформаційного пошуку;
- бібліотека компонентів семантичних моделей інтеграції знань і машинної обробки знань;
- бібліотека інтерпретаторів програм, відповідних різним мовам програмування;
- бібліотека різних стратегій рішення задач, різних моделей рішення задач і агентів, що входять до складу таких моделей;
- бібліотека компонентів користувацьких інтерфейсів.

Всі компоненти, що включаються до складу загальної бібліотеки компонентів інтелектуальних систем, оформляються як компоненти інтелектуальної власності (*intellectual property*), тому будемо їх також називати ір-компонентами.

Для забезпечення семантичної сумісності таких компонентів інтелектуальних систем, що є єдиними семантичними моделями, необхідно (1) узгодити семантику (смысл) всіх використовуваних ключових вузлів і (2) узгодити глобальні ідентифікатори ключових вузлів, використовуваних в різних компонентах. Після цього інтеграція всіх компонентів, що входять до складу бібліотеки, і в будь-яких комбінаціях здійснюється автоматично, без втручання розробника.

Немаловажним є забезпечення платформної незалежності технології і її окремих компонентів. Для максимальної платформної незалежності технології необхідно забезпечити чітке розділення процесу проектування формального опису логіко-семантичної моделі розроблюваної інтелектуальної системи від процесу реалізації (інтерпретації) цієї моделі на тій чи іншій платформі.

Якщо кожній інтелектуальній системі відповідає своя унікальна логіко-семантична модель, то кожен інтерпретатор абстрактних логіко-семантичних моделей інтелектуальних систем повинен забезпечувати інтерпретацію всього класу таких моделей, а в ідеалі – інтерпретацію будь-якої такої моделі. Тому розробка вказаних інтерпретаторів може здійснюватися абсолютно незалежно від розробки логіко-семантичних моделей конкретних інтелектуальних систем.

Таким чином, в мові SC-коду і мові SCP ми маємо стандарт повного формального опису логіко-семантичних моделей інтелектуальних систем, що забезпечує незалежність проектування абстрактних логіко-семантичних моделей конкретних інтелектуальних систем від розробки різних варіантів їх реалізації.

При розробці конкретних інтелектуальних

систем використовується *методика поетапного еволюційного проектування* інтелектуальних систем.

Вказана методика передбачає:

- швидке проектування;
- швидше введення в експлуатацію перших версій проектуваної системи з мінімальними, але практично корисними можливостями;
- еволюційне поетапне удосконалення проектуваної інтелектуальної системи шляхом її розширення новими знаннями і навичками безпосередньо в ході експлуатації інтелектуальної системи і активним залученням її кінцевих користувачів.

З формальної точки зору проектування уніфікованої логіко-семантичної моделі (sc-моделі) інтелектуальної системи в кінцевому рахунок зводиться до проектуванню sc-моделі бази знань цієї інтелектуальної системи, оскільки scr-програми, що описують поведінку sc-агентів, можна розглядати як частину бази знань.

Інтелектуальна метасистема. Представлена технологія реалізується як *інтелектуальна метасистема, орієнтована на підтримку проектування інтелектуальних систем*, побудована за тими ж самими принципами (т.е. за тими ж технологіями), що і інтелектуальні системи, розроблювані на її основі.

Вказана інтелектуальна система повинна включати в себе:

- теорію (принципи побудови) проектуваних інтелектуальних систем, що входять до складу бази знань метасистеми;
- бібліотеку типових багаторазово використовуваних компонентів (ір-компонентів) інтелектуальних систем, що входять до складу бази знань розроблюваної метасистеми;
- засоби автоматизації синтезу, аналізу і імітаційного моделювання проектуваних інтелектуальних систем і їх компонентів;
- інтелектуальну help-систему, що є підсистемою розроблюваної інтелектуальної метасистеми орієнтованою на інформаційне обслуговування і навчання розробників інтелектуальних систем;
- методику проектування інтелектуальних систем, що оформляється як частина бази знань метасистеми;
- методику навчання проектуванню інтелектуальної системи, що також є частиною бази знань метасистеми;
- інтелектуальну підсистему управління проектуванням самої метасистеми;
- інтелектуальну підсистему управління інформаційною безпекою метасистеми;
- сімейство різних варіантів реалізації інтерпретаторів уніфікованих абстрактних логіко-семантичних моделей інтелектуальних систем.

Враховуючи розроблювані вище принципи по-

строения предлагаемой нами технологии, она названа Открытой Семантической Технологией проектирования Интеллектуальных Систем (Open Semantic Technology for Intelligent Systems – OSTIS). Можно также её было бы назвать *SC-технологией*, поскольку основой этой технологии является SC-код. Соответственно этому, интеллектуальную метасистему, ориентированную на поддержку проектирования интеллектуальных систем, будем называть *метасистемой OSTIS*.

ВЫВОДЫ. В заключение подытожим ключевые вопросы, рассматриваемые в данной работе.

Ключевыми проблемами, решение которых лежит в основе предлагаемой технологии являются:

- обеспечение семантической совместимости (интегрируемости) различных моделей представления и обработки знаний;
- создание общей теории абстрактных семантических моделей интеллектуальных систем, не противопоставляя, а интегрируя самые различные подходы;
- обеспечение максимальной возможной независимости интеллектуальных систем от многообразия вариантов и платформ их технической реализации (в т. ч. и от будущих компьютеров, специально ориентированных на аппаратную поддержку обработки

знаний).

ЛИТЕРАТУРА

1. Кузнецов И.П. Семантические представления. – М.: Наука, 1986. – 268 с.
2. Лозовский В.С. Семантические сети // Представление знаний в человеко-машинных и робототехнических системах. – М.: ВИНТИ, 1984. – С. 84–121.
3. Плесневич Г.С. Представление знаний в ассоциативных сетях // Изв. АН СССР. Техн. кибернет. – 1982. – № 5. – С. 6–22.
4. Скороходько Э.Ф. Семантические сети и автоматическая обработка текста. – К.: Наукова думка, 1983. – 218 с.
5. Шенк Р. Обработка концептуальной информации. – М.: Энергия, 1980. – 360 с.
6. Conceptual Graphs / John F. Sowa, F. van Harmelen, V. Lifschitz, B. Porter // eds., Handbook of Knowledge Representation, Elsevier, 2008. – PP. 213–237.
7. Голенков В.В. Представление и обработка знаний в графодинамических ассоциативных машинах. – Мн.: БГУИР, 2001. – 135 с.

SEMANTIC TECHNOLOGY FOR INTELLIGENT SYSTEMS DESIGN

V. Golenkov, N. Guliakina, I. Davydenko, D. Koronchik, D. Shunkevich

Belarusian state university of informatics and radioelectronics

vul. P. Brovki, 6, Minsk, 220013, Belarus. E-mail: golenkov.v.v@gmail.com, shu.dv@tut.by

The article is devoted to the principles, which underlie semantic technology for intelligent systems design, oriented on the semantic knowledge representation, developers contingent extension and design terms reduction. Additional attention is paid to the modern problems in sphere of intelligent systems development, such as high level of platform dependence of concrete technologies, closeness and inaccessibility of such technologies, impossibility of integration of different approaches to knowledge representation and processing inside one technology and many others. With goal to solve given problems the technology for intelligent systems design is being developed, which is oriented on the accumulation and systematization of reusable components.

Key words: artificial intelligence, knowledge bases, semantic technologies, open project, multiagent systems.

REFERENCES

1. Kuznetsov, I.P. (1986) *Semanticheskie predstavleniya* [Semantic representations], Nauka, Moscow, Russian.
2. Lozovskiy, V.S. (1984) “Semantic networks”, *Knowledge representation in human-machine and robotics systems*, VINITI, Moscow, pp. 84–121.
3. Plesnevich, G.S. (1982) “Knowledge representation in associative networks”, *Izv. AS USSR. Tech. cybernetics*, no. 5, pp. 6–22.
4. Skorohodko, E.F. (1983) “Semantik networks and automated text processing”, *Naukova dumka*, Kiev, Ukraine.
5. Shenk, R. (1980) *Obrabotka kontseptualnoy informatsii* [Conceptual information processing], Energia, Moscow, Russia.
6. Sowa, J. (2008) *Conceptual Graphs*, *Handbook of Knowledge Representation*, Elsevier, pp. 213–237.
7. Golenkov, V.V. (2001) *Predstavlenie i obrabotka znaniy v grafodinamicheskikh assotsiativnykh mashinakh* [Knowledge representation and processing in graphodynamical associative machines], BSUIR, Minsk, Belorussia.

Стаття надійшла 30.10.2014.