

МАССОВАЯ ТЕХНОЛОГИЯ МОДУЛЬНОГО ПРОЕКТИРОВАНИЯ СЕМАНТИЧЕСКИ СОВМЕСТИМЫХ КОМПЬЮТЕРНЫХ СИСТЕМ РАЗЛИЧНОГО УРОВНЯ ИНТЕЛЛЕКТУАЛЬНОСТИ

Голенков В. В., Гулякина Н. А., Колб Д. Г.

Белорусский государственный университет информатики и радиоэлектроники
ул. П. Бровки, 6, г. Минск, 220013, Беларусь
тел.: +375-17-2932324, e-mail: golen@bsuir.by

Аннотация — Рассматриваются принципы построения технологий проектирования интеллектуальных систем, ориентированных на семантическое представление знаний, расширение контингента разработчиков и сокращение сроков проектирования.

I. Введение

Современное состояние в области проектирования компьютерных систем напоминает вавилонское столпотворение самых различных подходов, моделей, методов, средств, платформ [1]. Технологии искусственного интеллекта не ориентированы на широкий круг разработчиков интеллектуальных компьютерных систем и, следовательно, не получили массового распространения, на наш взгляд, по следующим причинам:

- 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;
- отсутствуют подходы, позволяющие на некоторой универсальной основе интегрировать научные и практические результаты в области искусственного интеллекта, что порождает высокую степень дублирования результатов;
- для эффективной реализации существующих моделей представления знаний и моделей решения трудно формализуемых задач современные компьютеры оказываются плохо приспособленными, что требует разработки принципиально новых компьютеров.

II. Основная часть

Нами предлагается следующий **подход**, направленный на устранение указанных выше недостатков:

- разработать принципы построения таких логико-семантических моделей любых компьютерных систем (в том числе и интеллектуальных), которые:
 - носили бы "рафинированный" характер, т.е. были бы лишены каких бы то ни было деталей, обусловленных тем или иным способом их технической реализации, были бы инвариантны многообразию таких способов;
 - обеспечили бы максимально возможное упрощение решения проблемы совместности и интегрируемости таких логико-семантических моделей;
- проектирование компьютерных систем (в т.ч. и интеллектуальных) осуществлять на основе унифицированных логико-семантических моделей, рассматривая разработку логико-семантической модели проектируемой системы как первый этап ее проектирования;

- обеспечить модульную (компонентную, крупноблочную) разработку логико-семантических моделей компьютерных систем на основе библиотек совместимых типовых многократно используемых компонентов (онтологий, логических операций и т.д.);

- рассматривать различные варианты технической реализации компьютерных систем как различные способы интерпретации унифицированных логико-семантических моделей компьютерных систем, что дает возможность разрабатывать такие интерпретаторы независимо от проектирования конкретных систем и включать эти интерпретаторы в состав среды проектирования;

- обеспечить полную совместимость средств проектирования с проектируемыми системами – среда проектирования строится как интеллектуальная система на основе унифицированных логико-семантических моделей;

- включить в состав среды проектирования компьютерных систем комплекс интеллектуальных help-систем, ориентированных на повышение квалификации разработчиков;

- ориентироваться на методику поэтапного эволюционного проектирования компьютерных систем на основе быстрого прототипирования.

В основе логико-семантических моделей лежит понятие семантической сети [2]. В рамках предлагаемого подхода это понятие требует уточнения. Будем считать, что **семантическая сеть** (G) — это знаковая конструкция (текст), представляющая собой математическую структуру, которая задается пятеркой $G = \langle V, C, I, M, K \rangle$, где V — множество вершин (первичных элементов), C — множество связей (вторичных элементов), I — семейство бинарных ориентированных отношений инцидентности, M — алфавит элементов семантической сети (набор меток, обеспечивающих синтаксическое выделение некоторых классов элементов), K — множество ключевых узлов.

При этом рассматриваемая математическая структура (G) должна удовлетворять следующим семантическим требованиям:

- вершины этой структуры должны быть знаками (обозначениями) различных описываемых объектов;

- связи этой математической структуры должны быть знаками (обозначениями) различных связей, связывающих между собой описываемые объекты, либо связей связывающих описываемые объекты с другими связями, либо связей, связывающих между собой различные связи;

- отношения инцидентности этой математической структуры должны быть знаками (обозначениями) различных ролей, которые выполняют различные описываемые объекты или связи в рамках тех связей, компонентами которых они являются;

- алфавит элементов этой математической структуры должен трактоваться как семейство знаков, каждый из которых обозначает соответствующий тип (класс) описываемых объектов и/или связей;

- в рамках рассматриваемой математической структуры среди перечисленных знаков (обозначений) должны отсутствовать синонимичные знаки, т.е. знаки, обозначающие одно и то же;

- в рамках рассматриваемой математической структуры среди перечисленных знаков (обозначений) должны отсутствовать омонимичные знаки, которые в разных контекстах, в разных обстоятельствах могут обозначать разные сущности.

Таким образом, семантическая сеть – это знаковая конструкция "рафинированного" вида, в которой нет ничего кроме знаков и связей инцидентности этих знаков. В частности, в семантической сети отсутствуют элементарные, незначащие фрагменты (символы), имена описываемых объектов, слова, из которых эти имена состоят, всевозможные разделители и ограничители, обеспечивающие структуризацию текста. В отличие от текстов традиционного вида, семантическая сеть имеет в общем случае нелинейный характер, поскольку каждый элемент семантической сети может быть инцидентен более чем двум другим элементам.

На основе понятия семантической сети вводится понятие **языка семантических сетей** в заданном алфавите и с заданным набором ключевых узлов.

Семантическую сеть G_b будем называть **бинарной семантической сетью**. Каждая небинарная связка семантической сети G в семантической сети G_b трактуется как множество связываемых им элементов семантической сети, связь которого с его элементами представляется явно – не в виде пар инцидентности, а в виде дополнительно вводимых связок принадлежности.

Почему в семантических сетях так важно "избавиться" от небинарных связок? Во-первых, бинарные связки легко изображать графически (в виде линий, каждая из которых соединяет графические изображения двух связываемых элементов семантической сети). Во-вторых, использование только бинарных связок существенно упрощает машинное кодирование семантических сетей и упрощает разработку специальной памяти для хранения семантических сетей.

Можно **минимизировать алфавит семантических сетей**. Метки элементов семантической сети без какого-либо изменения семантики этой семантической сети можно "превращать" (преобразовывать) в ее ключевые узлы. При этом семантическая интерпретация каждого такого ключевого узла будет совпадать с семантикой соответствующей преобразованной метки. Число меток семантической сети можно уменьшать ценой расширения множества её ключевых узлов. Вопрос в том, до какого предела это можно делать и как выглядит минимальный алфавит универсального языка семантических сетей.

Нами предлагается базовый язык семантических сетей, названный **SC-кодом** (Semantic Computer Code) [3]. Семантические сети, представленные в SC-коде, будем называть **sc-текстами**. Вершины, связки, метки и ключевые узлы sc-текстов будем называть **sc-элементами**. Вершины sc-текстов будем называть **sc-узлами**. Связки sc-текстов будем называть **sc-коннекторами**. Перечислим основные особенности SC-кода:

- SC-код является универсальным языком семантических сетей, обеспечивающим представление любых видов знаний;

- SC-код является языком бинарных семантических сетей, т.е. sc-коннекторы представляют собой бинарные связки либо неориентированного вида (**sc-ребра**), либо ориентированного вида (**sc-дуги**);

- SC-код имеет минимальный алфавит.

Алфавит SC-кода представляет собой набор из следующих пяти меток: метка sc-узла, метка sc-ссылки, метка sc-ребра, метка sc-дуги общего вида, метка sc-дуги основного вида. SC-код оперирует только семантически нормализованными множествами – множествами, элементами которых являются sc-элементы. Это дает возможность четко отличать первичные sc-элементы, являющиеся обозначениями внешних объектов, от вторичных sc-элементов, которые являются обозначениями множеств, элементы которых могут непосредственно входить в состав sc-текстов. **Ключевые узлы SC-кода** отражают универсальную базовую семантическую типологию sc-элементов и, соответственно, обозначаемых ими объектов самых различных предметных областей, в том числе и различных метаобластей, определяющих семантику различных видов знаний [3].

Язык визуального (графического) изображения семантических сетей, принадлежащих SC-коду, будем называть **SCg-кодом** (Semantic Code graphical).

Приведем алфавит графических примитивов текстов **SCg-ядра**, который полностью соответствует алфавиту SC-кода.

Табл. 1. Алфавит графических примитивов SCg-ядра

Table 1. Alphabet graphics primitives SCg-core

•	sc.g-узел общего вида является изображением sc-узла общего вида
≡≡≡≡	sc.g-ребро общего вида является изображением sc-ребра общего вида
⇒⇒⇒⇒	sc.g-дуга общего вида является изображением sc-дуги 1-го вида
→→→→	sc.g-дуга базового вида является изображением sc-дуги 2-го вида (т.е. обозначение конкретной, постоянной существующей пары принадлежности)
□	sc.g-рамка является ограничителем внешней информационной конструкции, визуально изображаемой в этой рамке, и является изображением sc-ссылки, обозначающей эту внешнюю информационную конструкцию

Переход от **SCg-ядра** к **SCg-коду** заключается в ослаблении требований, предъявляемых к изображениям семантических сетей, в целях обеспечения большого удобства для человеческого восприятия. Такое ослабление осуществляется в следующих направлениях: вводится приписывание идентификаторов изображаемых sc-элементов, расширяется алфавит графических примитивов, допускается уникальное изображение некоторых sc-узлов, допускается синонимия sc.g-элементов, но при этом синонимичным элементам должны быть приписаны одинаковые идентификаторы, вводятся специальные гра-

фические средства, направленные на повышение наглядности (шинные линии, контура).

Семантика базы знаний интеллектуальной системы – это соотношение между базой знаний и множеством описываемых ею предметных областей. Поэтому для формального рассмотрения семантики баз знаний необходимо иметь формальное уточнение понятия предметной области.

Предметная область (S) — это математическая структура, которая задается следующим набором компонентов $\langle A, H, Q, R, O, B, P \rangle$, где: A — непустое множество, которое будем называть носителем предметной области S , H — шкала множеств, Q — семейство классов, R — семейство отношений, O — семейство алгебраических операций, B — класс исследуемых объектов предметной области S , P — предмет исследования предметной области S .

Предметная область позволяет рассматривать исследуемые объекты на разных уровнях детализации. Детализацию рассмотрения исследуемых объектов можно осуществлять как в рамках исходной (заданной) предметной области, расширяя эту предметную область в соответствующих направлениях, а можно переходить к системе самостоятельных, но связанных между собой предметных областей.

Первым и важнейшим этапом проектирования базы знаний является уточнение структуры описываемой предметной области или нескольких взаимосвязанных предметных областей. Уточнение такой структуры — это, прежде всего, уточнение класса исследуемых объектов, уточнение предмета исследования, уточнение всего семейства сигнатурных элементов. При этом для заданного класса исследуемых объектов и заданного предмета исследования можно построить более качественную и менее качественную предметную область.

Построение семантической модели базы знаний интеллектуальной системы требует не только явного представления спецификаций каждой описываемой предметной области в виде, например, sc-текста, но и явного описания всевозможных связей между этими предметными областями.

Переходя к рассмотрению отношений, заданных на множестве предметных областей, мы фактически переходим к некоторой предметной метаобласти, объектами исследования которой являются всевозможные предметные области (в том числе и сама эта предметная метаобласть).

Обобщая понятия гомоморфизма и изоморфизма алгебраических систем, можно говорить о гомоморфизме и изоморфизме предметных областей, что дает хорошую основу для выявления глубоких нетривиальных аналогий между предметными областями.

Различные предметные области могут пересекаться. То есть элементы одной предметной области могут быть также и элементами другой предметной области. При этом возможны самые различные варианты такого пересечения. Это может быть строгое пересечение, строгое включение. Общие элементы пересекающихся предметных областей могут в рамках этих областей выполнять как одинаковые, так и разные роли. Так, например, первичные элементы одной предметной области могут входить в состав другой предметной области в качестве вторичных элементов, в качестве сигнатурных элементов. Объекты исследования одной предметной области могут

входить в состав другой предметной области в качестве вспомогательных элементов.

Как сделать так, чтобы **объединение (соединение, конкатенация) двух семантических сетей** привело также к новой объединенной (интегрированной) семантической сети? Очевидно, что теоретико-множественное объединение множества элементов одной семантической сети и множества элементов другой семантической сети приведет к знаковой конструкции. Такая знаковая конструкция удовлетворяет всем требованиям, предъявляемым к семантическим сетям, кроме одного – в этой объединенной конструкции могут появиться пары синонимичных элементов, в которых один элемент принадлежит одной объединяемой семантической сети, а другой элемент – другой объединяемой сети. Таким образом, интеграция семантических сетей сводится к поиску и склеиванию пар синонимичных элементов в рамках объединенного множества элементов. Процесс интеграции семантических сетей можно условно разбить на следующие этапы:

- приведение интегрируемых семантических сетей к бинарному виду и к общему алфавиту;
- склеивание sc-элементов, которые имеют одинаковые глобальные идентификаторы (внешние имена);
- склеивание sc-элементов на основании односторонности алгебраических операций
- склеивание sc-элементов на основании логических высказываний о существовании и единственности;
- склеивание кратных связей, принадлежащих отношениям, не имеющим кратных связей и имеющих кратные связи, но не для заданных типов компонентов (например, из знака канторовского множества не могут выходить кратные связи отношения принадлежности).

Логико-семантическая модель компьютерной системы (в том числе интеллектуальной системы) представляет собой абстрактную многоагентную систему, состоящую из **абстрактной семантической памяти**, в которой хранятся семантические сети, и из множества **агентов**, ориентированных на обработку семантических сетей, хранимых в указанной семантической памяти. Очевидно, что такая семантическая память носит **реконфигурируемый (структурно-перестраиваемый) характер**, поскольку процесс обработки семантических сетей, в конечном счете, сводится к генерации и удалению элементов семантических сетей, а также к генерации и удалению связей инцидентности между этими элементами. Другими словами, процесс обработки информации в семантической памяти сводится не к изменению состояния элементов памяти, а к изменению конфигурации связей между ними. Семантическую память можно трактовать как модель семантического пространства, в котором работает компьютерная система, или как абстрактную семантическую модель памяти компьютерной системы. Семейство абстрактных агентов, работающих над семантической памятью можно трактовать как **семантическую модель решения задач** (семантическую модель обработки информации), используемую в соответствующей компьютерной системе, или как операционную семантику этой компьютерной системы. Подчеркнем, что логико-семантическую модель можно построить для любой компьютерной системы (как для интеллектуальной системы, так и для компьютерной системы традиционного вида), обеспе-

чивая тем самым семантическую совместимость не только интеллектуальных систем, но и компьютерных систем любого уровня интеллектуальности. Подчеркнем также, что четкое выделение абстрактного логико-семантического уровня компьютерной системы позволяет не только обеспечивать их семантическую совместимость, но и сформулировать критерии сравнения компьютерных систем по уровню их интеллектуальности. Очевидно, что уровень интеллектуальности компьютерной системы определяется качеством (корректностью, полнотой, многообразием) ее знаний и эффективностью ее умений (т.е. эффективностью используемых ее моделей решения задач).

В основе предлагаемого нами уточнения понятия логико-семантической модели компьютерной системы лежит использование SC-кода. Логико-семантическую модель компьютерной системы, использующую SC-код в качестве языка представления информации в ее памяти будем называть **абстрактной sc-системой**. Память абстрактной sc-системы будем называть **sc-памятью**, а агенты, работающие над sc-памятью, 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. В частности, можно выделить следующие классы sc-агентов:

- sc-агенты, обеспечивающие реализацию правил логического вывода, соответствующих самым различным логическим исчислениям;
- sc-агенты, обеспечивающие реализацию самых различных программ, хранимых в семантической памяти (продукционных и непродукционных);
- sc-агенты анализа качества хранимой базы знаний (корректности, полноты);
- рецепторные sc-агенты;
- эффекторные sc-агенты.

Для формального описания sc-агентов нами предложен **язык программирования SCP** (Semantic Code Processing). Программы, написанные на этом языке являются sc-текстами и описывают обработку sc-текстов, хранимых в структурно перестраиваемой (реконфигурируемой) sc-памяти. В основе такой обработки лежит ассоциативный доступ к хранимым sc-текстам по различным заданным образцам.

Как должна быть устроена **массовая технология модульного проектирования интеллектуальных систем**? Такая технология должна включать в себя:

- Теорию проектируемых систем – уточнение того, как устроены проектируемые системы.
- Структурированную библиотеку типовых многократно используемых компонентов проектируемых систем.
- Инструментальные средства (средства автоматизации) проектирования.
- Методику проектирования.
- Методику обучения проектированию.
- Интеллектуальную help-систему для информационного обслуживания и обучения проектировщиков.
- Инфраструктуру, обеспечивающую организацию проектирования и развитие самой технологии.

III. Заключение

В работе приведены ключевые положения одного из возможных подходов к созданию массовой технологии проектирования компьютерных систем различного уровня интеллекта. Результаты, приведенные в работе, апробируются в рамках открытого проекта OSTIS (<http://ostis.net>).

Работа поддержана грантами БРФФИ-РРФИИ Ф10Р-149, Ф10Р-148, Ф10Р-175, Ф10М-085.

IV. Список литературы

- [1] *Хорошевский В. Ф.* Пространства знаний в сети Интернет и Semantic Web (Часть 1) // Искусственный интеллект и принятие решений. 2008. №1. С.80—97.
- [2] *Sowa J.* Semantic networks. Encyclopedia of Artificial Intelligence, edited by S. C. Shapiro. New York: Wiley, 1992.
- [3] *Голенков В. В., Гулякина Н. А.* Принципы построения массовой семантической технологии компонентного проектирования интеллектуальных систем // Междунар. научн.-техн. конф. «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2011): материалы конф. (Минск, 10—12 февр. 2011 г.). Минск: БГУИР, 2011. С. 21—59.

PUBLIC COMPONENT DESIGN TECHNOLOGY OF SEMANTIC COMPATIBLE COMPUTER SYSTEMS WITH DIFFERENT LEVEL OF INTELLIGENCE

Golenkov V. V., Guliakina N. A., Kolb D. G.
*Belarussian State University Informatics
and Radioelectronics*
6, P. Brovki Str., Minsk, 220013, Belarus
8017-2932324, e-mail: golen@bsuir.by

Abstract — The principles of construction technology of designing intelligent systems which are oriented to semantic representation of knowledge, expansion of the number of developers and shortening time of design are considered.

I. Introduction

The current state in the designing of computer systems reminds Babeldom of various approaches, models, methods, tools and platforms. Artificial intelligence technologies are not targeted to a wide range of developers of intelligent systems and therefore did not have mass distribution.

II. Main Part

We suggest the following approaches, which are aimed to eliminate the above drawbacks:

- orientation to semantic knowledge representation, which completely abstracts from the peculiarities of the technical implementation of intelligent systems.
- Development of the semantic-unified and easily integrable computer systems models in the first stage of their design.
- modular designing based on the libraries of the typical reusable components of intelligent systems.
- gradual evolutionary design based on a basis rapid prototyping.
- Fully compatible designing tools with the developed system - the tools are built as intelligent systems and are based on the same principles.

III. Conclusion

The paper presents the key provisions of the one of the possible approaches to the mass systems development with different levels of intelligence. The results presented in the paper are being tested within the open source project OSTIS (<http://ostis.net>) bounds. This work is supported by Byelorussian and Russian foundation for basic research.