

УДК 004.822:514

О.В. Пивоварчик

**ЯЗЫКОВЫЕ СРЕДСТВА ПОСТРОЕНИЯ БАЗ ЗНАНИЙ
ИНТЕЛЛЕКТУАЛЬНЫХ HELP-СИСТЕМ ДЛЯ РАЗРАБОТЧИКОВ
ПРОГРАММ, ОРИЕНТИРОВАННЫХ НА ОБРАБОТКУ ЗНАНИЙ**

Рассмотрена модель базы знаний интеллектуальной help-системы для разработчиков программ, ориентированных на обработку знаний, предложен семантический язык, предназначенный для описания базы знаний help-системы. Семантический язык является подязыком SC-кода, являющегося базовым языком представления знаний различного вида. Язык включает ключевые понятия, классы ключевых понятий и отношения. Использование языка представлено на примере проектирования базы знаний справочной подсистемы интеллектуальной help-системы по технологии проектирования программ на языке SCP.

База знаний; help-система; язык программирования; язык представления знаний; технология разработки программ; абстрактный синтаксис; денотационная семантика; операционная семантика

O.V. Pivovarchyk

**LANGUAGE TOOLS ARE USED TO CONSTRUCT OF KNOWLEDGE BASES
INTELLECTUAL HELP-SYSTEM FOR DEVELOPERS PROGRAMS,
ORIENTED TO TREATMENT OF KNOWLEDGE**

The article describes a model of the knowledge base of the intellectual help-system for programs' developers oriented to the knowledge treatment. The article proposes the semantic language aimed at describing the help-system knowledge base. The semantic language is a sub-language of the SC-code being the basic language for the presentation of different knowledge. The language includes key concepts, classes of key concepts and relations. The use of the language is presented by example of designing the knowledge base of the referral subsystem of the intellectual help-system according to the technology of programs' designing in the SCP language.

Knowledge base; help-system; programming language; knowledge presentation language; technology of program development; abstract syntax; denotational semantics; operational semantics.

Разработка интеллектуальных информационных систем требует глубоких знаний современных технологий программирования, в том числе ориентированных на обработку баз знаний. Изучение таких технологий – довольно трудоемкий процесс, особенно для программистов, имеющих мало опыта. Поэтому важной задачей является сокращение сроков освоения таких технологий, не приводящее к потере качества разрабатываемых информационных систем. Одним из способов решения задачи является создание программных средств поддержки разработки интеллектуальных систем, которые решают задачи консультантов по проектированию программного обеспечения. Такие системы будем называть интеллектуальными help-системами.

Интеллектуальная help-система представляет собой оформление документации по технологии разработки программ в виде интеллектуальной справочной и обучающей системы, которая обеспечивает всестороннее информационное обслуживание пользователей [1]. Интеллектуальная help-система решает такие актуальные задачи как информационное обслуживание и обучение пользователей, увеличение количества разработчиков интеллектуальных систем, сокращение сроков разработки интеллектуальных систем.

Для решения вышеперечисленных задач разрабатываемая help-система должна содержать информацию о технологии разработки программ, которая включает:

- ◆ теорию (принципы построения) программ;
- ◆ библиотеку типовых многократно используемых компонентов (ip-компонентов), которая входит в состав базы знаний help-системы;
- ◆ инструментальное средство проектирования программ;
- ◆ методику проектирования интеллектуальных систем, которая оформляется как часть базы знаний;
- ◆ методику обучения проектированию интеллектуальной системы, которая также является частью базы знаний [1].

Интеллектуальная help-система включает компоненты, каждый из которых содержит информацию об одной из вышеперечисленных категорий. Следовательно, help-система состоит из следующих компонентов:

- ◆ интеллектуальная help-система по семантической теории программ;
- ◆ интеллектуальная help-система по библиотеке программ;
- ◆ интеллектуальная help-система по комплексу инструментальных средств проектирования программ;
- ◆ интеллектуальная help-система по методике проектирования программ;
- ◆ интеллектуальная help-система по методике обучения проектированию программ.

Для проектирования интеллектуальной help-системы используется открытая семантическая технология компонентного проектирования интеллектуальных систем (Open Semantic Technology for Intelligent Systems – OSTIS) [2]. В основе семантической технологии проектирования интеллектуальных систем лежит семантическая модель представления знаний, которая использует семантическую сеть с базовой теоретико-множественной интерпретацией и представление знаний в s-коде. В соответствии с открытой семантической технологией каждый компонент интеллектуальной help-системы включает: справочную подсистему, подсистему мониторинга и анализа деятельности разработчика программ, подсистему управления обучением. Каждая из подсистем взаимодействует с другими подсистемами, а также может функционировать автономно. Разработка каждой подсистемы состоит из проектирования базы знаний, текстов программ, описывающих поведение соответствующих агентов, и пользовательского интерфейса. Базовым универсальным языком представления знаний технологии является SC-код (Semantic Computer code) [3]. Для описания способов решения задач и поведения агентов в технологии используется базовый язык программирования SCP (Semantic Code Programming) [4, 5, 6].

База знаний каждого компонента содержит формальную модель соответствующей области технологии программирования и иерархическую структуру разделов, каждый из которых описывает некоторый фрагмент базы. Разделы базы знаний именованы, нумеруются и входят в ее состав в качестве оглавления.

В базе знаний справочной подсистемы интеллектуальной help-системы по семантической теории программ представлена информация о теории программ, описаны абстрактный синтаксис и денотационная семантика языка программирования, ориентированного на обработку унифицированных семантических сетей, абстрактная машина, интерпретирующая язык программирования. Кроме этого, в базу знаний включены программы, описывающие конкретные способы решения

соответствующих классов задач. Для каждой программы представлена семантическая спецификация и условия ее применения. Такие программы трактуются как специальный вид знаний. Интеллектуальная help-система по библиотеке программ содержит семантические спецификации программ-компонентов, используемых интеллектуальными системами. Каждой программе ставится в соответствие определенный класс задач с явным описанием, который может быть решен с ее помощью. Интеллектуальная help-система по комплексу инструментальных средств проектирования программ содержит описание редактора текстов программ, трансляции программ, отладки программ, редактора связей между программами, модели используемой памяти. Интеллектуальная help-система по методике проектирования программ содержит описание методики проектирования программ, технологии проектирования, процесса проектирования, утверждения и правила, описывающие способы проектирования программных текстов. Интеллектуальная help-система по методике обучения проектированию программ содержит описание методов обучения и их классификацию, описание принципов построения обучения программированию, алгоритмы выбора методов обучения для конкретного пользователя.

Подсистемы мониторинга и анализа деятельности разработчика программ и управления обучением функционально одинаковы для каждого компонента. Отличие их состоит в том, что они взаимодействуют с различными базами знаний справочных подсистем. Подсистема мониторинга и анализа деятельности разработчика программ осуществляет сбор информации о разработке программ, определяет уровень знаний разработчика программ, анализирует действия и ошибки. Основой базы знаний является формальная модель разработчика программ, которая представляется в виде шаблона. Подсистема управления обучением несет организационную, обучающую и контролирующие функции, организует адаптивный диалог с пользователем. Основной задачей подсистемы является обеспечение пользователю помощи, соответствующей ситуации, и выдача рекомендаций с учетом истории его взаимодействия с help-системой, т.е. осуществление поддержки принятия решений при разработке программного обеспечения для конкретного разработчика программ. При начале работы help-системы используется общая модель пользователя, а затем подсистема управления настраивает эту модель на основании процесса их взаимодействия.

Также в базе знаний каждого компонента хранятся тексты программ, описывающих поведение соответствующих агентов. Такие программы можно рассматривать как часть базы знаний.

Для описания базы знаний используется специальный семантический язык, предназначенный для описания баз знаний help-систем по технологии программирования, в основе которого лежит SC-код. Язык является подязыком SC и включает ключевые узлы для описания формальной модели соответствующей области технологии программирования и разделов, необходимых для проведения последовательного обучения.

Рассмотрим язык проектирования базы знаний на примере проектирования базы знаний справочной подсистемы интеллектуальной help-системы по семантической теории программ на языке SCP. Язык SCP ориентирован на обработку унифицированных семантических сетей. Тексты программ языка являются унифицированными семантическими сетями.

Абстрактный синтаксис представляет собой правила построения корректных программных текстов, не зависящие от конкретного вычислителя. Он предоставляет более абстрактное представление синтаксиса конструкций языка. Определе-

ние синтаксиса предоставляет информацию для описания семантики языка. Для описания синтаксиса выделяется множество ключевых понятий, атрибутивных отношений, определяющих свойства объектов, и отношений, определяющих связи между понятиями. А также выделен ряд утверждений, определяющих правила построения текстов языка и задающие семантику языка. Определение денотационной семантики основывается на выделении некоторых функциональных термов языка и придания им значений [7]. Для языка SCP в качестве термов выделены scr-операторы и scr-программы.

Множество ключевых понятий языка программирования SCP: scr-программа, scr-константа, scr-переменная, scr-параметр, трехэлементная sc-конструкция, пятиэлементная sc-конструкция, одноэлементная sc-конструкция, scr-оператор, тип scr-оператора, scr-оператор генерации sc-конструкций, scr-оператор ассоциативного поиска sc-конструкций, scr-оператор удаления sc-конструкций, scr-оператор проверки условия, scr-оператор изменения свойств, scr-оператор управления scr-процессом, scr-оператор ассоциативной выборки sc-конструкций, genEl-оператор, genElStr3-оператор, genElStr5-оператор, eraseEl-оператор, eraseElStr3-оператор, eraseElStr5-оператор, eraseSetStr3-оператор, searchElStr3-оператор, searchElStr5-оператор, searchSetStr3-оператор, searchSetStr5-оператор, selectNStr3-оператор, selectYStr3-оператор, ifGrEq-оператор, ifGr-оператор, ifCoin-оператор, ifEq-оператор, ifGr-оператор, ifGrEq-оператор, ifVarAssign-оператор, varErase-оператор, varAssign-оператор, contErase-оператор, contAssign-оператор, add-оператор, sub-оператор, mult-оператор, div-оператор, pow-оператор, sin-оператор, asin-оператор, cos-оператор, acos-оператор, call-оператор, waitReturn-оператор, return-оператор.

Множество атрибутивных отношений: const_, var_, prm_, in_, out_, init_, goto_, then_, else_, error_, fixed_, assign_, node_, arc_, elem_, pos_, neg_, fuz_, f_, set1_, set2_, set3_, 1_, 2_, 3_.

На множестве ключевых понятий задаются отношения:

- ◆ отношение синонимичный идентификатор*, связывающее ключевое понятие с множеством его синонимов;
- ◆ отношение пояснение*, связывающее ключевое понятие с его пояснением на естественном языке;
- ◆ отношение определение*, связывающее ключевое понятие с его определением на естественном языке;
- ◆ отношение включение*, отображающее вхождение ключевого понятия в некоторое множество;
- ◆ отношение разбиение*, отображающие подклассы ключевого понятия;
- ◆ отношение значение*, указывающее на вычисленное значение scr-переменной;
- ◆ отношение компонент*, связывающее ключевое понятие с множеством утверждений, содержащих описание правила вхождения компонента в ключевое понятие;
- ◆ отношение семантика*, связывающее ключевое понятие с множеством утверждений, определяющих функции ключевого понятия и результаты функционирования.

При работе с системой пользователь может посмотреть синтаксис в виде семантической сети, в виде, представленном традиционно применяемыми средствами (форма Бэкуса-Наура и др.), в виде естественно-языковых текстов. А также посмотреть функциональное описание языковых конструкций.

Абстрактная машина определяет операционную семантику языка SCP и является формальной моделью интерпретатора языка. Абстрактная машина задается

запоминающей средой (памятью), в которой хранятся перерабатываемые sc-конструкции, и операционной семантикой. Операционная семантика задается коллективом sc-агентов над общей памятью, которые обеспечивают интерпретацию программ, хранящихся в одной памяти с обрабатываемыми знаниями.

Для описания абстрактной машины в базу знаний были включены ключевые понятия необходимые для представления в виде семантической сети:

- ◆ языковых средств, описывающих текущее состояние процесса интерпретации scp-программ: scp-процесс, тип состояния scp-процесса, then-состояние scp-процесса, else-состояние scp-процесса, repeat-состояние scp-процесса, run-состояние scp-процесса, error-состояние scp-процесса, dead-состояние scp-процесса;
- ◆ операционных семантик scp-операторов разных типов: операционная семантика genEl-оператора, операционная семантика genElStr3-оператора, операционная семантика genElStr5-оператора, операционная семантика eraseEl-оператора и т.д.;
- ◆ модели sc-памяти: sc-память, сегмент и др.

На множестве ключевых понятий задаются отношения:

- ◆ отношение пояснение*, связывающее ключевое понятие с его пояснением на естественном языке;
- ◆ отношение компонент*, связывающее ключевое понятие с множеством утверждений, содержащих описание правила вхождения компонента в ключевое понятие;
- ◆ отношение алгоритм*, связывающее ключевое понятие с множеством утверждений, содержащих описание алгоритма работы.

Описание базы знаний справочной подсистемы интеллектуальной help-системы по семантической теории программ содержится в [2].

Используемый язык является sc-языком и обеспечивает описание базы знаний интеллектуальной help-системы для разработчиков программ, ориентированных на обработку знаний. Язык определяется своим расширением множества ключевых узлов sc-кода. Язык позволяет легко интегрировать различные компоненты, построенные с его помощью и с помощью других sc-языков. Используемый язык является расширяемым, поскольку его всегда можно пополнить множеством новых ключевых узлов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Голенков В.В., Гулякина Н.А.* Графодинамические модели параллельной обработки знаний: принципы построения, реализации и проектирования // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». – Минск, 2012. – С. 23-52.
2. Открытая семантическая технология компонентного проектирования интеллектуальных систем [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://ostis.net/mediawiki/index.php/>.
3. *Голенков В.В., Елисеєва О.Е., Иващенко В.П. и др.* Представление и обработка знаний в графодинамических ассоциативных машинах / Под ред. В.В. Голенкова. – Минск: БГУИР, 2001. – 412 с.
4. *Голенков В.В., Осипов Г.С., Гулякина Н.А. и др.* Программирование в ассоциативных машинах. – Минск: БГУИР, 2001. – 276 с.
5. *Голенков В.В., Гулякина Н.А., Королев В.Г., Татаренко В.А., Золотой С.А.* Описание семантики языка SCPas. Операторы преобразования состояния SC-графа (Материалы по математическому обеспечению ЭВМ). – Минск: Ин-т техн. кибернетики АН Беларуси, 1994.
6. *Голенков В.В., Гулякина Н.А., Татаренко В.А., Гапонов П.А., Кузьмицкий В.М.* Описание семантики языка SCPas. Операторы поиска и проверки условий. Операторы управления вычислительным процессом (Материалы по математическому обеспечению ЭВМ). – Минск: Ин-т техн. кибернетики АН Беларуси, 1994.

7. Себеста Р.У. Основные концепции языков программирования. – 5-е изд.: Пер. с англ. – М.: Изд. дом «Вильямс», 2001.
8. Искусственный интеллект: В 3 кн. Кн. 2. Модели и методы: Справочник / Под ред. Д.А. Поспелова. – М.: Радио и связь, 1990.

Статью рекомендовал к опубликованию д.т.н., профессор Б.К. Лебедев.

Пивоварчик Ольга Васильевна – Белорусский государственный университет информатики и радиоэлектроники; e-mail: pivovarchyk@tut.by; Республика Беларусь, 220013, г. Минск, ул. П. Бровки, 6; тел.: +375297277127; кафедра ИИТ; аспирант.

Pivovarchyk Olga Vasilievna – Belarusian State University of Informatics and Radioelectronics; e-mail: pivovarchyk@tut.by; 6, P. Brovki street, Minsk, 220013, Republic of Belarus; phone: +375297277127; the department of ИТ; postgraduate student

УДК 004.89 + 004.4

Л.С. Родзина

ИСПОЛЬЗОВАНИЕ КОНТЕКСТОВ В МОБИЛЬНЫХ ПРИЛОЖЕНИЯХ*

Представлены результаты исследования, посвященного проектированию и применению контекстно-зависимых средств поддержки мобильных приложений. Отмечается, что работа с контекстом позволяет повысить эффективность информационных систем. Главная отличительная особенность САС-систем – способность к адаптации в изменяющихся условиях без вмешательства пользователя. Предлагается двухмерная модель классификации различных источников контекста в ситуации вождения автомобиля. На примере автомобильного приложения исследованы вопросы применения информации, получаемой из разнородных источников в контексте текущей ситуации и разрабатываются сервисы, релевантные решаемой задаче.

Адаптация; САС-системы; контекстная информация; навигация; распознавание речи.

L.S. Rodzina

CONTEXT USED IN MOBILE APPLICATIONS

The paper presents the results of research that is devoted to design and implementation of context-sensitive tools for supporting mobile applications. Furthermore, two-dimensional classification model of the various sources of the context in conditions of driving situation is proposed. Questions regarding of usage information that is gotten from disparate sources are studied on the example of in-car mobile application. Relevance service for the current task are developed.

Adaptation; context-aware computing systems; contextual information; navigation; speech recognition.

Введение. Контекст является основой человеческого общения. При человеко-машинном взаимодействии контекст – это информация, которая может быть использована для характеристики ситуации, имеющей отношение к взаимодействию пользователя с приложением. Современным компьютерным системам и технологиям, действующим по фиксированным программам и не склонным к обучению, явно не хватает контекстных знаний. *Context-Aware Computing (CAC)* представляет собой подход к информационным технологиям, который строится с учетом свойств окружающей среды и позволяет использовать контекстную информацию. *CAC* принято относить к разряду систем *Ubiquitous computing* (или *Pervasive computing*), представляющих особый класс интеллектуальных информационных систем. Основными ис-

* Работа опубликована при поддержке гранта РФФИ 11-07-00094-а.