

УДК 004.5:004.822

## УНИФИЦИРОВАННАЯ МОДЕЛЬ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Д.Г. КОЛЬ

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровка, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 17 марта 2009 года*

Описана модель пользовательского интерфейса интеллектуальных систем, построенных на основе семантических сетей. Приведена концепция построения пользовательских интерфейсов интеллектуальных систем на базе приведенной модели.

*Ключевые слова:* пользовательский интерфейс, управляющий элемент интерфейса, семантический код, интерфейсная задача, интерфейсная команда.

### Введение

Бурные темпы развития технологий искусственного интеллекта в последние годы привели к увеличению числа пользователей интеллектуальных систем. Большинство из них являются профессионально неподготовленными к работе с пользовательским интерфейсом интеллектуальных систем.

Данный факт ставит перед разработчиками пользовательских интерфейсов интеллектуальных систем ряд проблем:

- снижение требований к подготовке пользователя для работы с интеллектуальной системой;
- сокращение времени адаптации конечного пользователя к интерфейсу нового программного продукта;
- увеличение числа внешних языков взаимодействия пользователя и интеллектуальной системы для повышения когнитивного эффекта восприятия предметной области.

Реализация пользовательского интерфейса может быть выполнена на основе известных моделей: задачно-ориентированной модели, объектно-ориентированной модели или появившихся в последнее время их модификации – моделей индуктивного пользовательского интерфейса [1] и автоматизации [2]. Однако при этом разработчику необходимо прилагать специальные усилия для: минимизации времени разработки пользовательского интерфейса без потери качества; повышения качества пользовательского интерфейса без одновременного его усложнения и увеличения времени на разработку; преодоления несогласованности интерфейсных компонентов, которая приводит к потерям времени на их интеграцию в интерфейс интеллектуальной системы.

В настоящее время существует ряд работ, которые направлены на решение рассмотренного круга проблем. Одним из подходов является унификация методов проектирования пользовательских интерфейсов с помощью онтологий на основе модели ориентированного подхода, предложенного В. В. Грибовой [3]. Другим подходом является интеллектуализация различных функций пользовательского интерфейса с помощью экспертных систем, как это было предложено в работе [4]. Однако использование приведенных подходов хотя и снимает ряд проблем, связанных с проектированием пользовательских интерфейсов и дальнейшим их функционированием, но не решает такие задачи, как интеграция пользовательских интерфейсов схожих систем и построение пользовательских интерфейсов на основе имеющихся компонентов.

## Принципы, лежащие в основе предлагаемого подхода

Пользовательский интерфейс интеллектуальной системы (ИС) трактуется как специализированная ИС, являющаяся посредником диалога между пользователем и некоторой основной ИС (ОИС). Принципиальным является то, что и ОИС, и пользовательский интерфейс построены на основе SC-кода [5], то есть обе являются sc-системами. Это дает возможность ОИС и ее пользовательскому интерфейсу иметь общие фрагменты баз знаний (БЗ). Наличие общих фрагментов БЗ позволяет пользователю задавать ОИС любые вопросы, касающиеся пользовательского интерфейса.

Введем ряд положений, характеризующих предлагаемую модель пользовательского интерфейса:

- пользовательский интерфейс будем трактовать как sc-систему, решающую задачи взаимодействия пользователя с ОИС;
- в основе модели лежит использование SC-кода;
- управляющими элементами интерфейса являются узлы семантической сети;
- управляющие элементы интерфейса отображаются с помощью SCg-кода (Semantic Code graphical) [5] – унифицированного языка визуализации семантических сетей;
- выделяются управляющие элементы двух классов: элементы SC-кода, обозначающие окна; элементы SC-кода, обозначающие классы интерфейсных команд.

Для разработки унифицированной модели пользовательского интерфейса ИС необходимо решить следующие задачи:

- разработать содержательную структуру БЗ пользовательских интерфейсов ИС;
- разработать языки, ориентированные на представление различных видов знаний в БЗ пользовательских интерфейсов;
- унифицировать внешний облик пользовательских интерфейсов;
- унифицировать принципы взаимодействия пользователя и ИС на уровне пользовательских действий;
- разработать и реализовать операции пользовательских интерфейсов ИС;
- разработать принципы интеграции пользовательских интерфейсов ИС.

В соответствии с классами решаемых интерфейсных задач можно выделить следующие классы интерфейсных подсистем:

- просмотрщики информационных конструкций внешних языков, среди которых можно выделить просмотрщики информационных конструкций с временной составляющей или проигрыватели (например, видеопроигрыватели или аудиопроигрыватели) и просмотрщики информационных конструкций без временной составляющей (например, просмотрщики традиционных текстов);
- редакторы внешних информационных конструкций для различных способов отображения информации;
- трансляторы информационных конструкций из внешнего представления в SC-код;
- трансляторы информационных конструкций из SC-кода во внешнее представление.

Каждая выделенная интерфейсная подсистема трактуется как специализированная sc-система, имеющая свою БЗ и машину обработки знаний (МОЗ). Пользовательский интерфейс в целом является результатом интеграции всех его подсистем.

Для МОЗ пользовательского интерфейса характерны следующие классы операций:

- рецепторные операции, инициируемые пользователем и приводящие к изменению в sc-памяти. К таким операциям относятся операция синтеза sc.g-узла заданного типа (при инициировании ее пользователем), операция синтеза sc.g-дуги заданного типа (при инициировании ее пользователем), операция трансляции sc.g-конструкции с указанного sc-окна в семантически эквивалентную ей sc-конструкцию и другие операции;
- эффекторные операции, отображающие sc-конструкции из sc-памяти пользователю. К таким операциям относятся операция синтеза sc.g-узла заданного типа (при инициировании ее системой), операция синтеза sc.g-дуги заданного типа (при инициировании ее системой), операция трансляции sc-конструкции в семантически эквивалентную ей sc.g-конструкцию в указанном sc-окне и другие операции;

- операции «память-память», обрабатывающие информационные конструкции без участия пользователя. К таким операциям относятся операция размещения информационной конструкции согласно указанному алгоритму размещения, операция подсветки последней пришедшей в указанное sc-окно информационной конструкции, операции интерпретации правил трансляции с SC-кода на внешний язык.

### **Унифицированный способ визуализации семантических сетей и интерфейсной информации**

Визуальные модели обладают особой познавательной силой, фактически представляя средства когнитивной графики для структурирования информации. На практике используется более сотни методов визуального структурирования [6]. В большинстве своем в основу этих методов положены разнообразные сетевые модели представления знаний.

Пользовательский интерфейс каждой прикладной sc-системы визуализируется с помощью SCg-кода – унифицированного способа визуализации семантических сетей закодированных с помощью SC-кода. Поэтому в состав каждой прикладной sc-системы включается редактор SCg-кода, транслятор конструкций SCg-кода в SC-код и обратно, просмотрщик конструкций SCg-кода.

Минимальные, но семантически полные средства SCg-кода, обеспечивающие изображение любых конструкций sc-кода, назовем ядром SCg-кода или сокращенно SCg-ядром. Алфавит SCg-ядра приведен в таблице .

**Алфавит SCg-ядра**

| Алфавит SCg-ядра  | Пояснение  |
|---|--|
|  | sc.g-узел     /* является графическим изображением sc-узла */  |
|  | sc.g-ребро     /* является графическим изображением sc-ребра общего вида */  |
|  | обобщенная sc.g-дуга     /* является графическим изображением обобщенной sc-дуги */  |
|  | основная sc.g-дуга     /* является графическим изображением sc-дуги основного вида (т. е. обозначения константной постоянно позитивной пары принадлежности) */   |
|  | sc.g-рамка 1-го вида     /* является ограничителем внешних информационных конструкций, визуально изображаемых в этой рамке, и является обозначением класса внешних информационных конструкций аналогичного вида*/    |
|  | sc.g-рамка 2-го вида     /* является ограничителем внешней информационной конструкции, визуально изображаемой в этой рамке, и является изображением sc-узла, обозначающего эту внешнюю информационную конструкцию */ |

Конструкции SCg-ядра удобны для иллюстрации синтаксических и семантических свойств SC-кода. Однако для широкого практического использования конструкции SCg-ядра не пригодны, так как слишком громоздки и ненаглядны. Для решения этой проблемы предлагается использовать расширение SCg-ядра – SCg-код. SCg-код позволяет визуализировать наиболее часто используемые элементы согласно семантической классификации sc-элементов [5].

Вся информация, доступная пользователям sc-систем, отображается в окнах sc-системы или sc-окнах. Такие окна бывают разных видов, могут иметь разные типы и характеристики. Каждое sc-окно трактуется как sc-узел, отображенный на экране и имеющий содержимое. Отображение sc-узлов на экране осуществляется при помощи SCg-кода.

В зависимости от типа содержимого sc-узла различают текстовые, графические, речевые и другие sc-окна. Окно пользовательского интерфейса sc-системы, содержимым которого является sc.g-конструкция, называется sc.g-окном пользовательского интерфейса sc-системы. Соответственно sc-окно, содержимым которого является текст (строка или цепочка символов)

какого-либо линейного языка (например русского или английского), называется текстовым sc-окном пользовательского интерфейса sc-системы. По аналогии вводятся другие типы sc-окон.

Окна пользовательского интерфейса sc-системы в зависимости от типа отображаемой на них информации делятся на простые и гибридные. К простым sc-окнам относятся окна, содержание которых носит однородный характер, например, текст, рисунок, речь. Гибридные sc-окна характеризуются неоднородностью представляемой в них информации. В частности, гибридным может являться любое sc.g-окно, если оно содержит другие sc-окна.

### **Унификация пользовательских команд и элементарных пользовательских действий**

В процессе работы с пользовательским интерфейсом sc-системы пользователь формулирует пользовательскому интерфейсу sc-системы различные интерфейсные задачи. Формулировка таких задач осуществляется пользователем средствами специализированного языка спецификации задач. В алфавит данного языка включены всевозможные интерфейсные команды, а текстами языка являются протоколы решения интерфейсных задач.

Перейдем к рассмотрению интерфейсных команд. Интерфейсная команда – это sc-конструкция, описывающая (специфицирующая) и инициирующая определенную операцию sc-системы, выполняемую над некоторыми объектами операции. Пользовательская интерфейсная команда – это информационная конструкция, создаваемая пользователем, являющаяся пользовательским сообщением, адресуемым пользовательскому интерфейсу. Для формулировки пользовательских интерфейсных команд sc-системе используется унифицированный язык пользовательских действий, алфавит которого составляют типовые интерфейсные задачи или элементарные пользовательские действия.

Любое пользовательское сообщение, адресуемое ОИС, кроме команд редактирования информационных конструкций, можно сформулировать с помощью SCg-кода. Для того, чтобы пользователь мог выполнить действие (операцию) в sc-системе, он должен сформулировать команду инициирования этой операции с помощью SCg-кода.

Выделяют следующие классы интерфейсных команд:

- атомарные интерфейсные команды;
- неатомарные интерфейсные команды.

Элементами атомарного класса интерфейсных команд являются знаки однотипных интерфейсных команд, формируемых в разное время. Каждому классу интерфейсных команд сопоставлен некоторый ключевой элемент, являющийся sc-узлом, и обозначающий этот класс.

Элементами неатомарного класса интерфейсных команд являются классы однотипных классов атомарных команд и однотипных классов неатомарных команд. Каждая команда имеет параметры, множеству которых принадлежат:

- знак атомарного класса команд, соответствующий инициируемой команде;
- знаки аргументов, которые принадлежат команде и над которыми выполняется действие заданное командой.

Таким образом, пользовательская интерфейсная команда представляется знаком команды и ее параметрами. Аргументы – это элементы команды, относительно которых выполняется команда.

В начале работы пользователя с sc-системой формируется sc-окно, которое называется главным (резидентным) sc-окном. Главное sc-окно принадлежит к классу sc.g-окон. Оно содержит всю вспомогательную информацию, необходимую пользователю для работы с sc-системой. Элементы совокупности ключевых элементов пользовательского интерфейса sc-систем, непосредственно отображенных в главном sc-окне, назовем резидентными ключевыми элементами пользовательского интерфейса sc-систем.

### **Унифицированная модель просмотрщика**

Рассмотрим класс подсистем пользовательского интерфейса просмотрщика. БЗ просмотрщика включает следующие компоненты:

1) формальное описание алгоритмов размещения информационных конструкций внешних языков, использующихся в sc-системе;

2) формальное описание представления информационных конструкций внешних языков, использующихся в системе, которое декомпозируется на формальное описание правил вида представления и порядка представления информационных конструкций внешних языков, использующихся в sc-системе, и формальное описание правил указания локуса внимания на информационные конструкции внешних языков, использующихся в системе;

3) формальное описание стилей представления информационных конструкций на внешних языках, использующихся в sc-системе, которое декомпозируется на:

- формальное описание стилей цветового кодирования информационных конструкций внешних языков;

- формальное описание стилей кодирования шрифтом информационных конструкций внешних языков;

- формальное описание стилей кодирования размером информационных конструкций внешних языков;

- формальное описание стилей кодирования звуком информационных конструкций внешних языков;

- формальное описание стилей кодирования анимацией информационных конструкций внешних языков.

В онтологии просмотрщика представлены такие понятия как тип кодирования, стиль кодирования, последнее сообщение и другие понятия, характеризующие доступные для просмотра информационные конструкции.

К операциям МОЗ просмотрщика информационных конструкций относятся интерфейсные операции, которые позволяют отобразить информационные конструкции на внешних языках, доступных sc-системе. Отображение осуществляется в наглядном для пользователя виде на основе информации о семантической окрестности данных конструкций, присутствующей в базе знаний sc-системы. К таким операциям относятся операция размещения информационной конструкции согласно указанному алгоритму размещения, операция подсветки последней пришедшей на указанное sc-окно информационной конструкции и другие подобные операции.

### **Унифицированная модель редактора**

БЗ редактора включает следующие компоненты:

- формальное описание возможных интерфейсных задач редактирования, которые можно разделить на следующие классы: интерфейсные задачи синтеза визуальных объектов и интерфейсные задачи модификации визуальных объектов. К задачам первого класса можно отнести задачу синтеза визуального объекта заданного типа (генерацию sc.g-узла заданного типа, задачу построения sc.g-конструкции с заданными характеристиками), задачу синтеза объекта заданного типа в заданной точке экрана и другие подобные задачи. К задачам второго класса можно отнести задачи модификации идентификатора визуального объекта (задача изменения идентификатора sc.g-узла), задачи изменения цвета визуального объекта (задача изменения цвета sc.g-дуги) и другие подобные задачи;

- формальное описание интерфейсных задач удаления визуальных объектов;

- формальное описание интерфейсных задач перемещения визуальных объектов;

- формальное описание пользовательских интерфейсных команд на языке элементарных пользовательских действий;

- формальное описание внешних языков представления информации, с конструкциями которых позволяет работать редактор, включая спецификацию SCg-кода;

- формальное описание унифицированного языка простейших интерфейсных задач или элементарных пользовательских действий.

В онтологии редактора представлены такие понятия как элемент алфавита, алфавит, отображенный объект, элемент изображения, класс элемента, команда и другие понятия, характеризующие процесс редактирования.

МОЗ редактора для внешних языков представления информационных конструкций различного вида содержит интерфейсные операции, которые позволяют создавать и изменять ин-

формационные конструкции, представленные в окнах sc-системы, используя внешние языки доступные sc-системе. К таким операциям относятся классы операций синтеза конструкций внешних языков заданного типа (операция генерации sc.g-дуги заданного типа, операция генерации sc-окна заданного типа), операция удаления информационной конструкции внешнего языка, классы операций изменения идентификаторов информационных конструкций внешнего языка и другие подобные операции.

### **Унифицированная модель транслятора информационных конструкций внешнего представления в конструкции SC-кода**

БЗ транслятора включает следующие компоненты:

- формальное описание классификации типовых внешних информационных конструкций (в том числе и sc.g-конструкций) и семантически эквивалентных им sc-конструкций;
- формальное описание правил трансляции типовых внешних информационных конструкций в семантически эквивалентные им sc-конструкции;

В онтологии транслятора представлены такие понятия, как трансляция, семантическая эквивалентность текстов и другие понятия, характеризующие процесс трансляции.

МОЗ транслятора содержит интерфейсные операции, которые позволяют транслировать информационные конструкции, представленные на внешних языках доступных sc-системе в sc-конструкции. К операциям такого класса можно отнести операцию трансляции произвольной sc.g-конструкции в семантически эквивалентную ей sc-конструкцию и другие подобные операции.

### **Унифицированная модель транслятора конструкций SC-кода в информационные конструкции внешнего представления**

БЗ транслятора включает следующие компоненты:

- формальное описание классификации типовых sc-конструкций и семантически эквивалентных им sc.g-конструкций;
- формальное описание правил трансляции типовых sc-конструкций в семантически эквивалентные им типовые внешние информационные конструкции.

МОЗ транслятора содержит интерфейсные операции, которые позволяют транслировать конструкции sc-кода в информационные конструкции внешних языков доступных в sc-системе. К операциям такого класса можно отнести операцию трансляции произвольной sc-конструкции в семантически эквивалентную ей sc.g-конструкцию и другие подобные операции.

### **Интеграция пользовательских интерфейсов sc-систем.**

Приведенный подход к организации пользовательских интерфейсов sc-систем дает широкие интеграционные возможности. При интеграции пользовательских интерфейсов могут встретиться два случая:

- интеграция подсистем пользовательских интерфейсов, разработанных как sc-система, в пользовательский интерфейс sc-системы;
- интеграция подсистем пользовательских интерфейсов, построенных без использования SC-кода, в пользовательский интерфейс sc-системы.

Для первого случая интеграции необходимо выполнить следующие шаги.

*Шаг 1.* Декомпозировать интегрируемую подсистему на подсистемы таким образом, чтобы множество интерфейсных задач не пересекалось с множеством интерфейсных задач, решаемых пользовательским интерфейсом основной sc-системы.

*Шаг 2.* Исключить из интегрируемой подсистемы интерфейсные задачи, которые решаются в основной sc-системе.

*Шаг 3.* Для каждого внешнего языка в системе создать ключевой элемент пользовательского интерфейса, который обозначает неатомарный класс пользовательских интерфейсных команд, позволяющих общаться с sc-системой на этом внешнем языке.

*Шаг 4.* Провести интеграцию базы знаний и машины обработки знаний интегрируемой подсистемы пользовательского интерфейса с базой знаний и машиной обработки знаний пользовательского интерфейса основной sc-системы.

Для второго случая интеграции необходимо выполнить следующие шаги.

*Шаг 1.* Разработать транслятор для работы с внешним языком интегрируемой подсистемы.

*Шаг 2.* Специфицировать классы sc-окон для интегрируемой подсистемы.

*Шаг 3.* Специфицировать классы команд, с помощью которых будет происходить инициирование работы с интегрированной подсистемой.

*Шаг 4.* Включить знаки этих классов команд в перечень ключевых элементов.

После проведения процесса интеграции при работе sc-окон, интегрированных в перечень отображаемых резидентных элементов пользовательского интерфейса, будут добавляться элементы, являющиеся знаками классов команд интегрированных подсистем.

### **Заключение**

Использование предложенного подхода к проектированию пользовательского интерфейса интеллектуальных систем на практике дает следующие преимущества:

- унификацию способа организации диалога пользователя с интеллектуальной системой за счет разработки языковых средств ведения диалога и, как следствие этого, снижение начальных требований, предъявляемых к подготовке конечных пользователей;
- возможность интеграции базы знаний и машины обработки знаний пользовательского интерфейса интеллектуальных систем с другими базами знаний и машинами обработки знаний, построенными с использованием SC-кода.

Работа выполнена при поддержке БРФФИ-РФФИ (грант № Ф08Р-137).

## **UNIFIED MODEL OF USER INTERFACE OF INTELLIGENT SYSTEMS**

D.G. KOLB

### **Abstract**

Briefly described a model of intelligent systems with user interface built on the basis of semantic networks. A concept of user interfaces intelligent systems based on the present model is given.

### **Литература**

1. Microsoft Corporation. Индуктивный пользовательский интерфейс. [Электронный ресурс]. Режим доступа: <http://www.rsdn.ru/article/ui/UIGuidelines.xml>. Дата доступа 10.03.2009
2. Microsoft Corporation. Модель автоматизации пользовательского интерфейса. [Электронный ресурс]. Режим доступа: [http://msdn.microsoft.com/ru-ru/library/ms747327.aspx#UI\\_Automation\\_Model](http://msdn.microsoft.com/ru-ru/library/ms747327.aspx#UI_Automation_Model). Дата доступа 10.03.2009.
3. Грибова В.В., Клецев А.С. Концепция разработки пользовательского интерфейса на основе онтологий. Часть 1. Инструментарий для разработки пользовательского интерфейса (обзор литературы). Основная идея подхода. Владивосток, 2003.
4. Джарратано Д., Райли Г. Экспертные системы: принципы разработки и программирования. М., 2007.
5. Голенков В.В., Елисеева О.Е., Иващенко В.П. и др. Представление и обработка знаний в графодинамических ассоциативных машинах / Под ред. В.В. Голенкова. Минск, 2001.
6. Гаврилова Т. А., Гулякина Н.А. // Искусственный интеллект и принятие решений. 2008. № 1. С. 15-21