

МОДЕЛЬНО-ОРИЕНТИРОВАННЫЙ СПОСОБ ПРОЕКТИРОВАНИЯ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

А. Э. Зянчурин

*Санкт-Петербургский институт информатики и автоматизации РАН,
199178, Санкт-Петербург, Россия
E-mail: zwilas@gmail.com*

Предлагается модельно-ориентированный подход к проектированию пользовательских интерфейсов экспертных систем. Приводится ряд положений, где обосновывается необходимость применения декларативно-процедурных языков к описанию пользовательских интерфейсов с использованием модельно-ориентированного подхода. Представлены концептуальная модель пользовательского интерфейса экспертной системы, описывающая его основные свойства и компоненты, и концепция динамической реконфигурации пользовательского интерфейса в соответствии с состоянием вычислительного процесса. Рассмотренный подход является начальным этапом в формировании новой методологии проектирования пользовательских интерфейсов с использованием языков с визуальной нотацией.

Ключевые слова: *экспертные системы, адаптивный пользовательский интерфейс, языки представления знаний, когнитивный образ, модельно-ориентированный подход*

Введение. Один из важных моментов разработки экспертных систем (ЭС) — проектирование пользовательского интерфейса, поскольку именно интерфейс является средством взаимодействия конечного пользователя с системой. Таким образом, состав, структура, информационное содержание и функциональное наполнение пользовательского интерфейса определяют степень эффективности этого взаимодействия. Современные ЭС обрабатывают большие объемы разнородной, быстроменяющейся информации, что сложно для восприятия конечным пользователем вследствие отсутствия целостного (когнитивного) образа. Этим обуславливается актуальность научно-технической задачи по разработке новых подходов к проектированию пользовательских интерфейсов.

Для формирования целостного образа реального объекта интерфейс должен обладать свойством адаптивности по отношению к задачам конечного пользователя. Иными словами, состояние компонентов (графических элементов) интерфейса должно в режиме реального времени изменяться в соответствии с изменяющейся ситуацией. В современных информационных технологиях и языках, таких как, например, ASP.NET или Java, проектирование части программы решения задач (back-end) зачастую осуществляется совместно с проектированием пользовательского интерфейса (front-end) и выполняется одним и тем же разработчиком, который не обладает знаниями о формировании когнитивного образа реального объекта в предметной области. Это снижает эффективность функционирования экспертной системы и ее взаимодействия с конечным пользователем (лицом, принимающим решение). Использование методов теории искусственного интеллекта позволяет избежать указанных проблем и существенно упростить процесс проектирования пользовательского интерфейса за счет предоставления модельно-алгоритмических средств его разработки непосредственно специалистом в конкретной предметной области. Предлагаемая в работе [1] методология определяет ход проектирования автоматизированных систем как создание непосредственно экспертом

в предметной области прикладного программного обеспечения в доступных и понятных ему терминах, что важно и при разработке пользовательского интерфейса ЭС.

Модельно-ориентированный подход как основа проектирования пользовательских интерфейсов. Факты и правила предметной области формализуют в виде так называемых моделей представления знаний (МПЗ) и соответствующих языков представления знаний (ЯПЗ) [1, 2]. По мнению А. С. Нариньяни [3], модельно-ориентированный подход к проектированию в программной инженерии имеет ряд преимуществ по сравнению с традиционным (алгоритмическим) способом разработки программного обеспечения: модель носит описательный (декларативный) характер, тогда как алгоритм — процедурный и направлен на отражение отношений между элементами модели. Таким образом, Нариньяни предложил модельно-ориентированную методологию проектирования программного обеспечения, где алгоритм оказывается ниже модели, с точки зрения гносеологической цепочки представления знаний, и становится инвариантным блоком в составе модели [3]. Этот подход в явном виде применен в методологии проектирования Н-моделей Нариньяни и в концептуальном программировании Тыгу [4].

Отметим, что поскольку пользовательский интерфейс является программой и его модель также связана с фактами и правилами предметной области и соответственно знаниями эксперта, то изложенные выше положения актуальны и для проектирования интерфейса. Из этого следует, что проектирование интерфейса должно базироваться на разработке экспертом его модели, на основе которой впоследствии производится прикладное программное обеспечение. Следовательно, система должна предложить для эксперта в предметной области такую МПЗ и такой ЯПЗ, которые по своим выразительным свойствам близки к естественным языкам и позволят спроектировать прикладное программное обеспечение пользовательского интерфейса без привлечения профессиональных программистов.

Концептуальная модель пользовательского интерфейса на языке Express. Для определения модели пользовательского интерфейса рассмотрим понятие „модель“ в общем случае. Как отмечено в работе [5], „... модель является не просто образом — заменителем оригинала, не вообще каким-то отображением, а отображением целевым“. В случае модели пользовательского интерфейса проектирование осуществляется с целью определения состояния совокупности его графических элементов в квант времени с соблюдением свойства инвариантности состояния реального объекта.

Анализ современных ЯПЗ, согласно ГОСТ Р ИСО 10303-11-2009 и ГОСТ Р 50754-95, а также работам [6—9], показал, что визуальный язык Express обладает необходимыми выразительными свойствами и имеет визуальную нотацию [10, 11]. Таким образом, использование этого языка позволяет проектировать модели интерфейсов в терминах, понятных эксперту.

В языке Express модель состоит из одной или нескольких частей, называемых схемами. Схема — раздел описания области определения данных, необходимые типы которых вводятся в схеме. Основной элемент модели — сущность. Для того чтобы различать сущности между собой, необходимо присвоить определению сущности некое уникальное обозначение (идентификатор).

Описание схемы начинается с заголовка, состоящего из служебного слова „schema“ и идентификатора — имени схемы. Далее следует содержательная часть — тело схемы. Описание заканчивается служебным словом `end_schema`:

```
schema <имя схемы>; <тело схемы>; end_schema;
```

Сущность — тип данных, представляющих набор концептуальных или реальных физических объектов с некоторыми общими свойствами; служит для описания объектов предметной

области. Свойства сущности выражаются в виде атрибутов. Описание сущности начинается со служебного слова „entity“, за которым следует идентификатор сущности:

```
entity <имя сущности>; <идентификатор атрибута>:<тип атрибута>; . . .
end_entity;
```

В языке Express существует визуальная нотация — Express-G, благодаря чему имеется возможность в наглядной форме показать сущности концептуальной модели и отношения между ними.

Приведем описание модели пользовательского интерфейса средствами языка Express. Некоторые из графических элементов интерфейса предназначены для обработки пользовательских действий, другие создаются в целях информирования. Кроме того, существуют элементы, назначение которых — хранить другие элементы, формируя тем самым иерархическую структуру интерфейса. Таким образом, были выделены две основные составляющие в его модели: графические элементы и события [12].

На рис. 1 представлено концептуальное описание сущности elements_interface (графических элементов пользовательского интерфейса). Как видно из приведенной схемы, сущность elements_interface может быть представлена двумя обязательными сущностями ID (идентификатор) и TYPE (тип), каждая из которых представляется более простыми сущностями.

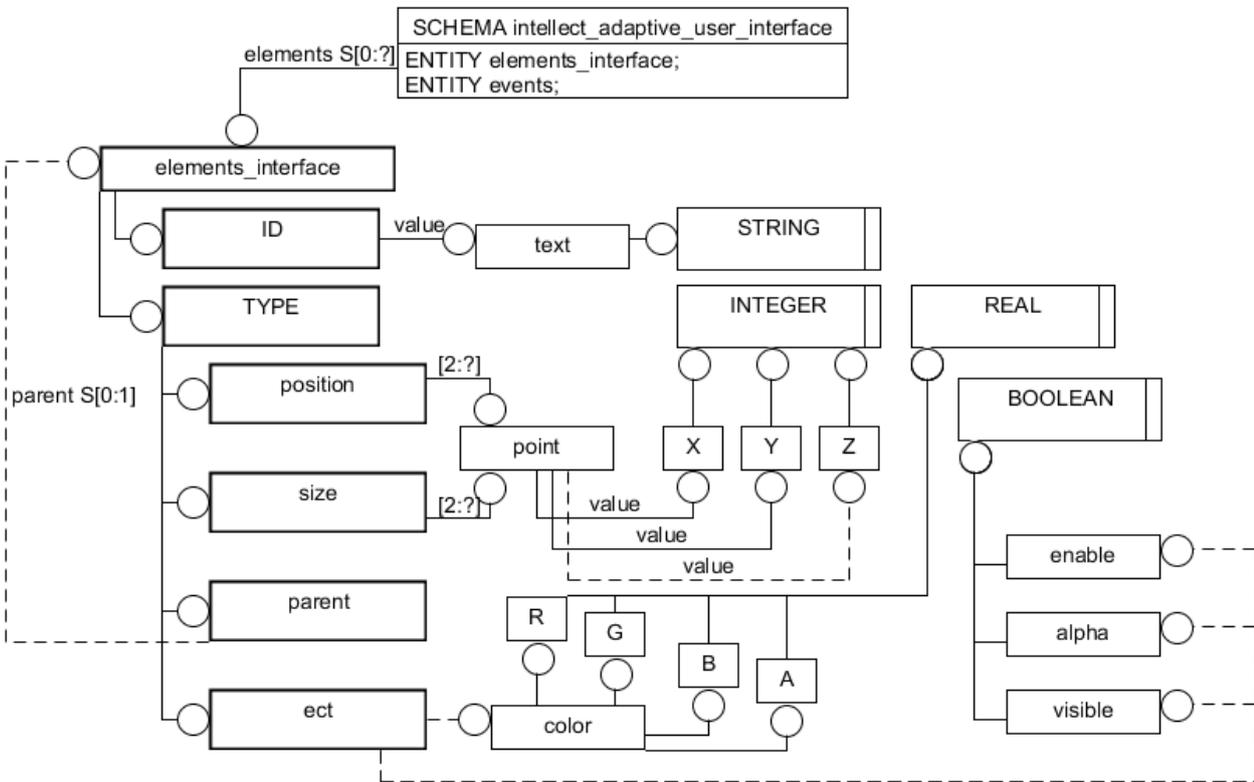


Рис. 1

На рис. 2 представлено концептуальное описание сущности events (возможные действия в пользовательском интерфейсе). Сущность events связана с сущностью elements_interface отношением, которое показывает, что один графический элемент может отражать сколь угодно много событий. Сущность event_interface представляется сущностями ID, STATE, ACT и необязательной сущностью CONTROLLER. Сущность STATE описывает множество состояний элемента интерфейса, в которых он может находиться в течение всего времени взаимодействия с пользователем. Сущность ACT описывает весь набор обрабатываемых пользовательских действий. В свою очередь, CONTROLLER описывает действия, которые растянуты во

времени: например, плавное перемещение элемента интерфейса, плавное изменение размера элемента интерфейса или плавное пропадание либо возникновение элемента интерфейса.

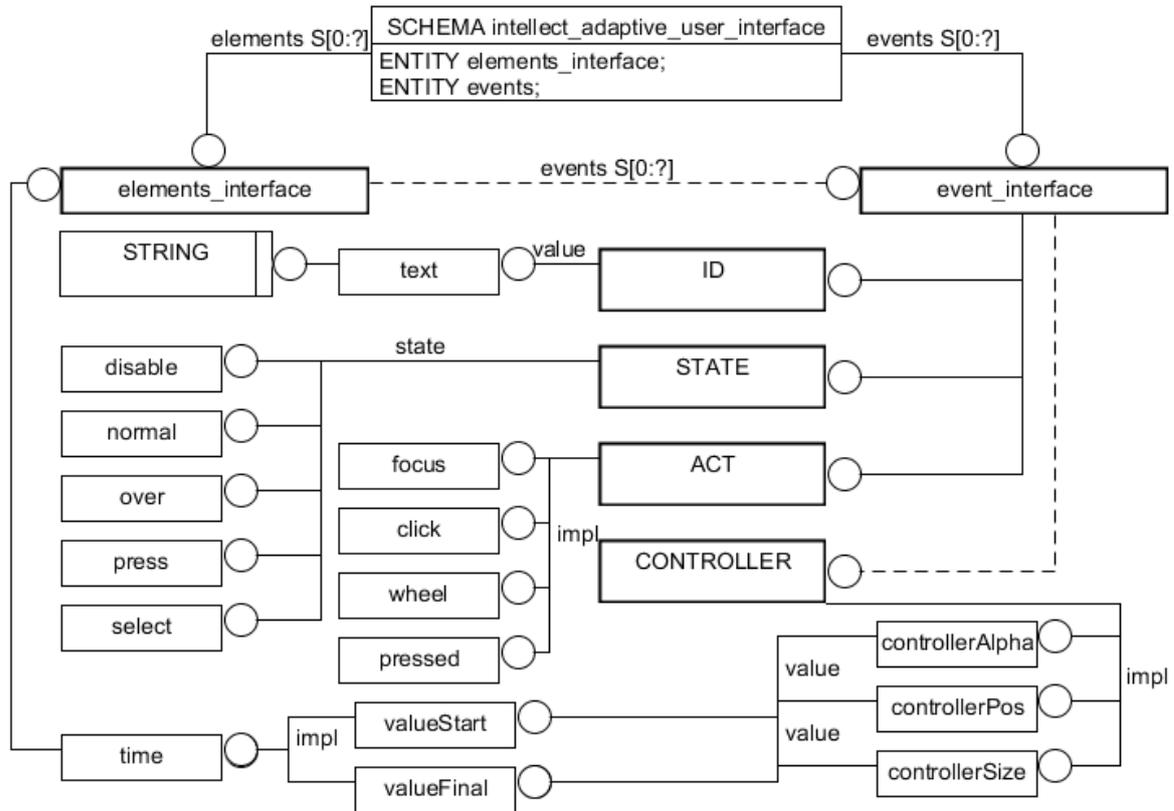


Рис. 2

Данная модель отражает статическую структуру интерфейса, что является недостаточным для формирования когнитивного образа. Необходимо обеспечить комплекс методов динамической реконфигурации графических элементов интерфейса, что позволит его адаптировать к задачам конечного пользователя.

Сценарная база знаний пользовательского интерфейса. Рассмотрим типичную схему взаимодействия пользователя с интерфейсом ЭС — они попеременно обмениваются сообщениями: таким образом, формируется своего рода диалог, который имеет множество вариантов развития, но ограничен задачами пользователя.

В теории искусственного интеллекта так называемые „стереотипные“ знания, как правило, описываются сценариями. Под сценарием понимается формализованное описание стандартной последовательности взаимосвязанных фактов, определяющих типичную ситуацию в предметной области. Это могут быть последовательности действий или процедур, описывающих способы достижения целей взаимодействующих объектов [13]. Исходя из этого предлагается формировать сценарную базу знаний (СБЗ), в которую эксперт будет закладывать сценарии реконфигурации графических элементов интерфейса в соответствии с задачами пользователя [13].

СБЗ описывает множество допустимых последовательностей действий для достижения той или иной выбранной цели. В ходе исследования в качестве языка СБЗ [14] был выбран базовый сценарный язык представления знаний — Scenario Knowledge Base Language (SKBL). Основным понятием в языке SKBL является элементарная модель изменения состояния графических элементов, которая содержит:

— начальные или входные данные — состояние, в котором находится пользовательский интерфейс в определенный момент времени;

— предварительное условие — задает ограничения на допустимость или возможность выполнения действия;

— действие — переход интерфейса из одного состояния в другое (действие исполнительной системы или пользователя);

— результат — состояние интерфейса после выполнения действия.

СБЗ состоит из структурированного множества сценарных моделей, за которыми закрепляются определенные сегменты информации обрабатываемой ЭС. Выходные данные каждого сценария — определенное состояние графических элементов интерфейса, которые формируют образ реального объекта, заложенный экспертом.

Заключение. Рассмотрены некоторые аспекты задачи проектирования концептуальных моделей пользовательского интерфейса экспертной системы. Перспективным направлением исследований является разработка интерфейса на основе формирования сценарной базы знаний, описывающей изменения состояния графических элементов интерфейсов, а также использования сценариев для адаптации состояния интерфейса к разнородной информации, формируемой в ЭС.

СПИСОК ЛИТЕРАТУРЫ

1. *Охтилев М. Ю.* Основы теории автоматизированного анализа измерительной информации в реальном времени. Синтез системы анализа. СПб: ВИКУ им. А. Ф. Можайского, 1999. 161 с.
2. Искусственный интеллект: Справочник / Под ред. *Д. А. Поспелова*. Кн. 2. Модели и методы. М.: Радио и связь, 1990. 304 с.
3. *Нариньяни А. С.* Модель или алгоритм: новая парадигма информационной технологии // Информационные технологии. 1997. № 4. С. 17—22.
4. *Тьугу Э. Х.* Концептуальное программирование. М.: Наука. 1984. 256 с.
5. *Перегудов Ф. И., Тарасенко Ф. П.* Основы системного анализа: Учебник. Томск: Изд-во НТЛ, 1997. 396 с.
6. *Шрайбер Т. Дж.* Моделирование на GPSS: Пер. с англ. М.: Машиностроение, 1980. 592 с.
7. *Боев В. Д.* Моделирование систем. Инструментальные средства GPSS World. СПб: БХВ-Петербург, 2004. 368 с.
8. IEEE Std 1076-2008. IEEE Standard VHDL Language Reference Manual Approved: 26 Sept. 2008. IEEE SA-Standards Board.
9. *Боргест Н. М.* Онтология проектирования: теоретические основы. Ч. 1. Понятия и принципы: Учеб. пособие. Самара: Изд-во СГАУ, 2010. 92 с.
10. *Шильников П. С.* Представление данных КИП // Lexa CAD/CAM/CAE System: Электронный журнал (МГТУ им. Н. Э. Баумана). 2009.
11. *Норенков И. П.* Основы автоматизированного проектирования: Учебник. М.: Изд-во МГТУ им. Н. Э. Баумана, 2002. 336 с.
12. *Купер А., Рейман Р., Кронин Д.* Алан Купер об интерфейсе. Основы проектирования и взаимодействия: Пер. с англ. СПб: Символ-Плюс, 2009. 688 с.
13. *Мальшев А. В., Городецкий В. И., Карсаев О. В., Самойлов В. В., Тихомиров В. В., Маньков Е. В.* Многоагентная система для разрешения конфликтных ситуаций в воздушном пространстве // Тр. СПИИРАН. 2006. Вып. 3, т. 1.
14. *Городецкий В. И., Троицкий Д. В.* Сценарная модель и язык описания знаний для оценки и прогнозирования ситуаций // Тр. СПИИРАН. 2009. № 8. С. 93—127.

Сведения об авторе

Александр Эдуардович Зянчурин — аспирант; СПИИРАН; лаборатория информационных технологий в системном анализе и моделировании; E-mail: zwilas@gmail.com

Поступила в редакцию
27.08.18 г.

Ссылка для цитирования: Зянчурин А. Э. Модельно-ориентированный способ проектирования пользовательских интерфейсов интеллектуальных систем // Изв. вузов. Приборостроение. 2018. Т. 61, № 11. С. 972—977.

MODEL-ORIENTED APPROACH TO DESIGNING USER INTERFACES FOR INTELLIGENT SYSTEMS

A. E. Zyanchurin

St. Petersburg Institute for Informatics and Automation of the RAS,
199178, St. Petersburg, Russia
E-mail: zwilas@gmail.com

A model-oriented approach to the design of user interfaces for expert systems is proposed. Several provisions are given to substantiate the necessity of using declarative-procedural languages to describe user interfaces using a model-oriented approach to system design. A conceptual model of user interface for expert system is developed to describe the basic properties and components of the interface. A concept of dynamic reconfiguration of the user interface for the state of the computing process is presented. The proposed approach is considered as an initial step in formation of a new methodology for designing user interfaces using languages with visual notation.

Keywords: expert systems, adaptive user interface, knowledge representation languages, cognitive image, model-oriented approach

REFERENCES

1. Okhtilev M.Yu. *Osnovy teorii avtomatizirovannogo analiza izmeritel'noy informatsii v real'nom vremeni. Sintez sistemy analiza* (Fundamentals of the Theory of Automated Analysis of Measurement Information in Real Time. Synthesis of Analysis System), St. Petersburg, 1999, 161 p. (in Russ.)
2. Pospelov D.A., ed., *Iskusstvennyy intellekt: Spravochnik. Kn. 2. Modeli i metody* (Artificial Intelligence: Reference book. Book 2. Models and Methods), Moscow, 1990, 304 p. (in Russ.)
3. Narin'yani A.S. *Information Technologies* (Informacionnye Tehnologii), 1997, pp. 17–22. (in Russ.)
4. Tyugu E.Kh. *Kontseptual'noye programmirovaniye* (Conceptual Programming), Moscow, 1984, 256 p. (in Russ.)
5. Peregudov F.I., Tarasenko F.P. *Osnovy sistemnogo analiza* (Basics of System Analysis), Tomsk, 1997, 396 p. ISBN 5-89503-004-1 (in Russ.)
6. Schriber T.J. *Simulation Using GPSS*, NY, Wiley, 1974.
7. Boyev V.D. *Modelirovaniye sistem. Instrumental'nyye sredstva GPSS* (World Systems Modeling. GPSS World Tools), St. Petersburg, 2004, 368 p. (in Russ.)
8. *IEEE Std 1076-2008 IEEE Standard VHDL Language Reference Manual Approved: 26 September 2008* IEEE SA-Standards Board.
9. Borgest N.M. *Ontologiya proyektirovaniya: teoreticheskiye osnovy. Chast' 1. Ponyatiya i printsipy* (Design ontology: Theoretical Bases. Part 1. Concepts and Principles), Samara, 2010, 92 p. (in Russ.)
10. Shil'nikov P.S. *Predstavleniye dannykh KIP* (Presentation of TRC data), Moscow, 2005, 150 p. (in Russ.)
11. Norenkov I.P. *Osnovy avtomatizirovannogo proyektirovaniya* (Bases of the Automated Design), Moscow, 2002, 336 p. ISBN 5-7038-2090-1 (in Russ.)
12. Cooper A., Reimann R., and Cronin D. *About Face 3: The Essentials of Interaction Design*, Wiley Publishing, 2007, 688 p.
13. Malyshev A.V., Gorodetskiy V.I., Karsayev O.V., Samoylov V.V., Tikhomirov V.V., Man'kov E.V. *Trudy SPIIRAN* (SPIIRAS Proceedings), 2006, no. 3(1).
14. Gorodetskiy V.I., Troitskiy D.V. *Trudy SPIIRAN* (SPIIRAS Proceedings), 2009, no. 8, pp. 93–127. (in Russ.)

Data on author

Alexander E. Zyanchurin

— Post-Graduate Student; St. Petersburg Institute for Informatics and Automation of the RAS, Laboratory of Information Technologies in Systems Analysis and Modeling; E-mail: zwilas@gmail.com

For citation: Zyanchurin A. E. Model-oriented approach to designing user interfaces for intelligent systems. *Journal of Instrument Engineering*. 2018. Vol. 61, N 11. P. 972—977 (in Russian).

DOI: 10.17586/0021-3454-2018-61-11-972-977