

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

INTELLIGENT SYSTEMS



УДК 004.514, 004.822.2
<https://doi.org/10.37661/1816-0301-2023-20-3-74-89>

Оригинальная статья
Original Paper

Семантические модели и средства проектирования адаптивных пользовательских интерфейсов интеллектуальных систем

М. Е. Садовский

*Белорусский государственный университет
информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
E-mail: sadovski@bsuir.by*

Аннотация

Цели. Предлагаются модели и средства проектирования адаптивных пользовательских интерфейсов интеллектуальных систем. Актуальность работы определяется необходимостью снижения накладных расходов и сроков разработки пользовательских интерфейсов, обеспечения их адаптации под особенности конкретного пользователя интеллектуальной системы.

Методы. Анализируются существующие подходы к проектированию пользовательских интерфейсов. Предлагается семантическая модель адаптивного пользовательского интерфейса интеллектуальных систем, реализованная при помощи базового универсального языка представления знаний, который основан на теории множеств и теории графов.

Результаты. Разработаны модель адаптивного пользовательского интерфейса интеллектуальных систем, которая включает модель базы знаний пользовательского интерфейса, агентно-ориентированную модель пользовательского интерфейса и библиотеку многократно используемых компонентов, обеспечивающих интеграцию пользовательского интерфейса в отдельные интеллектуальные системы и коллективы интеллектуальных систем, а также метод передачи компонентов пользовательского интерфейса в рамках коллектива интеллектуальных систем в процессе эксплуатации интеллектуальной системы.

Заключение. Представленные модели и средства позволяют упростить повторное (многократное) применение компонентов пользовательского интерфейса совместно с компонентами баз знаний и решателей задач при проектировании и разработке как отдельной интеллектуальной системы, так и коллектива семантически совместимых интеллектуальных систем, обеспечить автоматизацию интеграции пользовательских интерфейсов и их адаптацию для каждого пользователя. Набор компонентов пользовательского интерфейса был включен в библиотеку многократно применяемых компонентов пользовательского интерфейса. В дальнейшем планируется расширение набора компонентов указанной библиотеки и их внедрение в коллектив семантически совместимых интеллектуальных систем.

Ключевые слова: адаптивный пользовательский интерфейс, интеллектуальная система, коллектив интеллектуальных систем, персональный ассистент, онтология, семантическая модель

Для цитирования. Садовский, М. Е. Семантические модели и средства проектирования адаптивных пользовательских интерфейсов интеллектуальных систем / М. Е. Садовский // Информатика. – 2023. – Т. 20, № 3. – С. 74–89. <https://doi.org/10.37661/1816-0301-2023-20-3-74-89>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 14.07.2023

Подписана в печать | Accepted 14.08.2023

Опубликована | Published 29.09.2023

Semantic models and tools for designing adaptive user interfaces of intelligent systems

Mikhail E. Sadouski

*Belarusian State University of Informatics and Radioelectronics,
st. P. Brovki, 6, Minsk, 220013, Belarus
E-mail: sadovski@bsuir.by*

Abstract

Objectives. Models and tools for designing adaptive user interfaces for intelligent systems are being developed. The relevance is determined by the need to reduce overhead costs and development time for user interfaces and to provide their adaptation to the specific characteristics of the user of the intelligent system.

Methods. Existing approaches to designing user interfaces are being analyzed. A semantic model of an adaptive user interface for intelligent systems is proposed, implemented using a basic universal language for representing knowledge based on set theory and graph theory.

Results. An adaptive user interface model for intelligent systems has been developed, which includes a knowledge base model of the user interface, an agent-oriented model of the user interface, and a library of reusable components that provide integration of the user interface into both individual intelligent systems and intelligent systems groups. A method of transferring user interface components within an intelligent systems group during the operation of an intelligent system has also been developed.

Conclusion. Developed models and tools allow to simplify the reuse of user interface components together with knowledge base and problem-solving components in the design and development of individual intelligent systems, as well as in the design and development of a group of semantically compatible intelligent systems, ensuring automation of integration of user interfaces and their adaptation for each user. The developed set of user interface components has been included in a library of reusable user interface components. Further expansion of the set of components in the library and their integration into a group of semantically compatible intelligent systems is planned.

Keywords: adaptive user interface, intelligent system, intelligent systems group, personal assistant, ontology, semantic model

For citation. Sadouski M. E. *Semantic models and tools for designing adaptive user interfaces of intelligent systems*. Informatika [Informatics], 2023, vol. 20, no. 3, pp. 74–89 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-3-74-89>

Conflict of interest. The author declares of no conflict of interest.

Введение. Организация взаимодействия пользователей с компьютерными системами (в том числе и интеллектуальными) оказывает существенное влияние на эффективность автоматизации человеческой деятельности, опыт и уровень удовлетворенности пользователей.

Одним из ключевых свойств интеллектуальных компьютерных систем нового поколения является их интероперабельность – способность к эффективному взаимодействию. Такие системы являются автономными и самодостаточными субъектами деятельности наравне с человеком.

Однако в основе современной организации взаимодействия пользователя с компьютерной системой лежит парадигма грамотного пользователя, который знает, как управлять системой, и несет полную ответственность за качество взаимодействия с ней [1]. Многообразие форм и видов интерфейсов подводит пользователя к необходимости адаптироваться к каждой конкретной системе, обучаться принципам взаимодействия с ней для решения поставленных задач.

На современном этапе развития искусственного интеллекта для повышения эффективности взаимодействия необходим переход от парадигмы грамотного управления используемым инструментом к парадигме равноправного сотрудничества, партнерского взаимодействия интеллектуальной системы со своим пользователем. Дружественность пользовательского интерфейса должна заключаться в адаптивности системы к особенностям и квалификации пользователя, исключении любых проблем для пользователя в процессе диалога с интеллектуальной системой, постоянном совершенствовании коммуникационных навыков пользователя. Следовательно, необходимо отойти от привычной адаптации пользователя к системе (путем обучения ее использованию) в сторону адаптации самого интерфейса под цели, задачи и характеристики конкретного пользователя в режиме реального времени [2].

Актуальность выбранной темы обусловлена необходимостью снижения накладных расходов и сроков разработки пользовательских интерфейсов, обеспечения их адаптации под особенности конкретного пользователя. Поскольку пользователь любой системы общается с ней посредством пользовательского интерфейса, то проблемы, связанные с интерфейсом, часто формируют негативное мнение о всей системе в целом и не позволяют в полной мере использовать ее функционал. Особенно актуальна данная ситуация для интеллектуальных систем, возможности которых значительно шире, чем у традиционных систем [3].

Чтобы удовлетворить потребности каждого пользователя, пользовательский интерфейс должен быть адаптирован к конкретному человеку, что осуществляется с помощью адаптивного пользовательского интерфейса (АПИ). В отличие от адаптированных пользовательских интерфейсов, которые адаптированы к конечному пользователю при проектировании и не изменяются во время эксплуатации системы, адаптивные пользовательские интерфейсы изменяются на основе потребностей пользователя или контекста использования в процессе эксплуатации системы. На сегодняшний день существуют различные подходы и фреймворки для разработки АПИ. Однако эти подходы ограничены конкретными областями применения, а не обобщенным подходом, применимым ко всем областям, а также обеспечивают низкий уровень адаптивности пользовательского интерфейса, который может быть существенно повышен.

Целью настоящего исследования является разработка моделей и средств проектирования АПИ, обеспечивающих комплексный универсальный подход к проектированию АПИ интеллектуальных систем. В соответствии с поставленной целью выделены следующие задачи исследования:

- анализ современных подходов к проектированию АПИ;
- формулирование требований к моделям и средствам проектирования АПИ;
- разработка модели проектирования АПИ;
- разработка средства проектирования АПИ.

В рамках исследования рассматривается переход от проектирования адаптированных пользовательских интерфейсов к проектированию АПИ и предлагаются модели, методы и средства для проектирования таких интерфейсов. Уточняются требования к моделям и средствам проектирования АПИ и предлагается подход к проектированию АПИ на основе семантических моделей и средств.

Проектирование адаптированного пользовательского интерфейса. Адаптированный пользовательский интерфейс предполагает создание пользовательского интерфейса, который не будет изменяться в процессе эксплуатации. Часто такой подход к проектированию пользовательских интерфейсов называют классическим [4]. Он предполагает построение макета интерфейса на основании выявленных пользовательских требований, затем создание прототипа, определение структуры диалога и проработку возможных сценариев развития диалога с дальнейшей реализацией интерфейса с помощью подходящей среды программирования.

Один из наиболее важных шагов при проектировании адаптированных интерфейсов – анализ пользователей (целевой аудитории). При разработке такого интерфейса необходимо понимать, какие цели должны достичь пользователи и какие возможности должен предоставлять интерфейс. В данном случае интерфейс будет направлен на решение определенных задач для определенной группы пользователей вне зависимости от личностных характеристик пользователя и контекста использования.

В большинстве случаев проектированием пользовательского интерфейса занимается UI/UX-дизайнер. При создании интерфейса он должен руководствоваться общими принципами проектирования пользовательских интерфейсов, различными стандартами, правилами и руководствами по стилю.

Стоит отметить, что конечный интерфейс пользователя в большей степени может быть адаптирован под различные разрешения экранов устройств (например, мобильного устройства, планшета, компьютера), однако эта адаптация предопределена на этапе создания макета пользовательского интерфейса и разработки.

Проектирование адаптируемого пользовательского интерфейса. В адаптируемом пользовательском интерфейсе любая адаптация является предопределенной и может изменяться пользователем как перед началом работы с системой, так и во время эксплуатации.

Процесс проектирования и разработки адаптируемых пользовательских интерфейсов практически совпадает с проектированием и разработкой адаптированных интерфейсов. Возможные адаптации также предопределены заранее, однако пользователь сам решает, в какой момент некоторые из этих адаптаций использовать. Как и в адаптированных интерфейсах, адаптация под разрешение экрана устройства предопределена заранее и пользователь никак не может на нее повлиять.

Недостаток адаптируемого интерфейса заключается в том, что пользователю необходимо достаточно хорошо быть знакомым как с системой, так и со средствами, позволяющими изменить ее интерфейс.

Проектирование адаптивного пользовательского интерфейса. Адаптивность предполагает постоянное улучшение пользовательского интерфейса, которое происходит во время эксплуатации системы и зависит от поведения пользователя, а также его изменение на основе потребностей пользователя или контекста использования.

Как правило, контекст использования включает знания [5]:

- о пользователе (предпочтениях, целях и задачах, физическом или эмоциональном здоровье);
- об окружающей среде (температуре, освещенности, шуме, местоположении);
- о платформе (разрешении экрана, емкости аккумулятора, возможности подключения к сети Интернет, браузеру).

Вне зависимости от средств создания АПИ интеллектуальные системы должны эффективно хранить и обрабатывать все указанные выше знания.

В настоящее время существует несколько основных подходов к представлению модели пользовательского интерфейса:

- подход на основе специализированных языков описания;
- контекстно-зависимый;
- моделиориентированный;
- онтологический.

Подход на основе специализированных языков описания предполагает представление конкретного пользовательского интерфейса в платформенно независимом виде. В качестве примеров языков описания интерфейса можно привести UIML, UsiXML, XForms (URL: <https://www.w3.org/TR/xforms>) и JavaFX FXML (URL: https://openjfx.io/javadoc/12/javafx.fxml/javafx/fxml/doc-files/introduction_to_fxml.html) [6, 7]. Ключевой идеей представленных языков является создание модели диалогов и форм интерфейса в независимом от используемой технологии виде, описание визуальных элементов, а также взаимосвязей между ними и их свойств для создания конкретного пользовательского интерфейса.

Контекстно-зависимый подход интегрирует использование структурного описания интерфейса на основе языков описания с поведенческой спецификацией, т. е. генерация интерфейса основывается на действиях пользователя. В рамках подхода специфицируются переходы между различными видами конкретного пользовательского интерфейса. В качестве примеров реализации такого подхода можно привести языки CAP3 и MARIA [8, 9].

В рамках *моделеориентированного подхода* [10] модель является базовым средством разработки интерфейса, содержащим декларативные описания высокого уровня абстракции и не содержащим процедурный код. Наборы моделей различны для каждого моделеориентированного средства, также для каждого средства и уровня модели различаются декларативные языки. Это затрудняет не только создание интерфейса в рамках одного моделеориентированного средства, но и его последующее модифицирование и приводит к сильной зависимости разработанной модели от инструментального средства ее разработки.

Онтологический подход является развитием моделеориентированного подхода, в рамках которого онтологии используются в качестве способа представления информации о конкретном пользовательском интерфейсе. Например, по аналогии с подходом на основе специализированных языков описания был предложен фреймворк [11], применяющий онтологию для описания пользовательского интерфейса на основе понятий, хранящихся в базе знаний. По аналогии с контекстно-зависимым подходом в рамках работы [12] используется модель предметной области совместно с моделью пользовательского интерфейса, ассоциированная с онтологией действий. Проект ActiveRaUL [13] совмещает UIML с моделеориентированным подходом. В рамках данного проекта онтологическая модель предметной области сопоставляется с онтологическим представлением пользовательского интерфейса. Подход, предложенный в работе [14], совмещает данные приложения с онтологией пользовательского интерфейса для создания единого описания в базе знаний с целью последующей автоматической генерации различных вариантов интерфейса для приложений-опросников с готовыми сценариями взаимодействия с пользователем. Следует также отметить работы [15, 16], в рамках которых предложена концепция, позволяющая объединять однородную по содержанию информацию в компоненты модели интерфейса, освобождать разработчика интерфейса от кодирования и формировать информацию для каждого компонента модели интерфейса с помощью редакторов, управляемых соответствующими моделями онтологий.

Именно онтологический подход позволяет:

- создавать наиболее полное унифицированное описание различных аспектов пользовательского интерфейса;
- легко интегрировать различные аспекты пользовательского интерфейса;
- упрощать повторное использование модели интерфейса.

Большинство современных систем и фреймворков для проектирования АПИ используют онтологическую модель [11, 12, 15–17], а также декларативные языки описания пользовательского интерфейса [6, 7, 9, 18]. Однако для существующих решений остаются актуальными проблема интеграции пользовательского интерфейса с другими компонентами системы и трудоемкость переноса пользовательского интерфейса из одной системы в другую.

Модели и средства проектирования пользовательских интерфейсов традиционных компьютерных систем существенно отличаются от моделей и средств, используемых для проектирования остальных компонентов системы, что приводит к значительным затратам при интеграции спроектированного и разработанного пользовательского интерфейса в систему.

В рамках настоящей работы рассматривается проектирование пользовательского интерфейса интеллектуальных систем, которые включают базу знаний, решатель задач и пользовательский интерфейс.

Особенностями интеллектуальных систем являются:

- возможность унифицированного представления в рамках системы различных видов знаний, моделей обработки знаний и интерфейсов, что обеспечивает совместимость ее компонентов;
- отсутствие многообразия языков представления и обработки знаний;
- способность приобретать новые знания и навыки в процессе эксплуатации, сохраняя при этом корректность и целостность базы знаний.

Трудоемкость переноса пользовательского интерфейса из одной системы в другую обусловлена отсутствием общих принципов проектирования различных интеллектуальных систем (использованием несовместимых моделей и средств при их проектировании и разработке). В этой связи важной задачей является сохранение унифицированного представления пользовательского интерфейса в рамках коллектива интеллектуальных систем, что позволило бы упростить повторное (многократное) применение компонентов пользовательского интерфейса совместно с компонентами баз знаний и решателей задач для любой интеллектуальной системы в рамках указанного коллектива.

Предлагаемый подход. В качестве основы для проектирования пользовательских интерфейсов используется онтологический подход на базе семантической модели, предполагающий разработку формальной онтологии пользовательских интерфейсов интеллектуальных систем. Создание такой онтологии способствует:

- согласованию принципов и методов построения компонентов пользовательского интерфейса;
- унификации интерфейсной деятельности пользователя и интеллектуальной системы;
- декомпозиции процесса проектирования пользовательских интерфейсов и возможности его распараллеливания.

Ключевые принципы предлагаемого подхода заключаются в следующем:

- пользовательский интерфейс предлагается рассматривать в качестве встраиваемой интеллектуальной системы, ориентированной на решение интерфейсных задач пользователя и состоящей из базы знаний и решателя задач;

- при проектировании пользовательского интерфейса должны применяться те же модели и средства, которые используются при проектировании самой интеллектуальной системы. Это позволит легко интегрировать пользовательский интерфейс как в отдельную интеллектуальную систему, так и в коллектив семантически совместимых интеллектуальных систем и использовать те же методики проектирования баз знаний и решателей задач, которые применяются при проектировании интеллектуальной системы;

- при проектировании пользовательского интерфейса предлагается использовать компонентный подход, что даст возможность сократить затраты на последующее проектирование пользовательского интерфейса за счет повторного применения ранее спроектированных компонентов и создания библиотеки многократно используемых компонентов пользовательского интерфейса;

- модель базы знаний пользовательского интерфейса должна быть легко расширяемой и включать в себя [19]:

- модель пользовательского интерфейса и его компонентов;
- модель контекста использования (модель пользователя, действий пользователя, окружающей среды, платформы);
- правила адаптации пользовательского интерфейса;
- правила оценки пользовательского интерфейса;
- методику проектирования пользовательского интерфейса;

- модель решателя задач пользовательского интерфейса должна быть основана на многоагентном подходе и должна включать агента оценки пользовательского интерфейса, адаптации пользовательского интерфейса, анализа действий пользователя, предсказания действий пользователя;

- каждый компонент пользовательского интерфейса должен быть внешним отображением определенного элемента из базы знаний, что позволит использовать его в качестве аргументов пользовательских команд и правильно трактовать прагматику и семантику объектов интерфейсной деятельности, даст возможность пользователю задавать системе вопросы касательно любого из компонентов интерфейса;

- должны быть реализованы средства редактирования базы знаний и средства помощи для проектировщика и разработчика пользовательского интерфейса;

- описание модели базы знаний и решателя задач пользовательского интерфейса предлагается осуществлять на основе универсального унифицированного языка представления знаний, что обеспечит совместимость между этими компонентами.

Реализация подхода. В соответствии с предлагаемым подходом технология для проектирования и реализации АПИ должна поддерживать компонентный подход для создания семантических моделей, позволять осуществлять простую интеграцию различных семантических моделей в рамках единой системы и в рамках коллектива интеллектуальных систем, предоставлять возможность описания различных семантических моделей и составляющих их различных видов знаний в едином унифицированном виде.

В качестве используемой технологии была выбрана технология OSTIS, которая соответствует указанным требованиям. Она позволяет создавать семантически совместимые системы (ostis-системы), которые способны обрабатывать запросы и задачи пользователей, учитывая их контекст и смысл. Это достигается за счет использования семантических сетей, которые дают возможность описывать знания и связи между ними. Кроме того, технология OSTIS обеспечивает масштабируемость и гибкость системы, что позволяет ей адаптироваться к изменениям в поведении пользователей и их потребностях [20].

В качестве основы представления знаний в рамках технологии OSTIS используется унифицированный вариант кодирования информации любого рода на основе семантических сетей с базовой теоретико-множественной интерпретацией, названный SC-кодом [21]. Элементы текстов SC-кода (sc-текстов) названы sc-элементами, среди которых, в свою очередь, выделяются sc-узлы, sc-дуги и sc-ребра. В рамках технологии также предложено несколько универсальных вариантов визуализации конструкций SC-кода, таких как SCg-код (графический вариант), SCn-код (нелинейный гипертекстовый вариант) и SCs-код (линейный строковый вариант).

Каждая ostis-система состоит из полной семантической модели этой системы, описанной средствами SC-кода (sc-модели компьютерной системы), и платформы интерпретации sc-моделей, которая в общем случае может быть реализована как программно, так и аппаратно [22]. Таким образом обеспечивается полная платформенная независимость ostis-систем.

Принципы построения sc-моделей баз знаний и sc-моделей решения задач более подробно рассмотрены в работах [23, 24]. В основе sc-модели базы знаний лежат такие основные принципы, как выделение иерархической системы предметных областей и онтологий (включая онтологию представления и семейство онтологий верхнего уровня, которые входят в состав каждой разрабатываемой ostis-системы), а также выделение структур (знаков целых фрагментов базы знаний), которые могут быть впоследствии описаны в рамках этой же базы знаний. Использование этих и других принципов позволяет обеспечить в базе знаний возможность представления знаний различных видов, легкость внесения изменений в базу знаний, в том числе возможность расширения набора используемых видов знаний, структуризации базы знаний по произвольному набору признаков и представления в базе знаний метазнаний произвольного уровня.

В основе sc-модели решателя задач лежит принцип, согласно которому решатель трактуется как иерархическая система агентов, реагирующих на ситуации и события в sc-памяти (sc-агентов) и взаимодействующих между собой исключительно путем спецификации в sc-памяти информационных процессов, выполняемых соответствующими агентами. Подобные sc-агенты могут быть атомарными, т. е. такими, для которых явно указывается программа их действий, и неатомарными, которые декомпозируются на более простые sc-агенты. Классы функционально эквивалентных sc-агентов названы абстрактными sc-агентами. Каждый абстрактный sc-агент имеет соответствующую спецификацию, содержащую как минимум условие инициирования sc-агента и описание реализации sc-агента в зависимости от того, атомарный это sc-агент или неатомарный [25].

В рамках технологии OSTIS реализованы средства поддержки проектирования интеллектуальных систем (метасистема OSTIS, URL: <https://ims.ostis.net>) и средства редактирования модели базы знаний, рассматриваемые далее.

При проектировании адаптивных пользовательских интерфейсов ostis-систем для создания и редактирования семантической модели интерфейса используются такие средства, как редактор KBE (Knowledge Base source Editor, URL: <https://github.com/ostis-ai/kbe>) и онлайн-редактор SCg-кода (SCg-редактор), который является частью реализации интерпретатора sc-моделей пользовательских интерфейсов (URL: <https://github.com/ostis-ai/sc-web>).

КВЕ представляет собой приложение, которое направлено на помощь в создании и редактировании фрагментов баз знаний (фрагментов семантической модели) интеллектуальных систем, проектирование которых основано на технологии OSTIS. В основу данного редактора заложен принцип визуализации знаний, хранящихся в базе знаний ostis-системы, что намного упрощает процесс их редактирования и ускоряет проектирование баз знаний.

Пользовательский интерфейс инструмента содержит главное окно, в котором пользователь может создавать вкладки. В каждой вкладке возможно редактирование различных файлов исходных текстов баз знаний, представленных с помощью SCg-кода.

Основная идея, которая преследуется в КВЕ, – это упрощение и ускорение процесса редактирования семантической модели интеллектуальной системы.

В процессе редактирования пользователю доступны четыре режима:

выделения и создания sc.g-узлов – пользователь может работать со всеми объектами, выделяя и перемещая их, вызывая контекстное меню с командами;

создания sc.g-дуг – начинается с того, что пользователь указывает объект, из которого будет выходить дуга. Далее он может указать точки излома дуги, завершается создание указанием конечного объекта. В процессе создания пользователь может отменять последнее действие (указание начального объекта, точки излома);

создания sc.g-шин – используется для увеличения контактной площади узла, поэтому sc.g-шины могут создаваться лишь для sc.g-узлов. Создание шины начинается с указания sc.g-узла, далее, как и при создании sc.g-дуг, указываются точки излома. Пользователь может отменять последнее действие нажатием правой клавиши мыши;

создания sc.g-контуров – начинается с указания первой точки контура. Далее, как и в случае с sc.g-дугами и sc.g-шинами, указываются точки излома. Стоит отметить, что все объекты, которые попадут внутрь созданного контура, будут добавлены в него автоматически. Как и при создании дуг и шин, пользователь может отменять последнее действие [1].

Онлайн-редактор SCg-кода обладает схожим с КВЕ функционалом, но при этом позволяет редактировать семантическую модель в режиме реального времени при эксплуатации интеллектуальной системы и без создания файлов исходных текстов базы знаний.

Помимо четырех режимов редактирования, указанных для редактора КВЕ, SCg-редактор содержит также отдельный режим создания и редактирования файлов ostis-систем и имеет возможность сохранять изменения модели в базе знаний интеллектуальной системы.

Модель базы знаний пользовательского интерфейса была реализована с применением указанных средств. Так, семантическая модель компонентов пользовательского интерфейса описывает классификацию компонентов пользовательского интерфейса и связи между ними (рис. 1). Множество компонентов пользовательского интерфейса включает в себя множество компонентов пользовательского интерфейса для отображения, которое в свою очередь включает множество компонентов вывода.

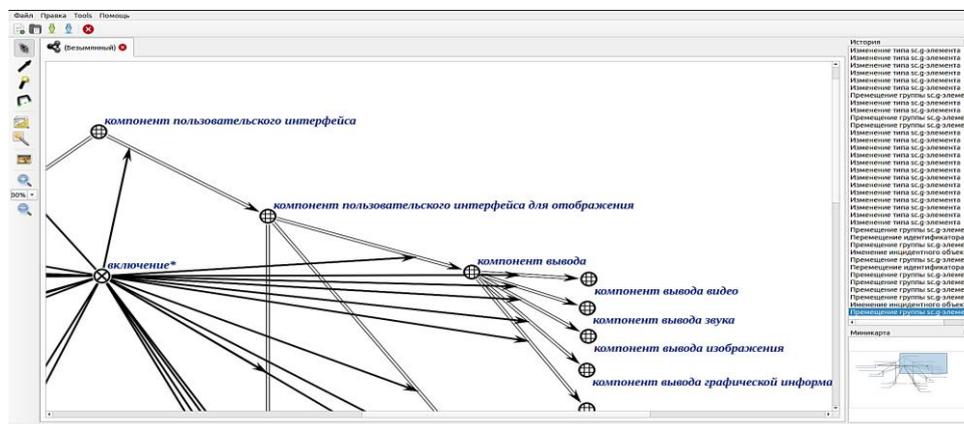


Рис. 1. Фрагмент семантической модели компонентов пользовательского интерфейса

Fig. 1. Fragment of the semantic model of user interface components

Детальное описание семантической модели адаптивного пользовательского интерфейса, его предметных областей и онтологий дано в работе [26]. Реализованная модель легко расширяется и модифицируется.

В рамках технологии OSTIS вводится понятие экосистемы OSTIS [27].

Экосистема OSTIS – это социотехническая сеть, которая представляет собой коллектив взаимодействующих самих ostis-систем, пользователей указанных ostis-систем (как конечных пользователей, так и разработчиков) и некоторых компьютерных систем, не являющихся ostis-системами (в качестве дополнительных информационных ресурсов или сервисов).

Задачами экосистемы OSTIS являются:

- оперативное внедрение всех согласованных изменений ostis-систем;
- перманентная поддержка высокого уровня взаимопонимания всех систем, входящих в экосистему OSTIS, и всех их пользователей;
- корпоративное решение различных сложных задач, требующих координации деятельности нескольких (чаще всего априори неизвестных) ostis-систем, а также, возможно, некоторых пользователей.

В рамках экосистемы OSTIS выделяется понятие персонального ostis-ассистента. Персональный ostis-ассистент – это ostis-система, являющаяся персональным ассистентом пользователя в рамках экосистемы OSTIS. Такая система предоставляет возможности:

- анализа деятельности пользователя и формирования рекомендаций по ее оптимизации;
- адаптации под настроение пользователя, его личностные качества, общую окружающую обстановку и задачи, которые чаще всего решает пользователь;
- перманентного обучения самого ассистента в процессе решения новых задач, при этом обучаемость потенциально не ограничена;
- ведения диалога с пользователем на естественном языке, в том числе в речевой форме;
- ответа на вопросы различных классов, при этом если системе что-то непонятно, то она сама может задавать встречные вопросы;
- автономного получения информации от всей окружающей среды, а не только от пользователя (в текстовой или речевой форме).

Пользовательский интерфейс персонального интеллектуального ассистента предоставляет пользователю средства управления его индивидуальной деятельностью, осуществляемой совместно с соответствующим ему персональным интеллектуальным ассистентом и обеспечивает унифицированный характер взаимодействия пользователей в рамках различных сообществ, в которые он входит. Простейшим видом сообществ является разовый диалог двух пользователей.

Поскольку взаимодействие пользователя с экосистемой OSTIS происходит только через персонального ostis-ассистента, пользовательский интерфейс экосистемы OSTIS для пользователя есть пользовательский интерфейс его персонального ostis-ассистента. Такой интерфейс должен быть адаптивным [28].

Модель пользователя и его действий в контексте пользовательского интерфейса экосистемы OSTIS необходимо хранить только в рамках персонального ostis-ассистента пользователя и обмениваться этими знаниями с другими ostis-системами по необходимости.

Проектирование пользовательского интерфейса основано на компонентном подходе. Любой компонент пользовательского интерфейса может быть описан в базе знаний ostis-системы. Пример компонента пользовательского интерфейса «калькулятор» с соответствующим ему фрагментом описания в базе знаний ostis-системы в SCg-коде показан на рис. 2.

Пользовательский интерфейс персонального ostis-ассистента по умолчанию содержит лишь компоненты интерфейса, необходимые пользователю для начала взаимодействия. При этом интерфейс постоянно адаптируется на основе модели пользователя и правил адаптации. Компоненты пользовательского интерфейса могут автоматически встраиваться в пользовательский интерфейс персонального ostis-ассистента в процессе эксплуатации системы. Ostis-системы, которые не являются персональными ostis-ассистентами, как правило, не должны иметь пользовательский интерфейс. Однако разработчики таких систем могут добавлять в базу знаний не-

обходимые компоненты пользовательского интерфейса, которые отображаются персональным ostis-ассистентом при взаимодействии с пользователем.

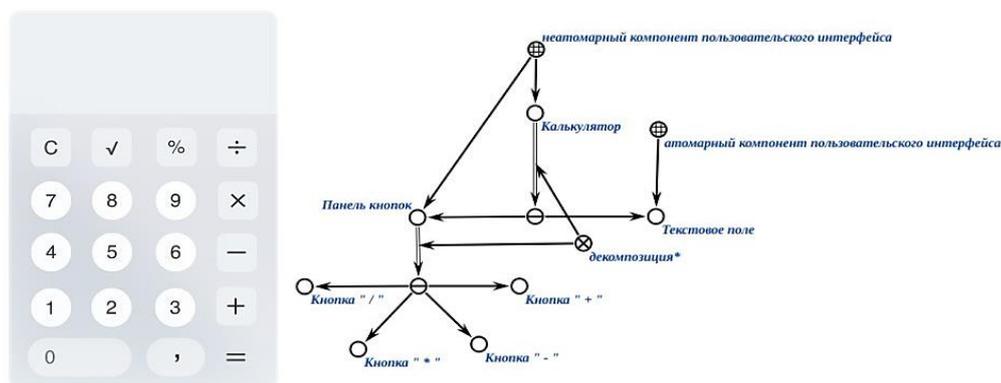


Рис. 2. Компонент «калькулятор» с соответствующим ему фрагментом описания в SCg-коде
Fig. 2. The "calculator" component with its corresponding description fragment in the SCg-code

Так, общий процесс взаимодействия персонального ostis-ассистента с пользователем может быть описан следующим образом:

- пользователю отображается интерфейс персонального ostis-ассистента по умолчанию;
- при взаимодействии с пользователем интерфейс персонального ostis-ассистента автоматически адаптируется;
- персональный ассистент при взаимодействии с пользователем может использовать компоненты пользовательского интерфейса любой ostis-системы в рамках экосистемы OSTIS, адаптировать такой компонент и отображать пользователю при необходимости (рис. 3). Пользователь, его персональный ostis-ассистент и другие ostis-системы являются частью экосистемы OSTIS.

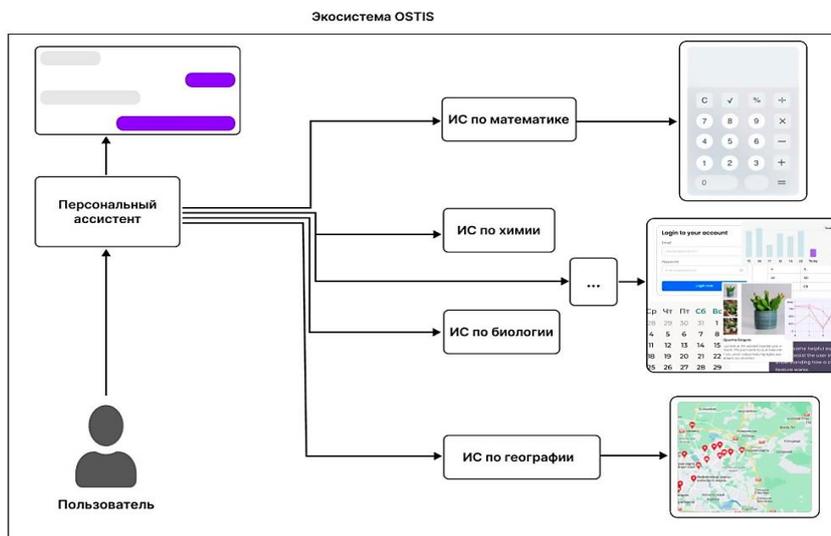


Рис. 3. Пример использования компонента пользовательского интерфейса в рамках экосистемы OSTIS
Fig. 3. An example of user interface component usage within the OSTIS Ecosystem

Разработчик каждой интеллектуальной системы может описать необходимые для этой системы компоненты пользовательского интерфейса. В приведенном примере для системы по математике был описан компонент «калькулятор», для системы по географии – компонент «карты».

Поскольку пользователь взаимодействует со всеми ostis-системами только через персонального ostis-ассистента, ассистент сам обращается к нужной в текущий момент системе и встраивает необходимый компонент пользовательского интерфейса из некоторой прикладной ostis-системы в свою часть базы знаний с целью последующего отображения пользователю с учетом правил адаптации, которые также хранятся в базе знаний персонального ассистента.

Важно отметить, что большое разнообразие пользовательских интерфейсов влечет за собой разработку большого числа компонентов. В качестве многократно используемых компонентов интерфейсов ostis-систем могут выступать как уже спроектированные интерфейсы, так и специфицированные компоненты интерфейсов. Большое число многократно используемых компонентов интерфейсов ostis-систем создает проблему их хранения и поиска. Чтобы решить эту проблему, в технологии OSTIS применяются библиотека многократно используемых компонентов пользовательских интерфейсов ostis-систем и менеджер многократно используемых компонентов ostis-систем [29]. Применение библиотеки многократно используемых компонентов интерфейсов ostis-систем при проектировании интерфейса прикладной системы дает возможность значительно сократить сроки проектирования, а также снизить требования, предъявляемые к начальной квалификации разработчика. Это достигается за счет проектирования интерфейса из уже заранее подготовленных моделей интерфейса, что также позволяет повысить качество проектируемого интерфейса.

Уникальным свойством технологии OSTIS является обеспечение совместимости компонентов баз знаний, решателей задач и интерфейсов ostis-систем за счет единой унифицированной формальной основы.

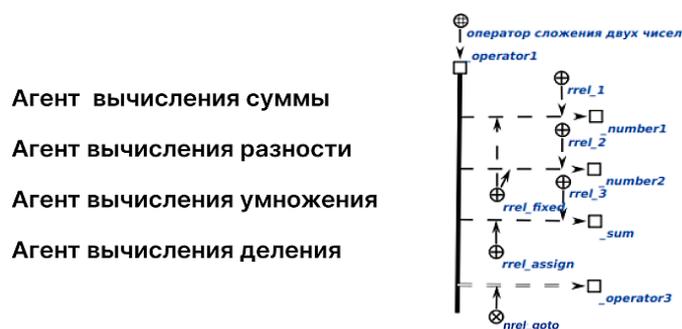


Рис. 4. Фрагмент представления оператора суммы двух чисел в базе знаний

Fig. 4. Fragment of the representation of the operator of the sum of two numbers in the knowledge base

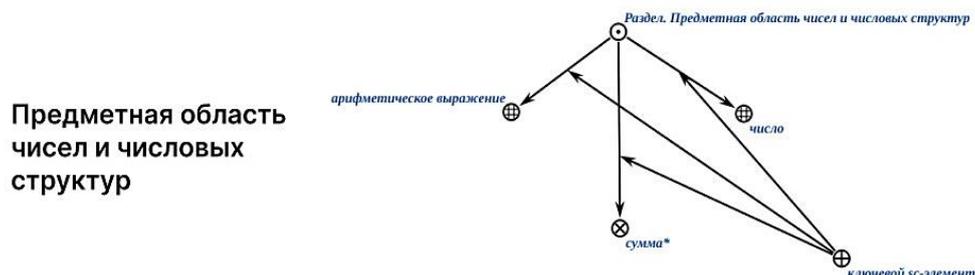


Рис. 5. Фрагмент предметной области чисел и числовых структур

Fig. 5. Fragment of the subject domain of numbers and numerical structures

Так, компонент пользовательского интерфейса зачастую для работы должен включать не только описание в базе знаний своих визуальных характеристик, но и компоненты решателя задач (sc-агенты), а также необходимые фрагменты некоторой предметной области. На рис. 2, 4 и 5 изображены фрагменты компонентов интерфейса, решателя задач и базы знаний соответственно для компонента пользовательского интерфейса «калькулятор». Данные фрагменты

включают оператор суммы двух чисел и фрагмент предметной области чисел и числовых структур, необходимые для работы компонента «калькулятор». За счет унифицированного представления всех необходимых частей такой компонент может быть легко интегрирован в любую *ostis*-систему, в том числе систему «персональный *ostis*-ассистент».

Заключение. В рамках исследования был рассмотрен переход от проектирования адаптированных пользовательских интерфейсов к проектированию АПИ. В качестве основы для проектирования АПИ интеллектуальных систем был предложен онтологический подход на основе семантической модели, предполагающий разработку формальной онтологии пользовательских интерфейсов интеллектуальных систем. Предложенный подход позволяет упростить повторное (многократное) использование компонентов пользовательского интерфейса совместно с компонентами баз знаний и решателей задач как для отдельной интеллектуальной системы, так и для коллектива семантически совместимых интеллектуальных систем, обеспечить автоматизацию интеграции пользовательских интерфейсов и их адаптацию для каждого пользователя.

С целью апробации семантической модели АПИ интеллектуальных систем был разработан набор компонентов пользовательского интерфейса, включенных в библиотеку многократно используемых компонентов пользовательского интерфейса. В дальнейшем планируется расширение набора компонентов указанной библиотеки и их внедрение в коллектив семантически совместимых интеллектуальных систем.

Список использованных источников

1. Садовский, М. Е. Общие принципы организации интерфейсов *ostis*-систем / М. Е. Садовский // Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения / под общ. ред. В. В. Голенкова. – Минск, 2023. – Гл. 4.1. – С. 417–432.
2. Фомина, Т. А. Проектирование адаптивного интерфейса ИС для поддержки деятельности образовательного учреждения / Т. А. Фомина, Г. М. Новикова // Вестник Алтайской академии экономики и права. – 2020 – Т. 6, № 1. – С. 125–133. <https://doi.org/10.17513/vaael.1174>
3. *Ontology-based design of intelligent systems user interface* / A. S. Boriskin [et al.] // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2017) : материалы Междунар. науч.-техн. конф., Минск, 16–18 февр. 2017 г. / редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2017. – С. 95–106.
4. Белоусова, С. А. Анализ подходов к созданию пользовательского интерфейса / С. А. Белоусова, Ю. И. Рогозов // Изв. ЮФУ. Техн. науки. – 2014. – № 6(155). – С. 142–148.
5. *Model-based adaptive user interface based on context and user experience evaluation* / J. Hussain [et al.] // J. on Multimodal User Interfaces. – 2018. – Vol. 12. – P. 1–16. <https://doi.org/10.1007/s12193-018-0258-2>
6. *UIML: An appliance-independent XML user interface language* / M. Abrams [et al.] // Proceedings of the 8th Intern. Conf. on the World Wide Web, Toronto, 11–14 May 1999. – Toronto, 1999. – P. 1695–1708.
7. *Limbourg, Q. USIXML: A user interface description language supporting multiple levels of independence* / Q. Limbourg, J. Vanderdonck // Engineering Advanced Web Applications : Proceedings of Workshops in Connection with the 4th Intern. Conf. on Web Engineering (ICWE 2004), Munich, Germany, 28–30 July 2004. – Munich, 2004. – P. 325–338.
8. *Van den Bergh, J. CAP3: Context-sensitive abstract user interface specification* / J. Van den Bergh, K. Luyten, K. Coninx // Proceedings of the 3rd ACM SIGCHI Symp. on Engineering Interactive Computing Systems EICS '11, Pisa, Italy, 2011. – Pisa, 2011. – P. 31–40. <https://dl.acm.org/doi/10.1145/1996461.1996491>
9. *Paterno, F. Maria: A universal, declarative, multiple abstraction-level language for service-oriented applications in ubiquitous environment* / F. Paterno, C. Santoro, L. D. Spano // ACM Transactions on Computer-Human Interaction. – 2009. – Vol. 16, iss. 4. – P. 1–30. <https://doi.org/10.1145/1614390.1614394>
10. *Declarative interface models for user interface construction tools: the MASTERMIND approach* / P. Szekely [et al.] ; eds.: L. J. Bass, C. Unger // Engineering for Human-Computer Interaction. EHCI 1995. IFIP Advances in Information and Communication Technology. – Springer, Boston, MA, 1996. – P. 120–150. https://doi.org/10.1007/978-0-387-34907-7_8
11. *Liu, B. Deriving user interface from ontologies: A model-based approach* / B. Liu, H. Chen, W. He // 17th IEEE Intern. Conf. on Tools with Artificial Intelligence (ICTAI'05), Hong Kong, 14–16 Nov. 2005. – Hong Kong, 2005. – P. 254–259. <https://doi.org/10.1109/ICTAI.2005.55>
12. *Gaulke, W. Using profiled ontologies to leverage model driven user interface generation* / W. Gaulke, J. Ziegler // The 7th ACM SIGCHI Symp. on Engineering Interactive Computing Systems, Duisburg, 23–26 June 2015. – Duisburg, 2015. – P. 254–259. <https://doi.org/10.1145/2774225.2775070>

13. ActiveRaUL: Automatically generated Web interfaces for creating RDF data / A. Sahar [et al.] // Proceedings of the 12th Intern. Semantic Web Conf. (Posters & Demos Track), Sydney, 21–25 Oct. 2013. – Sydney, 2013. – P. 117–120.
14. Hitz, M. Using application ontologies for the automatic generation of user interfaces for dialog-based application / M. Hitz, T. Kessel // Research and Practical Issues of Enterprise Information Systems. – 2016. – Vol. 268. – P. 16–31. https://doi.org/10.1007/978-3-319-49944-4_2
15. Грибова, В. В. Автоматизация разработки пользовательских интерфейсов с динамическими данными / В. В. Грибова, Н. Н. Черкезишвили // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2011) : материалы Междунар. науч.-техн. конф., Минск, 10–12 февр. 2011 г. / редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2011. – С. 287–292.
16. Грибова, В. В. Генератор кода пользовательского интерфейса, управляемый онтологией / В. В. Грибова, А. В. Тарасов // Искусственный интеллект. – 2005. – Т. 4. – С. 457–464.
17. Hepp, M. Ontologies: State of the art, business potential, and grand challenges / M. Hepp // Ontology Management, Part of the Computing for Human Experience Book Series (ADSW, vol. 7). – Boston : Springer, 2008. – P. 3–22. https://doi.org/10.1007/978-0-387-69900-4_1
18. Beyond data models for automated user interface generation / A. R. Puetra [et al.] // People and Computers IX, Proceedings of HCI '94. – Glasgow, Aug. 1994. – P. 353–366.
19. Sadowski, M. The structure of next-generation intelligent computer system interfaces / M. Sadowski // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2022) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2022. – Вып. 6. – С. 199–208.
20. Голенков, В. В. Открытая технология онтологического проектирования, производства и эксплуатации семантически совместимых гибридных интеллектуальных компьютерных систем / В. В. Голенков, Н. А. Гулякина, Д. В. Шункевич. – Минск : Бестпринт, 2021. – 690 с.
21. Голенков, В. В. Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Ч. 1: Принципы создания / В. В. Голенков, Н. А. Гулякина // Онтология проектирования. – 2014. – № 1(11). – С. 42–64.
22. From training intelligent systems to training their development tools / V. V. Golenkov [et al.] // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2018) : материалы Междунар. науч.-техн. конф., Минск, 15–17 февр. 2018 г. / редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2018. – С. 81–98.
23. Davydenko, I. Semantic models, method and tools of knowledge bases coordinated development based on reusable components / I. Davydenko // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2018) : материалы Междунар. науч.-техн. конф., Минск, 15–17 февр. 2018 г. / редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2018. – С. 99–118.
24. Shunkevich, D. V. Agent-oriented models, method and tools of compatible problem solvers development for intelligent systems / D. V. Shunkevich // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2018) : материалы Междунар. науч.-техн. конф., Минск, 15–17 февр. 2018 г. / редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2018. – С. 119–132.
25. Principles of organization and automation of the semantic computer systems development / V. Golenkov [et al.] // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2019) : материалы Междунар. науч.-техн. конф., Минск, 21–23 февр. 2019 г. / редкол.: В. В. Голенков (гл. ред.) [и др.]. – Минск : БГУИР, 2019. – С. 53–90.
26. Sadowski, M. Semantic-based design of an adaptive user interface / M. Sadowski // Open Semantic Technologies for Intelligent Systems OSTIS-2021, Communications in Computer and Information Science. – Springer, Cham, 2022. – Vol. 1625. – P. 165–191. https://doi.org/10.1007/978-3-031-15882-7_10
27. Zagorskiy, A. Principles for implementing the ecosystem of next-generation intelligent computer systems / A. Zagorskiy // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2022) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2022. – Вып. 6. – С. 347–356.
28. Sadowski, M. User interface of the OSTIS Ecosystem / M. Sadowski // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2023) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2023. – Вып. 7. – С. 153–158.

29. Orlov, M. Control tools for reusable components of intelligent computer systems of a new generation / M. Orlov // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2023) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2023. – Вып. 7. – С. 191–206.

References

1. Sadowski M. E. *General principles of organization of interfaces of ostis-systems*. Tehnologija kompleksnoj podderzhki zhiznennogo cikla semanticheski sovместimyh intellektual'nyh komp'yuternyh sistem novogo pokolenija [*Comprehensive Support Technology for the Life Cycle of Semantically Compatible Intelligent Computer Systems of a New Generation*]. In V. V. Golenkov. Minsk, 2023, chap. 4.1, pp. 417–432 (In Russ.).
2. Fomina T., Novikova G. *Designing an adaptive interface for an information system to support the activities of an educational institution*. Vestnik Altajskoj akademii jekonomiki i prava [*Bulletin of Altai Academy of Economics and Law*], 2020, vol. 6, no. 1, pp. 125–133. <https://doi.org/10.17513/vaael.1174> (In Russ.).
3. Boriskin A., Koronchik D., Zhukau I., Sadowski M., Khusainov A. *Ontology-based design of intelligent systems user interface*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh sistem : materialy Mezhdunarodnoj nauchno-tehnicheskoy konferencii, Minsk, 16–18 fevralja 2017 g. [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2017) : Proceedings of the International Scientific and Technical Conference, Minsk, 16–18 February 2017*]. Ed. board: V. V. Golenkov, L. S. Globa, V. A. Golovko, N. A. Guljakina, O. P. Kuznecov, ..., V. F. Horoshevskij. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2017, pp. 95–106.
4. Belousova S., Rogozov Y. *Analysis of approaches to user interface creation*. Izvestija Juzhnogo federal'nogo universiteta. Tehnicheskie nauki [*Proceedings of the Southern Federal University. Technical Science*], 2014, vol. 6(155), pp. 142–148 (In Russ.).
5. Hussain J., Hassan A., Hafiz S., Rahman A., Muhammad A., ..., Sungyoung L. Model-based adaptive user interface based on context and user experience evaluation. *Journal on Multimodal User Interfaces*, 2018, vol. 12, pp. 1–16. <https://doi.org/10.1007/s12193-018-0258-2>
6. Abrams M., Phanouriou C., Batongbacal A., Williams S., Shuster J. UIML: An appliance-independent XML user interface language. *Proceedings of the 8th International Conference on the World Wide Web, Toronto, 11–14 May 1999*. Toronto, 1999, pp. 1695–1708.
7. Limbourg Q., Vanderdonck J. USIXML: A user interface description language supporting multiple levels of independence. *Proceedings of Workshops in Connection with the 4th International Conference on Web Engineering (ICWE 2004), Munich, Germany, 28–30 July 2004*. Munich, 2004, pp. 325–338.
8. Van den Bergh J., Luyten K., Coninx K. CAP3: Context-sensitive abstract user interface specification. *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems EICS, Pisa, Italy, 2011*. Pisa, 2011, pp. 31–40. <https://dl.acm.org/doi/10.1145/1996461.1996491>
9. Paterno F., Santoro C., Spano L. D. Maria: A universal, declarative, multiple abstraction-level language for service-oriented applications in ubiquitous environment. *ACM Transactions on Computer-Human Interaction*, 2009, vol. 16, iss. 4, pp. 1–30. <https://doi.org/10.1145/1614390.1614394>
10. Szekely P., Sukaviriya P., Castells P., Muthukumarasamy J., Salcher E. Declarative interface models for user interface construction tools: the MASTERMIND approach. In L. J. Bass, C. Unger (eds.). *Engineering for Human-Computer Interaction. EHCI 1995. IFIP Advances in Information and Communication Technology*. Springer, Boston, MA, 1996, pp. 120–150. https://doi.org/10.1007/978-0-387-34907-7_8
11. Liu B., Chen H., He W. Deriving user interface from ontologies: A model-based approach. *17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05), Hong Kong, 14–16 November 2005*. Hong Kong, 2005, pp. 254–259. <https://doi.org/10.1109/ICTAI.2005.55>
12. Gaulke W., Ziegler J. Using profiled ontologies to leverage model driven user interface generation. *The 7th ACM SIGCHI Symposium on Engineering Interactive Computing Systems, Duisburg, 23–26 June 2015*. Duisburg, 2015, pp. 254–259. <https://doi.org/10.1145/2774225.2775070>
13. Sahar A., Armin B., Shepherd H., Lexing L. ActiveRaUL: Automatically generated Web interfaces for creating RDF data. *Proceedings of the 12th International Semantic Web Conference (Posters & Demos Track), Sydney, 21–25 October 2013*. Sydney, 2013, pp. 117–120.
14. Hitz M., Kessel T. Using application ontologies for the automatic generation of user interfaces for dialog-based application. *Research and Practical Issues of Enterprise Information Systems*, 2016, vol. 268, pp. 16–31. https://doi.org/10.1007/978-3-319-49944-4_2
15. Gribova V. V., Cherkezishvili N. N. *Automation of development of user interfaces with dynamic data*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh sistem : materialy Mezhdunarodnoj

nauchno-tehnicheskoy konferencii, Minsk, 10–12 fevralja 2011 g. [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2011) : Proceedings of the International Scientific and Technical Conference, Minsk, 10–12 February 2011*]. Ed. board: V. V. Golenkov, T. A. Gavrilova, V. A. Golovko, V. V. Gribova, N. A. Guljakina, ..., S. S. Shherbak. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2011, pp. 287–292 (In Russ.).

16. Gribova V. V., Tarasov A. V. *Ontology-driven user interface code generator*. *Iskusstvennyj intellekt [Artificial Intelligence]*, 2005, vol. 4, pp. 457–464 (In Russ.).

17. Hepp M. Ontologies: State of the art, business potential, and grand challenges. *Ontology Management, Part of the Computing for Human Experience Book Series (ADSW, vol. 7)*. Boston, Springer, 2008, pp. 3–22. https://doi.org/10.1007/978-0-387-69900-4_1

18. Puetra A. R., Eriksson H., Gennari J. H., Musen M. A. Beyond data models for automated user interface generation. *People and Computers IX, Proceedings of HCI '94*. Glasgow, August 1994, pp. 353–366.

19. Sadowski M. *The structure of next-generation intelligent computer system interfaces*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : sbornik nauchnyh trudov [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2022) : Research Papers Collection*]. Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2022, iss. 6, pp. 199–208.

20. Golenkov V. V., Guliakina N. A., Shunkevich D. V. *Open Technology of Ontological Design, Production, and Operation of Semantically Compatible Hybrid Intelligent Computer Systems*. Minsk, Bestprint, 2021, 690 p.

21. Golenkov V. V., Guliakina N. A. *An open semantic technology project for component-based design of intelligent systems. Part 1: Principles of creation*. Ontologija proektirovanija [*Design Ontology*], 2014, no. 1(11), pp. 42–64.

22. Golenkov V. V., Guliakina N. A., Grakova N. V., Davydenko I. T., Nikulenk V. Y., ..., Tarasov V. B. *From training intelligent systems to training their development tools*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : materialy Mezhdunarodnoj nauchno-tehnicheskoy konferencii, Minsk, 15–17 fevralja 2018 g. [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2018) : Proceedings of the International Scientific and Technical Conference, Minsk, 15–17 February 2018*]. Ed. board: V. V. Golenkov, T. A. Gavrilova, V. A. Golovko, P. S. Grabust, N. A. Guljakina, ..., A. A. Sharipbaj. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2018, pp. 81–98.

23. Davydenko I. *Semantic models, method and tools of knowledge bases coordinated development based on reusable components*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : materialy Mezhdunarodnoj nauchno-tehnicheskoy konferencii, Minsk, 15–17 fevralja 2018 g. [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2018) : Proceedings of the International Scientific and Technical Conference, Minsk, 15–17 February 2018*]. Ed. board: V. V. Golenkov, T. A. Gavrilova, V. A. Golovko, P. S. Grabust, N. A. Guljakina, ..., A. A. Sharipbaj. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2018, pp. 99–118.

24. Shunkevich D. V. *Agent-oriented models, method and tools of compatible problem solvers development for intelligent systems*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : materialy Mezhdunarodnoj nauchno-tehnicheskoy konferencii, Minsk, 15–17 fevralja 2018 g. [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2018) : Proceedings of the International Scientific and Technical Conference, Minsk, 15–17 February 2018*]. Ed. board: V. V. Golenkov, T. A. Gavrilova, V. A. Golovko, P. S. Grabust, N. A. Guljakina, ..., A. A. Sharipbaj. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2018, pp. 119–132.

25. Golenkov V. V., Shunkevich D., Davydenko I. T., Grakova N. V. *Principles of organization and automation of the semantic computer systems development*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : materialy Mezhdunarodnoj nauchno-tehnicheskoy konferencii, Minsk, 21–23 fevralja 2019 g. [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2019) : Proceedings of the International Scientific and Technical Conference, Minsk, 21–23 February 2019*]. Ed. board: V. V. Golenkov, Gavrilova T. A., V. A. Golovko, P. S. Grabust, N. A. Guliakina, ..., A. A. Sharipbay. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2019, pp. 53–90.

26. Sadowski M. Semantic-based design of an adaptive user interface. *Open Semantic Technologies for Intelligent Systems OSTIS-2021, Communications in Computer and Information Science*. Springer, Cham, 2022, vol. 1625, pp. 165–191. https://doi.org/10.1007/978-3-031-15882-7_10

27. Zagorskiy A. *Principles for implementing the ecosystem of next-generation intelligent computer systems*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : sbornik nauchnyh trudov [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2022) : Research Papers Collection*].

Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2022, iss. 6, pp. 347–356.

28. Sadowski M. *User interface of the OSTIS Ecosystem*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : sbornik nauchnyh trudov [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2023) : Research Papers Collection*]. Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2023, iss. 7, pp. 153–158.

29. Orlov M. *Control tools for reusable components of intelligent computer systems of a new generation*. Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh system : sbornik nauchnyh trudov [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2023) : Research Papers Collection*]. Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2023, iss. 7, pp. 191–206.

Информация об авторе

Садовский Михаил Ефимович, исследователь технических наук, кафедра интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: sadovski@bsuir.by
<https://orcid.org/0000-0003-0699-8388>

Information about the author

Mikhail E. Sadowski, Researcher of Technical Sciences, the Department of Intelligent Information Technologies, Belarusian State University of Informatics and Radioelectronics.
E-mail: sadovski@bsuir.by
<https://orcid.org/0000-0003-0699-8388>