

М.В. Евланов, М.А. Керносов, М.Э. Керносова

Информационная технология ускоренной разработки информационных систем

Предложена информационная технология, обеспечивающая эффективное повторное использование требований к соответствующим системам, представленным в виде информации, данных и знаний. Технология основана на отображении требований в элементы программного и информационного обеспечения системы.

The information technology that provides effective reuse requirements for the corresponding systems, presented in the form of information, data and knowledge, is proposed. The technology is based on the requirements mapping to the software and database elements of the system.

Запропоновано інформаційну технологію, що забезпечує ефективне повторне використання вимог до відповідних систем, поданих у вигляді інформації, даних та знань. Технологію засновано на відображенні вимог в елементи програмного та інформаційного забезпечення системи.

Введение. В настоящее время практически любая информационная система (ИС) рассматривается как совокупность ИТ-услуг, реализованных организацией–Поставщиком и необходимых организации–Потребителю для автоматизации своей основной деятельности. Под термином *ИТ-услуга* будем понимать самостоятельную функциональную задачу ИС, использование которой для выполнения отдельной работы процесса предприятия/организации Потребителя ИТ-услуг или для управления этой работой экономически и технически целесообразно. Термин «Потребитель ИТ-услуг» – организации, нуждающиеся в этих услугах и использующие их в своих процессах для достижения целей, поставленных перед ними. Под термином «Поставщик ИТ-услуг» будем понимать организации, считающие основной целью своей деятельности эффективное и качественное выполнение работ по созданию, внедрению, эксплуатации и модернизации ИС, отдельных ИТ-услуг и реализующих эти услуги ИТ-сервисов [1].

Исходя из данного представления ИС, разрабатываемая конкретным Поставщиком ИТ-услуг для конкретного Потребителя ИТ-услуг, должна обладать следующими качествами:

- удовлетворять выдвинутым требованиям Потребителя ИТ-услуг, иметь низкую стоимость и как можно более короткие сроки разработ-

ки, обеспечивать максимальную функциональность и надежность;

- представлять собой продукт Поставщика ИТ-услуг, разработка, внедрение, сопровождение и модернизация которого осуществляется с минимальными временными, финансовыми и трудовыми затратами.

Помимо отмеченных особенностей, любая ИС может быть охарактеризована потенциальным наличием множества невыявленных или несформулированных требований Потребителя ИТ-услуг, множества априорно выдвинутых предположений и утверждений Поставщика ИТ-услуг, а также множеством рисков изменения требований к ИС в ходе ее разработки, внедрения, эксплуатации и модернизации.

Таким образом, проблему своевременного выявления требований к ИС на любой стадии ее жизненного цикла и внесения изменений в ее обеспечивающую часть, реализующих выявленные требования, следует считать актуальной как с учетом Потребителя ИТ-услуг, так и Поставщика ИТ-услуг. Не менее актуальна проблема повторного использования Поставщиком требований, сформулированных к ранее реализованным ИС, в проекте создания новой ИС или адаптации типовой ИС к особенностям конкретного Потребителя ИТ-услуг. Поэтому разработка новых и совершенствование существующих моделей, методов и инструментальных средств автоматиза-

ции работы с требованиями в ходе создания новой или модернизации существующей ИС – весьма актуальное и перспективное направление исследований в области компьютерных наук.

Анализ существующих подходов к описанию требований, выдвигаемых к информационной системе

Выделение единого определения таких понятий, как «требование» и «требование к ИС» связано с серьезными трудностями. Ряд специалистов придерживается определения понятия «требование», изложенного в стандарте *IEEE 610.12–1990*, но при возникновении практической необходимости меняют эти определения так, чтобы они лучше отражали особенности конкретных частных методов и методик формирования, анализа и управления требованиями [2–4]. Аналогичным образом осуществляется разделение требований на группы по неким общим признакам. К настоящему времени классификация подобных признаков не получила общепринятого научного обоснования.

Схожая ситуация наблюдается и для попыток описания требований к ИС с помощью различных моделей. Исследование атрибутивных описаний требований к ИС, требований к программному обеспечению (ПО) как более широкого класса требований, а также изучение документов, используемых для описания и спецификации подобных требований, позволяет утверждать следующее [2–4]:

- единой атрибутивной модели описания требования не существует;
- атрибутивные модели описания требований могут уточняться, изменяться и дополняться новыми атрибутами в зависимости от особенностей разрабатываемой системы, предметной области (ПрО), Поставщика, Потребителя и других факторов;
- описания бизнес-требований, потребностей пользователей и аналогичных по назначению групп требований носят практически неформализованный характер и очень слабо используют атрибутивные модели;
- наиболее формальными есть описания функциональных требований к ПО, которые основаны на атрибутивных моделях и совме-

стным использовании формального аппарата той или иной разновидности визуальных моделей.

Особо следует отметить практически полное отсутствие освещения вопросов применимости излагаемых в научной литературе общих методов и методик формирования, анализа и управления требованиями. Многочисленные неудачи повторного использования требований к ИС в новых проектах во многом являются следствиями невозможности использования «в чистом виде» преимуществ конкретной методики или инструментального средства. В результате возникает своеобразный симбиоз практик, методов и методик формирования, анализа и управления требованиями как попытка формирования некой компромиссной методологии для разработки конкретной ИС [5]. Однако реализация такой компромиссной методологии опирается на инструментальные средства и программные продукты, созданные в рамках какой-либо одной из участвовавших в симбиозе методологий. Такое расхождение во многом ограничивает применение средств управления требованиями и делает затруднительным повторное использование наполненных репозитариев требований в других проектах.

Выделение нерешенных частей проблемы и постановка задачи исследования

Повторное использование компонентов, полученных в результате выполнения предыдущих проектов, один из наиболее распространенных способов сокращения трудозатрат на разработку ИС как разновидности программных продуктов. Однако основной недостаток, затрудняющий использование данного способа, – сложность анализа портфеля ранее выполненных проектов создания ИС на предмет возможности применения их компонентов при разработке новой системы [1].

Задачи определения оптимального объема повторно используемых компонентов при разработке новой ИС и обеспечения максимального соответствия ИС специфике конкретного уникального бизнес-процесса (БП) Потребителя ИТ-услуг во многом противоречивы. Для их решения необходимо проведение анализа ПрО

и требований к создаваемой ИС, а также сопоставление результатов анализа с терминами аналогичных ПрО и требованиями к разработанному ранее ИС. Это позволит определить возможность полного или частичного применения ранее сформулированных требований и соответствующих готовых проектных решений в новой ИС. Решение этих задач требует хорошего знания аналитиками как ПрО, для которой разрабатывается ИС, так и ПрО, для которых были выполнены предыдущие проекты. При этом значительно возрастает риск ошибок, допускаемых в ходе предпроектного обследования процессов Потребителя ИТ-услуг, формирования и анализа требований к ИС, а также сложность и критичность качественной реализации этих работ.

Поэтому и с теоретической, и с практической стороны необходимо решение задачи разработки информационной технологии (ИТ), позволяющей осуществлять автоматизированный анализ множеств ПрО и ранее выполненных проектов ИС с целью поддержки принятия решений о целесообразности повторного применения полученных ранее решений в ходе создания новой ИС. Разработка и применение подобной ИТ позволит повысить эффективность повторного использования компонентов ИС путем автоматизации процедур сравнительного анализа терминов различных ПрО и снижения количества ошибок, допускаемых в ходе проведения подобного анализа.

Основные положения информационной технологии ускоренной разработки информационных систем

В ходе решения поставленной задачи исследования был определен набор положений, на основе которого следует осуществлять разработку подобной ИТ. Первым положением есть представление главной цели деятельности ИС как формирования и отображения единого целостного информационного представления объекта автоматизации или бизнес-процесса в соответствии с поставленными перед ними целями. Это позволяет рассматривать создание ИС как процесс формирования единого целостного описания ПрО и последующего его преобразова-

ния в описания функций и видов обеспечений создаваемой ИС. Для обеспечения возможностей формирования такого описания ПрО предлагается использовать положение о сервисном подходе к созданию ИС как совокупности ИТ-услуг, состав и содержание которых определяется особенностями ПрО и автоматизируемых БП.

Для обеспечения возможности преобразования описания ПрО в описания функций и видов обеспечений ИС предлагается использовать концепцию представления требований к ИС как совокупности следующих положений [6]:

- представление требований к ИС как элементов универсума требований и методов их формирования, известных и неизвестных разработчику, заказчику или им обоим;
- изначальное многообразие представлений требований к ИС в виде данных, информации и знаний;
- процессный подход к описанию требований;
- подход к управлению требованиями, основанный на постепенном преобразовании множества начальных значений атрибутов, описывающих требование, во множество желаемых их значений.

Данные положения позволяют отказаться от оценки возможности повторного использования ранее сформулированных требований к ИС и перейти к повторному использованию знаний о структурах данных и процессах их обработки, позволяющих оценить достижимость главной цели деятельности ИС. Такие знания представляют собой паттерны проектирования требований к ИС [7] и обуславливают введение дополнительного положения, которое заключается в существовании взаимно однозначного отображения паттернов проектирования требований в элементы программного (ПО) и информационного (ИО) обеспечений ИС.

Для описания паттернов проектирования требований наиболее предпочтителен – аппарат теории фреймов. Применение данного аппарата для описания знаний о ПрО, формального представления требований к ИС и описания знаний о разработанных элементах ПО ИС позволяет обеспечить взаимно однозначное отображение требований в элементы ПО. Взаимно

однозначное отображение требований к ПрО в элементы ИО ИС основано на идее объектно-реляционного отображения элементов ИО и ПО ИС друг в друга. Тогда множество описаний терминов ПрО (онтологию ПрО), характеризующих их структур данных и правил их обработки, а также совокупность элементов ПО (иерархии классов и пакетов) и ИО (таблицы, представления, процедуры, триггеры и т.д.) обеспечения, реализующих данное требование, следует называть реализованным паттерном проектирования требования к ИС.

Этапы технологии ускоренной разработки информационных систем

На основе рассмотренных положений была разработана ИТ ускоренной разработки ИС, которая в общем случае включает в себя следующие этапы:

- 1) выявление требований к ИС;
- 2) формирование иерархий терминов ПрО для создаваемой ИС;
- 3) сравнение сформированных иерархий терминов ПрО создаваемой ИС с разработанными ранее иерархиями терминов ПрО;
- 4) выделение и уточнение паттернов проектирования требований к ИС;
- 5) синтез архитектуры ИС на основе выделенных паттернов проектирования требований к ИС;
- 6) осуществление взаимно однозначного отображения паттернов проектирования ИС в структуры базы данных и классы программного обеспечения разрабатываемой ИС.

Первый этап ИТ предполагает сбор требований к проектируемой ИС (независимо от способа их представления), а также их последующий анализ, обработку и формализацию. Конечная цель осуществления данного этапа – сбор достаточного количества информации и данных, необходимых для создания онтологии ПрО создаваемой ИС.

Второй этап ИТ предполагает непосредственное создание на основе различных описаний требований к ИС онтологий терминов данной ПрО (представление описаний ПрО в виде знаний). Следует отметить, что онтологии ПрО для создаваемой ИС и онтологии элементов реализованных ИС имеют принципиальные от-

личия. В онтологии ПрО создаваемой ИС отражаются термины, характерные для конкретного объекта автоматизации и его БП. Эти онтологии могут быть избыточными, допускать синонимию и неоднозначность описаний отдельных терминов и структур и не предполагают обязательное отражение в элементы ИО и ПО ИС в неизменном виде. Онтологии элементов реализованной ИС – системцентричные описания реализованных паттернов проектирования требований, не допускающие избыточности и неоднозначности, поскольку любое их изменение требует модификации соответствующих элементов разработанных решений ИО и ПО ИС.

Конечной целью осуществления второго этапа ИТ есть создание или обновление двух следующих универсумов:

- универсум понятий, используемых в создаваемой ИС, представляемый как онтология ПрО создаваемой ИС;
- универсум реализованных требований, описывающий библиотеку реализованных элементов ИС и являющийся универсумом понятий, терминов, структур данных и методов, доступных разработчику в ходе создания новой ИС на базе существующих систем.

Над этими универсумами возможно выполнение операций расширения (введения в онтологию новых терминов) и сужения (исключения неиспользуемых терминов). Критерий добавления (или удаления) в онтологию иерархии терминов ПрО – наличие (отсутствие) горизонтальных понятийных связей между анализируемым термином и остальными элементами онтологии.

Третий этап ИТ предполагает анализ и сопоставление иерархий терминов ПрО создаваемой ИС с иерархиями терминов универсума реализованных требований с целью формирования базовых онтологий ИС, позволяющих найти ответ на следующие вопросы:

- какие готовые элементы ИС и в каком объеме могут быть применены;
- в каком объеме необходима разработка новых или расширение (дополнение) существующих элементов ИС.

Выполнение данного этапа позволяет также сформулировать и включить в план проекта создания ИС основные проектные работы, осуществить последующую структурную декомпозицию работ, сформировать графики их выполнения, назначить исполнителей, т.е. осуществить переход от стадии предпроектного анализа к стадии планирования проекта ИС.

Четвертый этап ИТ предполагает выполнение предварительных работ, необходимых для синтеза варианта конфигурации создаваемой ИС. Выделенные на третьем этапе базовые онтологии элементов создаваемой ИС могут потребовать доработки. Так, в случае использования для описания ПрО создаваемой ИС онтологий из универсума реализованных требований, понятия ПрО формируются на основе этих онтологий путем расширения базовых понятий новыми атрибутами. Отметим, что полностью формализовать принятие решений о применении тех или иных паттернов проектирования требований в качестве базовых невозможно. Окончательно решение о применении паттерна, как и при анализе синонимов терминов ПрО, принимает эксперт. Поэтому конечная цель данного этапа – уточнение паттернов проектирования требований, применяемых при разработке новой ИС.

На *пятом этапе* ИТ осуществляется синтез слабосвязной сервис-ориентированной архитектуры с применением существующих, доработанных или новых описаний паттернов проектирования требований к ИС. Данный синтез предполагает использование методов искусственного интеллекта, позволяющих решать задачи оптимизации объема повторно используемых паттернов проектирования требований к ИС и максимизации соответствия адаптируемых паттернов проектирования требований к ИС специфике конкретных БП Потребителя ИТ-услуг создаваемой ИС.

На *шестом этапе* ИТ осуществляется взаимно однозначное отображение паттернов проектирования требований к ИС в сущности ИО и классы ПО создаваемой ИС для уточняемых или новых паттернов проектирования требований к создаваемой ИС, а также организация взаимодействия полученных таким образом эле-

ментов ИО и ПО в синтезированном варианте конфигурации создаваемой ИС.

Для реализации рассмотренных этапов ИТ необходимо обеспечить взаимодействие представлений требований к ИС на уровнях данных, информации и знаний в соответствии со схемой, приведенной на рисунке.

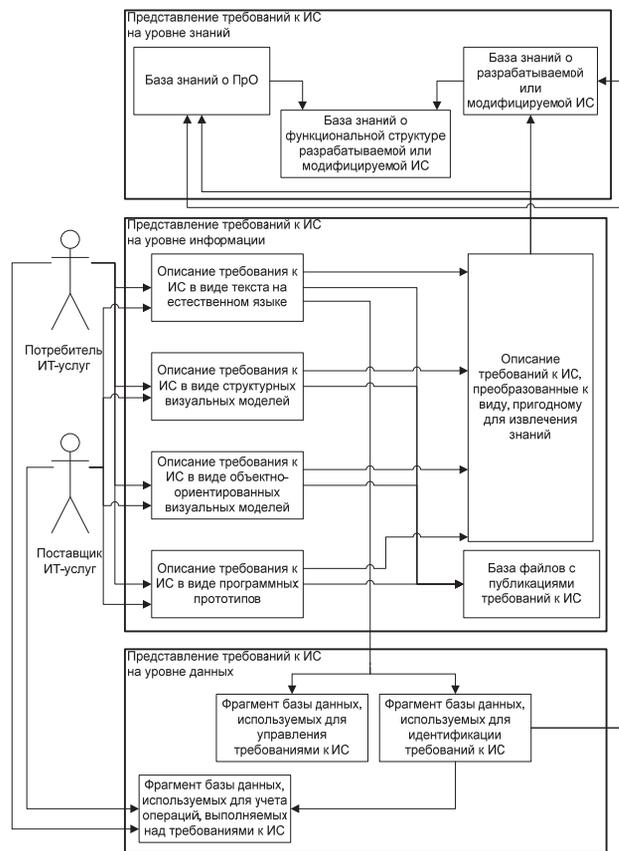


Схема взаимодействия представлений требований к ИС на уровнях информации, данных и знаний в рамках ИТ ускоренной разработки информационных систем

Предлагаемая ИТ позволяет сократить затраты времени на выполнение процессов создания ИС принятием решения о целесообразности повторного использования содержимого библиотеки реализованных элементов ИС, имеющейся в распоряжении Поставщика ИТ-услуг, уже в ходе предпроектного обследования БП Потребителя ИС. Поэтому данную ИТ предлагается называть ИТ ускоренной разработки ИС.

Особенности отображения онтологий в элементы информационного и программного обеспечения системы

Как показано выше, шестой этап ИТ ускоренной разработки ИС предполагает осуществле-

ние взаимно однозначного отображения сети фреймов, описывающей онтологию ПрО создаваемой ИС в виде паттернов проектирования требований к ИС, в элементы ПО и ИО ИС. Такое отображение сводится к решению задачи объектно-реляционного отображения. В рамках ИТ ускоренной разработки ИС для реализации данного отображения предложена эмуляция механизма наследования с использованием идентифицирующих связей между сильными (родительскими) и слабыми (дочерними) сущностями. В этом случае база данных создаваемой ИС формируется в результате отображения онтологии ПрО как множества новых и уточненных паттернов проектирования требования к ИС как набор взаимосвязанных витрин данных. В схемах каждой из этих витрин таблицей фактов есть корневой элемент конкретной иерархии фреймов, а консольным таблицам и таблицам измерений соответствуют уточняющие понятия ПрО. При этом наследуемые фреймы представляются как слабые сущности, зависящие от корневого понятия соответствующей иерархии фреймов онтологии ПрО.

Тогда названия таблиц соответствуют наименованиям фреймов, атрибуты таблиц – атрибутам фреймов, а присоединенные процедуры фреймов отображаются в хранимые процедуры, функции и триггеры реляционной схемы данных. Иерархические связи отображаются в наборы сущностей, связанных между собой идентифицирующими связями. Связи–ассоциации отображаются в ссылочные ограничения целостности данных, реализуемые с применением первичных и внешних ключей.

Для реализации наследования в реляционной базе данных предлагается выделение абстрактной обобщенной сильной сущности, соответствующей корневому элементу конкретной иерархии фреймов и являющейся универсальной для различных ПрО. Атрибуты такой сущности носят общий характер. Применение их предполагается при реализации требований, характерных для нескольких ИС. Фреймы второго и более высоких уровней иерархии отображаются в слабые сущности, полностью зависящие от более общей сущности предыду-

щего уровня иерархии и дополняют ее, поскольку содержат дополнительную информацию, характерную для более конкретных классов объектов ПрО ИС. Такая слабая сущность в качестве ключевого атрибута (*PK*) содержит внешний ключ (*FK*), реализующий связь «1:0...1» с обобщающим сильным типом сущности. При этом если для соответствующего сущности фрейма выделены фреймы–потомки, такая сущность, в свою очередь, есть сильной для сущностей, соответствующим им.

Распределение набора атрибутов из описания одного термина ПрО между несколькими таблицами, связанными идентифицирующими связями и отношениями «1:0...1» и созданными с целью реализации в реляционной базе данных иерархических связей, требует организации обеспечения целостности данных на уровне всех этих таблиц. Поскольку ни целостность сущностей, ни ссылочная целостность в полной мере не обеспечивает решение данной задачи, в рамках ИТ ускоренной разработки ИС предлагается использовать триггеры. Триггеры реализуют сложные ограничения целостности данных, которые нельзя реализовать описательными ограничениями при создании таблиц. Следует выделить четыре основных типа триггеров, обеспечивающих целостность данных в обобщенной и уточняющих таблицах:

- триггеры, обеспечивающие сюръективное отображение данных в двух таблицах (обязательное наличие соответствующих записей в главной и зависимой таблицах, связь «1:1», поскольку наличие идентифицирующей связи обеспечивает только наличие связи «1:0...1»);
- триггеры, обеспечивающие уникальность значений комбинаций атрибутов, расположенных в разных таблицах (аналог уникального ключа в одной таблице);
- триггеры, обеспечивающие контроль внесения данных в зависимые таблицы, когда сильная сущность одновременно связана с несколькими слабыми, и данные в зависимых таблицах не должны пересекаться между собой (например, сильная сущность «человек», слабые – «преподаватель» и «студент»);

• триггеры, обеспечивающие заполнение обязательных полей сильных сущностей (аналог *not null option*), если это требуется для зависящих от них слабых сущностей (например, для абстрактной сильной сущности «человек» поле «контактный телефон» может быть необязательным, а для наследуемой от нее сущности «сотрудник» это же поле, физически расположенное в главной таблице, обязательно).

Тогда задачу взаимно однозначного отображения онтологий ПрО в описания элементов ИО и ПО создаваемой ИС можно рассматривать как задачу последовательного построения параллельно существующих описаний создаваемой ИС. Основные этапы последовательности построения таких описаний приведены в таблице.

Применение рассмотренной ИТ значительно повышает требования к ведению проектной документации на создаваемую ИС. Целесообразна разработка правил именования классов ПО, таблиц ИО, а также их атрибутов, методов, процедур и других элементов. Применение единой системы именования элементов ПО и ИО необходимо для обеспечения двустороннего отображения диаграмм бизнес-классов ПО в элементы ИО ИС. Кроме того, такая система

позволяет осуществлять автоматическую генерацию ограничений целостности (первичных, внешних ключей и триггеров на основе заданных шаблонов) при отображении сети фреймов, описывающих требования к ИС, в элементы ИО создаваемой ИС.

Описанный подход к реализации объектно-реляционного отображения данных уменьшает производительность системы в целом множеством дополнительных сущностей и связей «1:1», которые не рекомендуется использовать в схеме базы данных. Однако его применение сводит к минимуму контролируемую избыточность данных, сохранив преимущества применения реляционной модели для организации хранения данных в ИС. Таким образом, рассмотренный подход к осуществлению отображения сети фреймов ПрО в элементы ИО и ПО ИС в рамках ИТ ускоренной разработки ИС позволяет облегчить сотрудникам Поставщика ИТ-услуг создание последующих ИС на основе ранее разработанных систем, используемых в качестве прототипов. Это, в свою очередь, очень сокращает время разработки новых ИС, а также в случае значительных изменений БП, характерных для нескольких ИС, в ходе сопровождения этих сис-

Этапы формирования сети фреймов, диаграммы классов и схемы данных ИС с применением технологии ускоренной разработки ИС

№	Описание ПрО	Описание ПО	Описание ИО
1	Выделение абстрактных понятий и их характеристик, универсальных для различных ПрО.	Выделение обобщенных классов и наборов полей, универсальных для различных ПрО.	Выделение обобщенных сущностей и наборов атрибутов, универсальных для различных ПрО.
2	Группировка выделенных понятий по признаку совместного использования.	Выделение пакетов классов.	Выделение отдельных схем (витрин) данных.
3	Формирование иерархий обобщенных понятий (выделение корневых фреймов, построение деревьев).	1. Формирование иерархий абстрактных классов. 2. Формирование параметризуемых <i>Generic</i> -классов.	1. Выделение обобщенных сильных и слабых сущностей (связи 1: 0..1). 2. Формирование представлений, объединяющих сильные и слабые физические сущности. 3. Анализ функциональных зависимостей и определение необходимых ограничений целостности данных, включая триггеры.
4	Выделение понятий и их атрибутов, характерных для конкретной ПрО.	Выделение классов и наборов полей, характерных для конкретной ПрО.	Выделение сущностей и наборов атрибутов, характерных для конкретной ПрО.
5	Анализ возможности наследования фреймов выделенных понятий от существующих.	Анализ возможности наследования классов от существующих абстрактных.	Анализ возможности формирования слабых сущностей, уточняющих обобщенные сильные.
6	Формирование иерархий понятий, характерных для конкретной ПрО.	1. Формирование иерархий классов для конкретной ПрО. 2. Параметризация классов-листьев.	1. Формирование иерархий сущностей для конкретной ПрО. 2. Формирование представлений. 3. Анализ функциональных зависимостей и задание ограничений целостности данных.
7	Расширение библиотеки паттернов новыми обобщенными понятиями.	Расширение ПО библиотеки паттернов новыми обобщенными классами.	Расширение ИО библиотеки паттернов новыми обобщенными сущностями.

тем достаточно скорректировать только обобщенную часть описания ИС, не выполняя индивидуальную доработку каждой системы.

Заключение. Применение ИТ ускоренной разработки ИС позволяет значительно сократить трудозатраты на создание новой ИС путем автоматизации решения задачи оценивания возможности повторного использования требований и соответствующих им проектных решений ИО и ПО ИС, а также задачи автоматизации формирования описаний ИО и ПО создаваемой ИС на основе сформированного описания знаний о ПрО создаваемой ИС. Данное утверждение основывается на результатах апробации рассмотренной ИТ в ходе создания двух ИС различного назначения. Результаты апробации позволили выделить следующие основные преимущества рассмотренной ИТ:

- формальная оценка возможности повторного использования элементов универсума реализованных требований к ИС в ходе создания новой ИС;

- сокращение затрат времени на создание каждой новой ИС за счет наращивания, а не реинжиниринга существующей ИС, используемой в качестве прототипа;

- аккумуляция знаний о различных ПрО, их многократное использование;

- сокращение затрат на сопровождение и модернизацию нескольких ИС, созданных на базе одних и тех же базовых онтологий ПрО.

Кроме того, данная ИТ есть самоописательной, что позволило использовать онтологии, сформированные в ходе анализа требований к ИТ, в процессе создания новых ИС. Данное свойство ИТ значительно упрощает ее модернизацию и дальнейшее развитие.

Тем не менее, рассмотренная ИТ имеет следующие недостатки, а именно:

- ✓ усложнение работ, выполняемых на стадии анализа требований к ИС;

- ✓ повышение требований к качеству ведения проектной документации;

- ✓ введение дополнительных иерархий в описаниях онтологий ПрО, диаграммах классов ПО и схемы данных ИО создаваемой ИС;

- ✓ повышение сложности моделей создаваемой ИС;

- ✓ возможное снижение производительности ИО ИС в целом путем дополнительных операций соединения при выборке данных и применения триггеров.

Отмеченные достоинства и недостатки ИТ ускоренной разработки ИС позволяют определить основные направления дальнейших исследований:

- усовершенствование моделей и методов синтеза варианта конфигурации ИС на основе знаниеориентированных описаний требований и повторно используемых элементов ИС;

- поиск новых и усовершенствование предлагаемых способов осуществления взаимно однозначного преобразования онтологий ПрО в описания ИО и ПО ИС;

- поиск и формальное доказательство ограничений целостности описания ИС как онтологии ПрО и синтезируемых на ее основе схем базы данных и диаграмм классов программных сервисов.

1. *Евланов М.В., Неумывакина О.Е., Карамышева А.Ю.* Глобальные цели Поставщика и Потребителя ИТ-услуг // Восточно-Европейский журнал передовых технологий. – 2012. – № 5/2 (59). – С. 12–17.
2. *Леффингуэлл Д., Уидриг Д.* Принципы работы с требованиями к программному обеспечению. Унифицированный подход. – М.: Вильямс, 2002. – 448 с.
3. *Виггерс К.И.* Разработка требований к программному обеспечению. – М.: Русская редакция, 2004. – 576 с.
4. *Коберн А.* Современные методы описания функциональных требований к системам. – М.: Лори, 2002. – 288 с.
5. *Волков Ю.О.* Управление требованиями и автоматизация этого процесса. – http://yurivolkov.com/articles/Requirements_management_automation_ru.html
6. *Евланов М.В.* Концепция представления требований к информационной системе // Вісн. Нац. техн. ун-ту «ХПІ». Серія «Нові рішення в сучасних технологях». – Харків: НТУ «ХПІ», 2012. – № 68(974). – С. 32–40.
7. *Левыкин В.М., Евланов М.В., Керносов М.А.* Подход к использованию паттернов проектирования при работе с требованиями к информационной системе // Системний аналіз. Інформатика. Управління (САІУ-2013): матеріали ІV Міжнар. наук.-прак. конф. (м. Запоріжжя, 13–16 березня 2013 р.). – Запоріжжя: КПУ, 2013. – С. 150–152.

Поступила 10.11.2013

Тел. для справок: +38 057 702-1451, 702-1736 (Харьков)

E-mail: evlanov_max@mail.ru, iust@list.ru,

lotfulina-marina@rambler.ru

© М.В. Евланов, М.А. Керносов, М.Э. Керносова, 2014