

УДК 355.01: 004.056

К.С. Беляев, В.А. Маклаев

СРЕДСТВА ДИНАМИЧЕСКОЙ ВИЗУАЛИЗАЦИИ В ПРОЕКТИРОВАНИИ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Беляев Константин Сергеевич, магистр техники и технологий, окончил Ульяновский государственный технический университет. Аспирант кафедры «Вычислительная техника» УлГТУ. Имеет статьи в области САПР. [e-mail: k.belyaev@ulstu.ru].

Маклаев Владимир Анатольевич, кандидат технических наук, окончил радиотехнический факультет Ульяновского политехнического института. Генеральный директор ФНПЦ ОАО «НПО «Марс». Имеет статьи в области САПР. [e-mail: mars@mv.ru].

Аннотация

В статье представлены инструментальные средства динамической визуализации (СДВ), используемые в разработке автоматизированных систем (АС) для отображения фронта работ по проектной деятельности, принимаемых решений и других полезных материалов, требующих оперативного доступа для работы с ними.

Ключевые слова: автоматизированное проектирование, динамическая визуализация, вопросно-ответная система, база опыта.

Konstantin Sergeevich Beliaev, Master of Engineering and Technologies; graduated from Ulyanovsk State Technical University, post-graduate student of the 'Computer Engineering' Chair at Ulyanovsk State Technical University; author of articles in the field of CAD. e-mail: k.belyaev@ulstu.ru.

Vladimir Anatolievich Maklaev, Candidate of Engineering, graduated from the Faculty of Radioengineering of Ulyanovsk Polytechnic Institute, Director General of Federal Research-and-Production Center Open Joint-Stock Company 'Research-and-Production Association 'Mars'; author of articles in the field of CAD e-mail: mars@mv.ru.

Abstract

The article presents dynamic visualization means used in computer-aided systems development to display the amount of work of project activity, decision-making and other useful information requiring operational access to work with them.

Key words: computer-aided design, dynamic visualization, question-and-answer system, experience base.

ВВЕДЕНИЕ

В настоящее время в большинстве современных технологий разработки сложных АС на различных этапах и в разных формах используются средства визуализации проектных работ с различных точек зрения и с разными целями. Как фон текущей операционной обстановки и по оперативным запросам визуализируются состояние работ, их потоки, отдельные задачи, проектные решения, а также результаты их обобщений и оценок. Особо важное направление визуализации связано с организационной составляющей проектирования, регистрирующей исполнителей работ, распределение задач между проектировщиками и их согласованное взаимодействие.

Существующие средства и формы визуализации в достаточной мере отражают общепризнанные технологии корпораций IBM и Microsoft. Корпорация IBM уже более 10 лет использует и распространяет технологию, в основе которой лежит мастер-методология Rational Unified Processes (RUP) [1], реализованная в инструментальной среде IBM Rational Method Composer (RMC), в том числе

в ее открытом (open-source) варианте Eclipse Process Framework.

В роли базового средства визуализации в RUP используется вебсайт, на котором визуализируется список членов рабочей группы над проектом, показано содержание работ, представлены в единообразной форме задачи и обеспечен доступ к каталогу предопределенных процессов для типовых проектных ситуаций, которые могут быть адаптированы для индивидуальных нужд. В процессах визуализации также широко используются различные «block and line», среди которых центральное место занимают диаграммы, построенные (проектировщиками) на языке Unified Modeling Language.

Визуализация, используемая корпорацией Microsoft, в своей основной части встроена в инструментальную систему Application Lifecycle Management [2], реализованную в связке инструментальных сред MS Visual Studio и MS Microsoft Team Foundation Server. Подход, используемый корпорацией, в большей мере ориентирован на разработку программных систем, в том числе программных составляющих АС.

В инструментальных средствах ALM на технологическом этапе Define широкое применение находит инструмент Visual StoryBoard. С помощью этого инструмента, в основе которого лежит известное приложение MS PowerPoint, возможно создавать богатое визуальное представление процесса управления требованиями, выдвигаемыми заказчиками проекта.

К сожалению, используемые в отмеченных передовых технологиях средства визуализации не лишены недостатков. Одним из существенных недостатков является то, что они жестко ограничены. Так, создаваемая при их помощи визуализация зачастую напоминает статичную фотографию, составленную из палитры преопределенных элементов этого средства. Другим существенным недостатком можно назвать их небогатые возможности по визуализации символической информации, представленной в виде текста или комбинации текста с графической и табличной информацией, что крайне необходимо, чтобы увидеть общую картину. Стоит отметить отсутствие возможности в этих средствах визуализации подвергать визуализации то, что выходит за рамки работы над текущим проектом. Также проектировщики не отказались бы от инструмента, отсутствующего в этих средствах, который бы помог им программировать то, что они хотят визуально наблюдать по тем материалам, которые им доступны.

В статье представлены средства динамической визуализации, учитывающие эти недостатки, реализованные в рамках инструментальной среды WIQA [3], обслуживающей концептуальное проектирование АС, то есть обеспечивающей взаимодействие проектировщиков с проектом на том этапе разработок АС, когда понимание визуализируемого проектного материала носит принципиальный характер.

1 КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ АС В ИНСТРУМЕНТАЛЬНОЙ СРЕДЕ WIQA

В среде WIQA текущее состояние процесса (концептуального) проектирования регистрируется и визуализируется с помощью дерева задач, распределенных и решаемых (а также уже решенных) между членами коллектива проектировщиков. Для каждой назначенной ему задачи проектировщик регистрирует в специальном протоколе

вопросно-ответные рассуждения (QA-рассуждения), которые ведут или привели его к решению задачи. Регистрация в протоколе (QA-протоколе) осуществляется так, что он представляет собой для каждой задачи Z_i интерактивную сборку «вопросов» и «ответов», которая способна выполнять роль модели (QA-модели) этой задачи (обозначим $QA(Z_i)$). Модели всех задач регистрируются в общей для коллектива вопросно-ответной базе (QA-базе), что обобщенно представлено на рисунке 1.

Модели решенных задач, а также модели, связанные с этими задачами, если они представляют ценность для других проектов, выполняемых в текущий момент времени или планируемых к реализации в будущем, помещаются как «активы повторного использования» [4] в Базу Опыта (группы проектировщиков или коллектива проектной организации). База Опыта может пополняться моделями активов и из других источников [5]. Более детально вопросно-ответное моделирование задач и активов отражено на рисунке 2.

Визуальное отображение QA-модели представляет собой QA-протокол, соответствующий определенной задаче дерева задач. Любой QA-протокол имеет вид дерева вопросов и ответов, использованных в ходе рассуждений в процессе решения задачи.

В своей простейшей версии, когда у моделируемой задачи отсутствуют подзадачи, логико-лингвистический вид изображен на рисунке 2. За каждым узлом схемы рисунка 1, помеченным символами Q и A с индексами, стоит интерактивный объект, представляющий соответствующий вопрос или ответ. Уникальные индексные имена этих объектов и их содержательное описание на языке L , связанные отношениями логики вопросов и ответов, образуют интерактивную сеть объектов, открытую для развития, визуальных просмотров, навигации по сети и доступа к глубинному содержанию ее узлов.

Лингвистическая составляющая вида дополняет логическую схему текстами каждой ее единицы. Динамику вида определяет то, что он создается и регистрируется как протокол, во-первых, процесса решения задачи, а во-вторых – процесса перевода рассуждений, используемых при решении задачи, на язык вопросов и ответов L^{QA} [4].

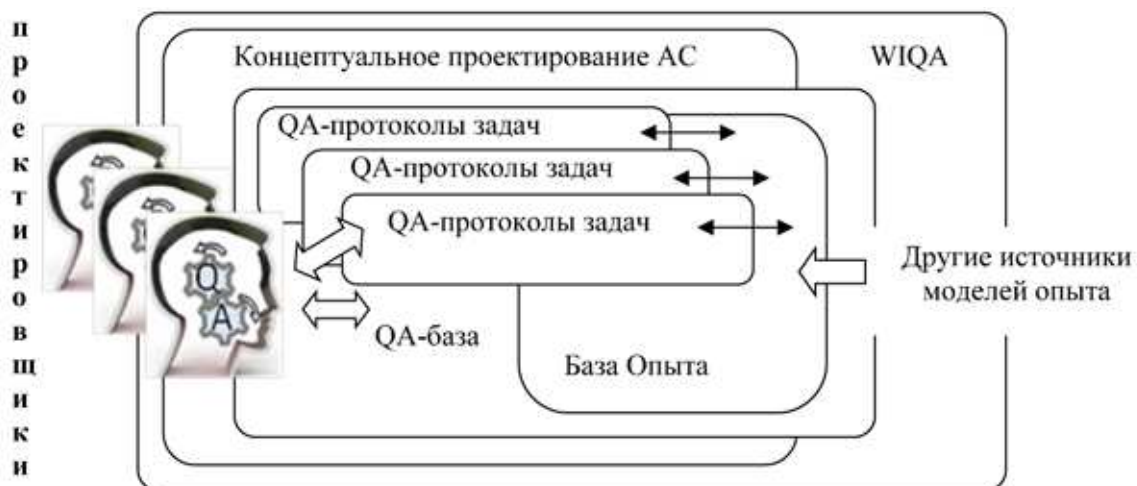


Рис. 1. Взаимодействие проектировщиков с проектом АС



Рис. 2. Нормативная визуализация задач в среде WIQA

2 ПРИМЕНЕНИЕ ТЕХНИК «АКТИВНОГО ЗРИТЕЛЬНОГО ВОСПРИЯТИЯ» К ВИЗУАЛИЗАЦИИ ЕДИНИЦ QA-ПРОТОКОЛА

Как было отмечено выше, лингвистическая составляющая вида просто отображает текст, ассоциированный с вопросами и ответами, в то время как логическая составляющая вида описывает иерархическую структуру QA-протокола. Такой способ организации представления информации эффективен лишь только с точки зрения компьютера и неудобен для извлечения ее человеком. Проектировщику приходится делать много движений глазами по дереву вопросно-ответного протокола, часто щелкать мышкой и т. п. для поиска нужной информации. Однако главным недостатком является то, что на экране одновременно может отображаться лишь одна QA-единица, таким образом проектировщик лишен возможности видеть весь путь принятия того или иного решения.

Преодолеть такую ситуацию можно, используя специальные средства визуализации вопросно-ответных единиц. Такие средства должны быть спроектированы с учетом подхода «активного зрительного восприятия» [3]. Подход «активного зрительного восприятия» учитывает многие психологические аспекты и свойства, позволяя ускорить процесс визуального поиска (получение информации с экрана монитора) и улучшить визуальное восприятие информации. К таким свойствам и аспектам можно отнести: всплывающие свойства (pop-out), движение, визуальное пространство и время, цвет.



Рис. 3. Диаграмма вариантов использования СДВ

3 СРЕДСТВА ДИНАМИЧЕСКОЙ ВИЗУАЛИЗАЦИИ ВОПРОСНО-ОТВЕТНЫХ ЕДИНИЦ ВОПРОСНО-ОТВЕТНОГО ПРОЦЕССА WIQA

Средства динамической визуализации вопросно-ответных единиц реализованы в виде плагина к вопросно-ответному процессору WIQA с применением техник «активного зрительного восприятия». Основное предназначение плагина – при помощи визуализации вопросно-ответных единиц ускорять процесс визуального поиска нужной информации и улучшать ее восприятие, тем самым позволяя сокращать время, затраченное на принятие проектных решений (рис. 3).

СДВ построены с использованием подсистемы WPF [6, 7], входящей в состав .NET Framework [8]. Особенностью этой подсистемы, предназначенной для создания клиентских приложений Windows с визуально привлекательными возможностями взаимодействия, является использование в качестве графической системы DirectX (Direct3D) [9]. Применение DirectX в WPF позволяет создавать быстро отрисовываемые трехмерные графические интерфейсы.

Основной сущностью, которой оперируют СДВ, является визуализируемая единица данных (ВЕД) – VisualizationDataItem (рис. 4). В ВЕД определены три свойства: адрес (ID), имя (Name) и визуализируемые данные (DataForVisualization). Визуализируемые данные хранятся в формате XAML [10], что делает возможным хранить для визуализации данные, представленные текстом, изображением, медиаэлементами (аудио/видео) и их композициями.

ВЕД, в свою очередь, объединяются в последовательности визуализируемых данных (ПВД) VDISequence (рис. 4). Каждая последовательность задается номером (ID) и именем (Name).

ПВД визуализируется на экране монитора согласно созданному сценарию визуализации VisualScenario (рис. 4). Параметры, задаваемые в этом сценарии, являются общими для

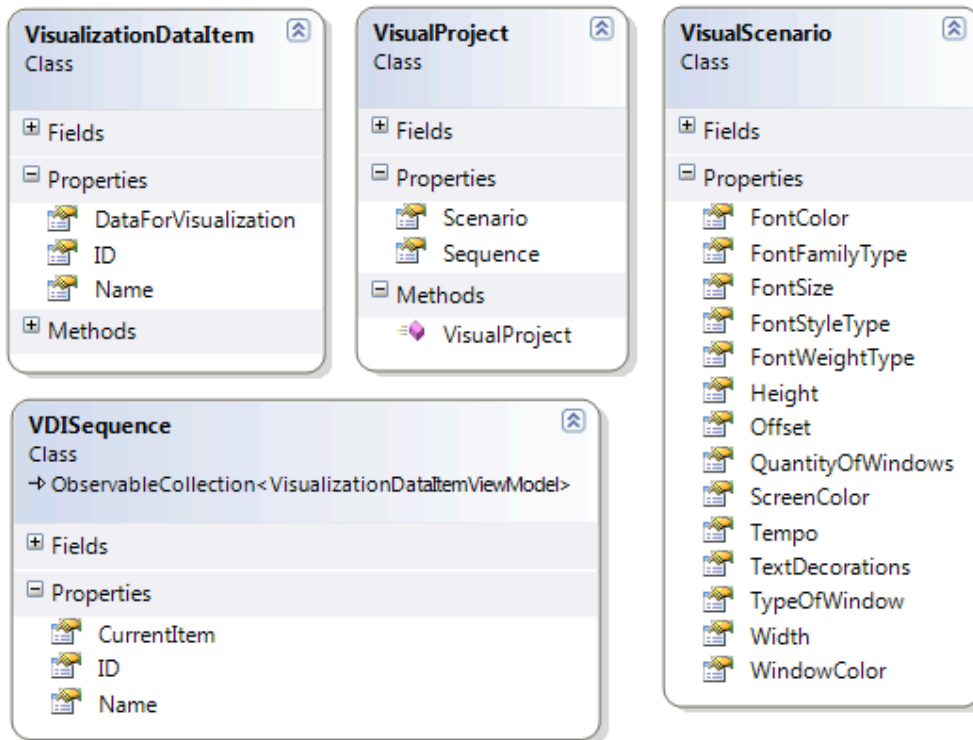


Рис. 4. Основные сущности СДВ

всех ВЕД из ПВД. Параметры, представленные в сценарии визуализации, можно разбить на несколько групп:

- параметры шрифта (гарнитура шрифта, кегль, текстовые декорации, способ начертания);
- геометрические параметры «окна» ВЕД (высота, ширина);

- цветовые параметры (цвет шрифта, цвет «окна», цвет фона);

- настройки визуализации (количество одновременно показываемых на экране «окон», вид «окна» расстояние между ними, траектория расположения (рис. 5), темп предъявления их на экране).

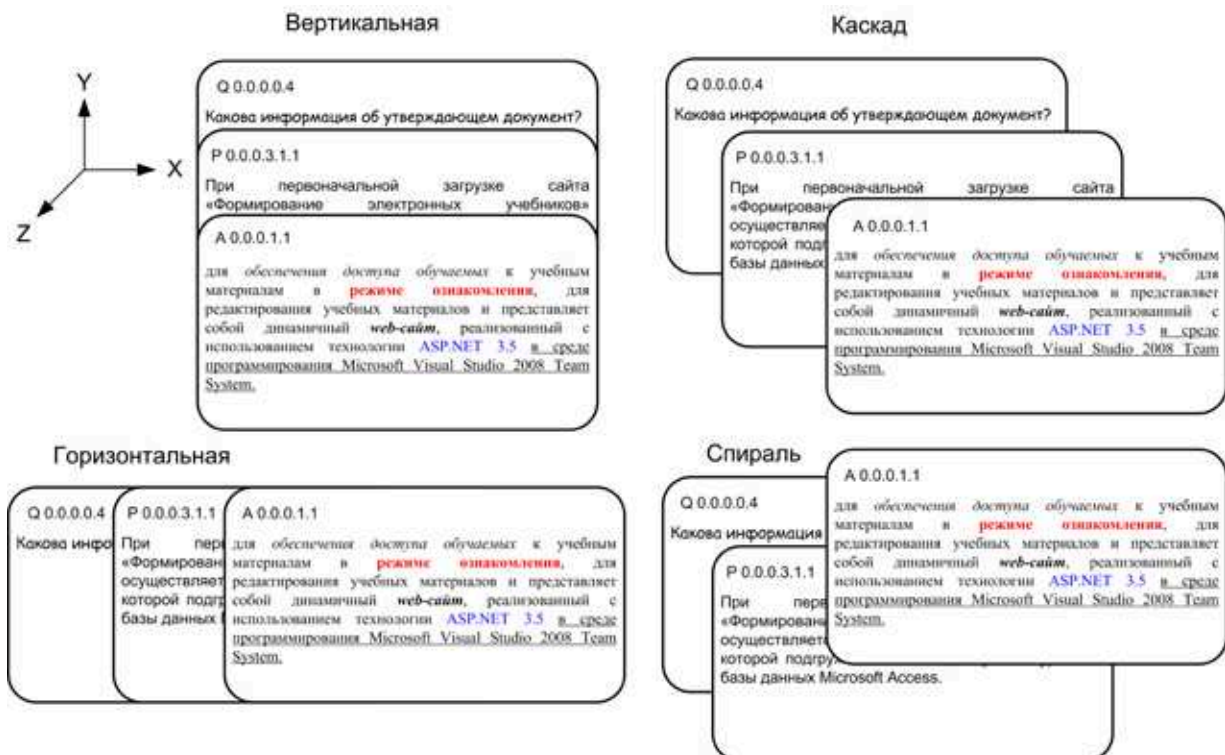


Рис. 5. Возможные траектории предъявления ВЕД на экране

Для каждой в отдельности ВЕД в дополнение к общему сценарию визуализации можно задать индивидуальные параметры визуализации. Такими индивидуальными параметрами визуализации могут быть:

- выделение отдельных слов, словосочетаний в тексте посредством применения различных параметров шрифта;
- задание для отдельных слов, словосочетаний эффектов вращения, увеличения и т. п.;
- создание индивидуальной разметки расположения содержимого в «окне» для ВЕД.

Сценарий визуализации и ПВД образуют проект визуализации VisualProject (рис. 4).

СДВ спроектированы с использованием шаблона проектирования MVVM (Model-View-ViewModel) [11]. MVVM используется в СДВ для разделения модели данных и ее представления на экране монитора (рис. 6).

К виду (View) в СДВ относится класс VisualisationPanel (рис. 4) – основной класс, ответственный за динамическую визуализацию. Он визуализирует ПВД на экране монитора, а также класс GetDataForVisualizationPanel (рис. 4), позволяющий формировать ПВД, и класс VisualScenarioPanel (рис. 4), ответственный за задание параметров сценария визуализации. VisualisationPanel подписан на событие изменения значений свойств или команд, предоставляемых моделью вида (View Model), класса VisualizationProjectViewModel.

В качестве Модели (Model) выступает класс VisualizationProject. Изменение модели посредством мо-

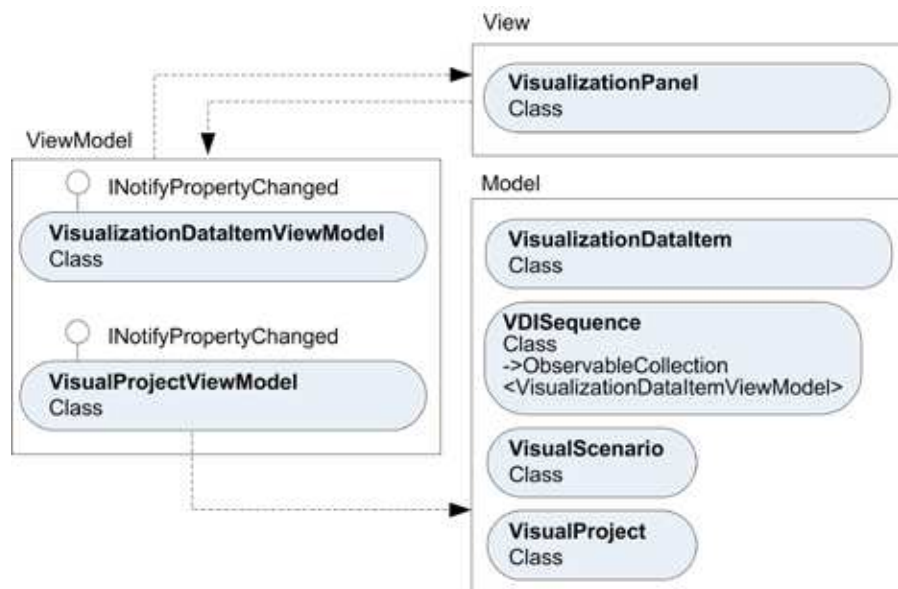


Рис. 6. Модель архитектуры СДВ

дели вида VisualProjectViewModel достигается через виды GetDataForVisualizationPanel и VisualScenarioPanel. Такая конструкция позволяет мгновенно реагировать на изменения как в сценарии визуализации, так и на добавление или удаление ВЕД из ПВД, что, в свою очередь, сразу же отображается на виде VisualizationPanel.

4 МАСТЕР СОЗДАНИЯ ПРОЕКТА ВИЗУАЛИЗАЦИИ ВОПРОСНО-ОТВЕТНЫХ ЕДИНИЦ

В СДВ для быстроты создания проекта визуализации проектировщикам доступен мастер создания проекта визуализации. Его работа состоит из трех этапов: выбора данных для визуализации, задание параметров визуализации, общих для всей ПВД, и задание параметров визуализации, индивидуальных для каждой в отдельности ВЕД.

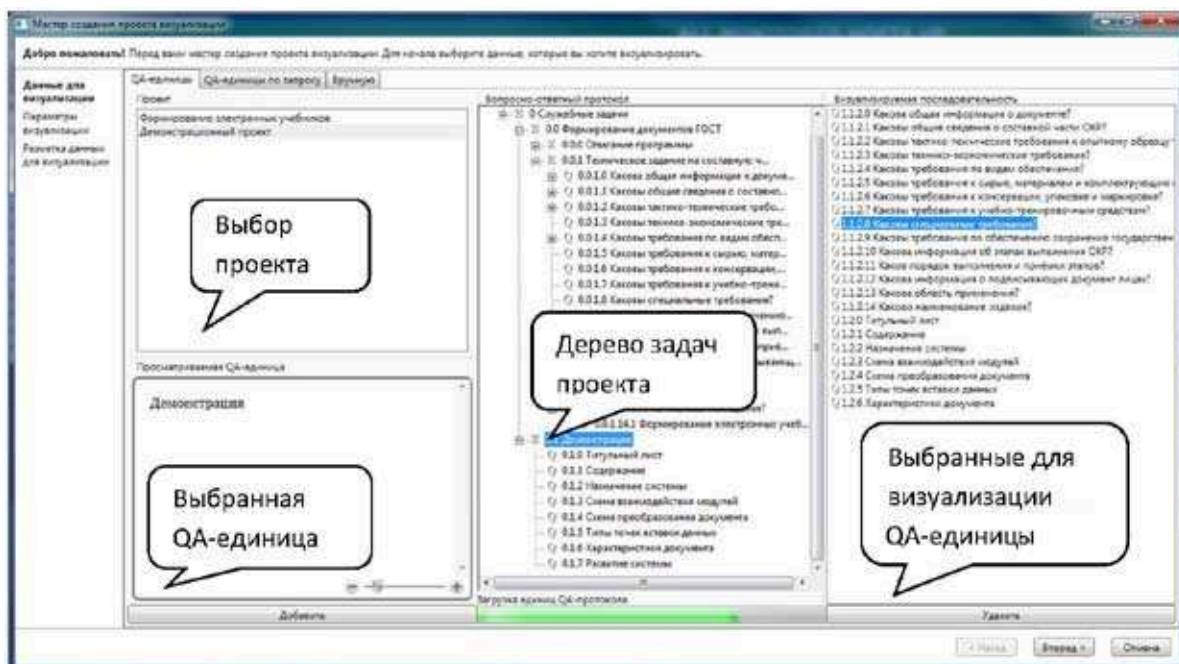


Рис. 7. Создание ПВД. Ручной выбор QA-единиц

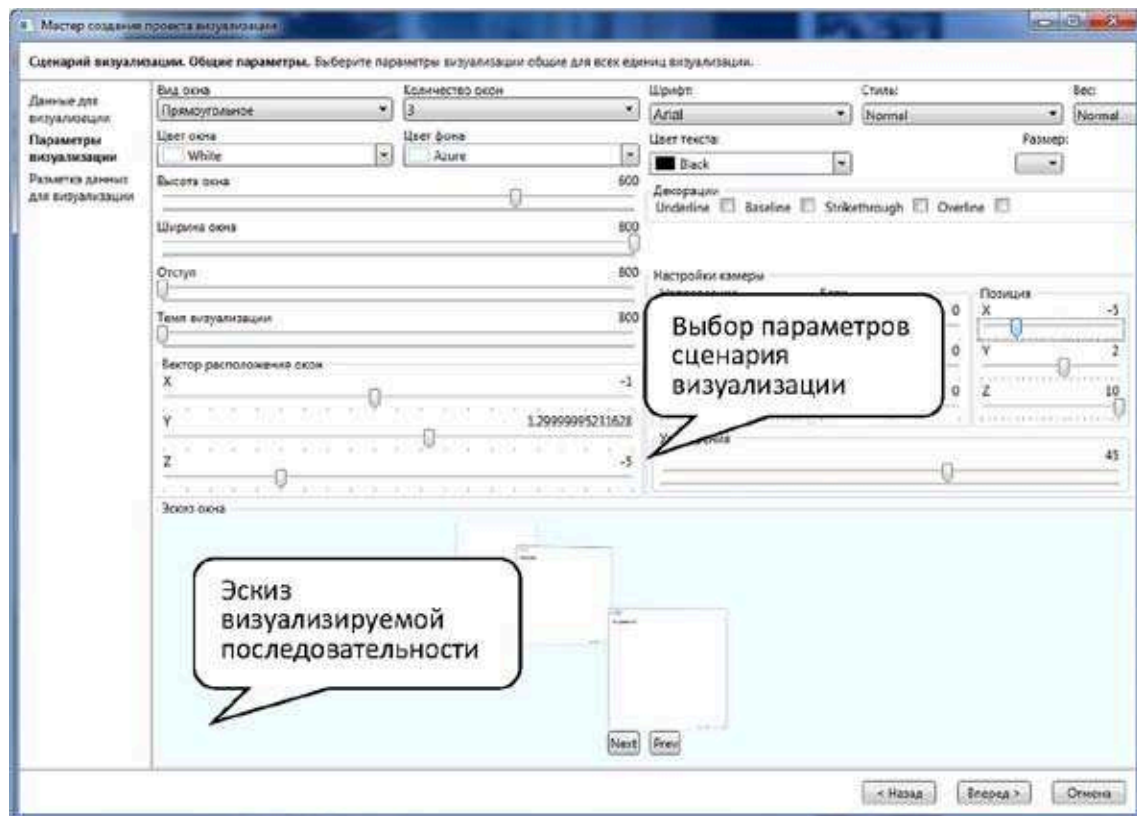


Рис. 8. Настройка общих параметров сценария визуализации

Визуализируемыми данными в СДВ являются QA-единицы из QA-протокола. Мастер позволяет выбирать QA-единицы двумя способами. В первом случае проектировщику показывается весь перечень проектов и относящиеся к ним деревья QA-протоколов (рис. 7).

Все, что остается сделать проектировщику, – это выбрать в дереве QA-протокола нужную ему QA-единицу и перетащить (Drag&Drop) ее в список визуализируемых единиц, или просто нажать кнопку «Добавить». Также возможно автоматическое добавление списка QA-единиц, являющихся дочерними по отношению к родительскому узлу дерева QA-протокола, и добавление всех QA-единиц, нижестоящих по отношению к выбранному узлу (рекурсивный обход дерева).

Во втором случае проектировщик может отобразить нужные единицы для визуализации, используя фильтр. Фильтр применяется для поиска по всей базе QA-протоколов. Полями, по которым можно задать фильтрацию, являются: принадлежность к определенному проекту, тип QA-единиц, статус QA-единицы, текст QA-единицы, временной интервал создания единицы, автор единицы. После применения фильтра формируется список отобранных QA-единиц, в который можно добавить все или выбрать самостоятельно только необходимые. Кроме визуализации QA-единиц, можно создавать собственные ВЕД, содержащие текст и графическую информацию.

После формирования ПВД проектировщик на следующем шаге задает общие параметры сценария визуализации, применяемые ко всей ПВД. Производимые изменения параметров мгновенно отображаются на эскизе, показы-

вающим как в данный момент выглядит визуализируемая последовательность (рис. 8).

На следующем шаге проектировщик получает возможность задавать параметры визуализации, индивидуальные для каждой ВЕД в отдельности. Эти индивидуальные параметры позволяют переопределять общие параметры сценария визуализации и добавлять специальные эффекты визуализации (эффекты вращения, увеличения и т. п.). Сделать это можно как в автоматизированном режиме при помощи инструментальной панели, так и в ручном режиме, используя редактор разметки XAML. Панель настройки индивидуальных эффектов визуализации можно вызвать дважды щелкнув на ВЕД (рис. 9).

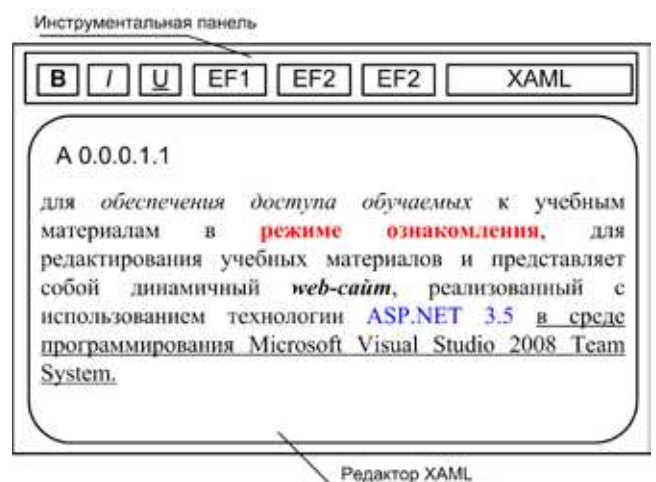


Рис. 9. Настройка индивидуальных параметров визуализации для ВЕД

ЗАКЛЮЧЕНИЕ

Рассмотренные в статье СВД вопросно-ответных единиц процессора WIQA, построенные с применением техник «активного зрительного восприятия», позволяют ускорить процесс визуального поиска вопросно-ответных единиц и улучшить восприятие их содержания, тем самым сокращая время, затраченное на принятие проектных решений при проектировании АС. СВД разработаны с учетом недостатков, присущих средствам визуализации передовых технологий разработки АС, таким как RUP и Microsoft ALM. Для облегчения создания проектов визуализации в средствах используется специальный мастер создания проектов визуализации.

СПИСОК ЛИТЕРАТУРЫ

1. Rational Unified Process. – URL: <http://www-01.ibm.com/software/awdtools/rup/>.
2. Application Lifecycle Management. – URL: <http://www.microsoft.com/visualstudio/eng/alm>.
3. Соснин П.И. Вопросно-ответное программирование человеко-компьютерной деятельности. – Ульяновск : УлГТУ, 2010. – 240 с.
4. Ware C. Visual Thinking for Design. – Morgan Kaufmann Publishers, 2008. – 202 p.
5. Windows Presentation Foundation. – URL: <http://msdn.microsoft.com/en-us/library/ms754130.aspx>.
6. Sosnin P. Question-Answer Approach to Human-Computer Interaction in Collaborative Designing. Chapter in the book “Cognitively Informed Intelligent Interfaces: Systems Design and Development” Published IGI Global, 2012, pp. 157-176.
7. Sosnin P. Experiential Human-Computer Interaction in Collaborative Designing of Software Intensive Systems // In Proc. of the 11-th International conference on Software Methodology and Technics (SoMeT’2012), Genua, Itali, 2012, pp. 180-197.
8. Overview of the .NET Framework. – URL: <http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>.
9. DirectX Graphics and Gaming. – URL: <http://msdn.microsoft.com/en-us/library/windows/apps/ee663274.aspx>.
10. Overview of XAML. – URL: <http://msdn.microsoft.com/en-us/library/ms752059.aspx>.
11. WPF Apps With The Model-View-ViewModel Design Pattern. – URL: <http://msdn.microsoft.com/en-us/magazine/dd419663.aspx>.