

УДК 681.3.06

Л. В. Рудикова

ИСПОЛЬЗОВАНИЕ ВОЗМОЖНОСТЕЙ POWERDESIGNER ДЛЯ ПОДДЕРЖКИ ЗАДАЧ ПРОЕКТИРОВАНИЯ

Дается характеристика по использованию CASE-средства PowerDesigner для поддержки задач проектирования информационных систем, а также описывается методология, лежащая в основе их создания. Дается детальная характеристика имеющихся средств, с помощью которых проводится моделирование данных, а также – объектное моделирование.

Основные возможности PowerDesigner

Современные разработчики различных информационных систем активно используют в своей деятельности различные CASE-средства (Computer-Aided Software/system Engineering). Кроме специализированных CASE-пакетов, поддерживающих решение определенных задач на конкретной стадии разработки проекта, существуют так называемые интегрированные комплексы программного обеспечения, которые позволяют автоматизировать, практически, весь жизненный цикл разрабатываемой информационной системы. Так, данные интегрированные комплексы можно использовать, начиная на этапе изучения и анализа предметной области и заканчивая непосредственно внедрением и поддержкой информационной системы.

Можно перечислить следующие средства интегрированных CASE-пакетов:

- репозиторий – средство организации совместной работы разработчиков, обеспечивающий хранение версий проекта, синхронизацию обновлений, контроль целостности;
- визуальные средства анализа и проектирования – поддерживают процесс разработки проекта (UML-диаграммы, ER-диаграммы, DFD-диаграммы и др.);
- прямой инжиниринг – поддерживает генерацию модели более низкого уровня или генерацию программного кода из рассматриваемой модели;
- обратный инжиниринг (реинжиниринг) – обеспечивает преобразование из модели или программного кода в рассматриваемую модель более высокого уровня;
- среда быстрой разработки приложений (RAD) – содержит оболочку и компилятор языка 4GL, генераторы программных кодов, различные мастера и др.;
- система документирования;
- средства тестирования.

Power Designer представляет собой интегрированное CASE-средство, поддерживающее разработку информационных систем и позволяющее аналитикам, проектировщикам и разработчикам работать в совместной используемой среде, создавать согласованные и надежные приложения.

Отметим некоторые, наиболее существенные возможности Power Designer.

Моделирование бизнес-процессов. С использованием средств Power Designer в различных организациях могут разрабатываться и моделироваться бизнес-процессы, ориентированные лишь на бизнес-задачи. Соответствующие диаграммы поддерживают генерацию и реинжиниринг XML-кода.

Моделирование данных. PowerDesigner позволяет разрабатывать и генерировать схему базы данных посредством двухуровневого (концептуального и физического) моделирования реляционной базы данных, которое поддерживает классические методики проектирования баз данных. Power Designer имеет также встроенные средства моделирования хранилища данных.

Объектное моделирование. В Power Designer предложена также технология анализа и проектирования систем с использованием стандарта UML (диаграммы бизнес-процессов, последовательности выполнения, классов и компонентов). На основе диаграммы классов Power Designer автоматически осуществляет генерацию и реинжиниринг кода для популярных инструментальных сред, таких как Java™, XML, Web Services, C++, PowerBuilder, Visual Basic и других, посредством настраиваемого генератора.

Моделирование отчетов. Различные встроенные средства редактора отчетов Power Designer позволяют моделировать и создавать разнообразные отчеты, которые могут понадобиться в тех либо иных приложениях. Усовершенствованный генератор отчетов функционирует независимо от модели. Доступны как стандартные, так и настраиваемые структуры отчетов, в том числе многомодельные отчеты, которые обеспечивают обзор проекта. Отчеты могут включать в себя диаграммы по любому типу моделей и могут быть сохранены в форматах RTF или HTML.

Создание XML-моделей. Поддержка и создание XML-схем с последующей кодогенерацией актуальны при разработке широкого класса приложений. Графическое представление позволяет просмотреть весь схе-

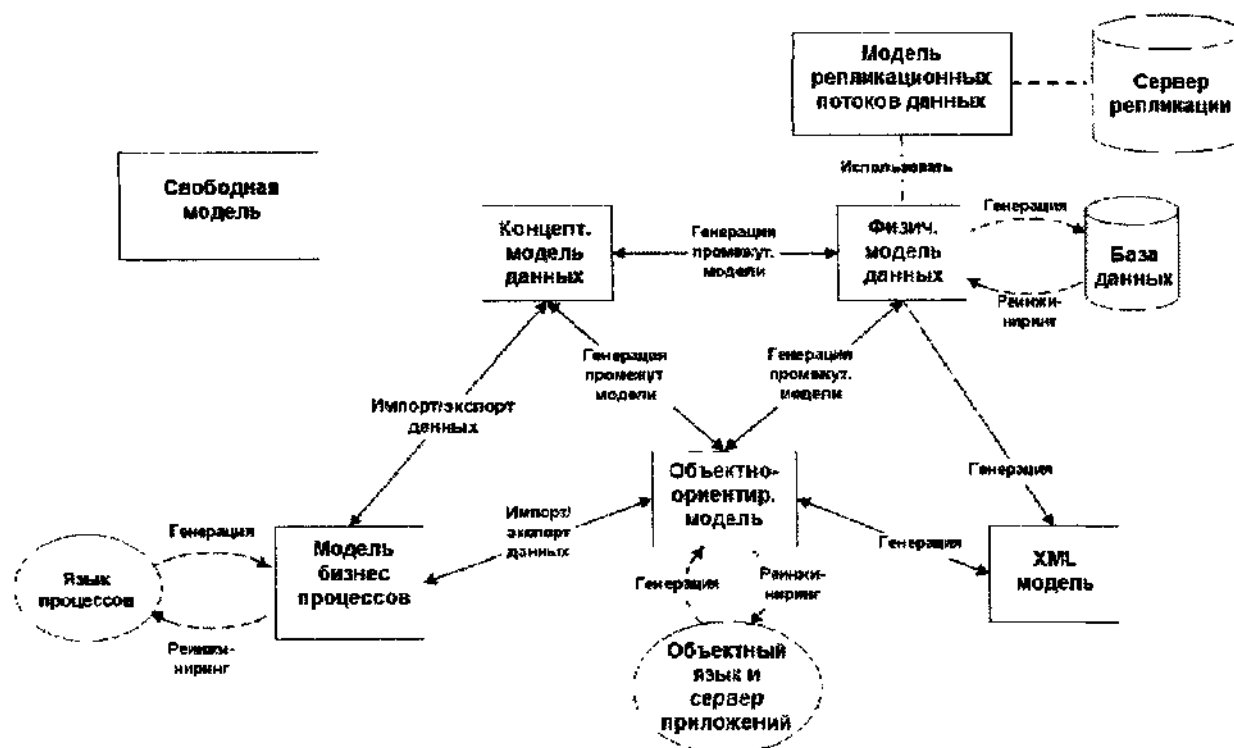


Схема преобразования моделей в среде Power Designer 12.5

матический вид XML-файла, что актуально при проверке либо модификации сложной структуры соответствующего XML-файла.

Свободное моделирование. Создание различного рода схем и диаграмм, предназначенных для общего представления деятельности организаций, иерархических структур, разнообразных моделируемых ситуаций и т. д. предназначено, прежде всего, для нетехнических специалистов и менеджеров.

Репозиторий. Репозиторий позволяет всем разработчикам легко просматривать модели и другую информацию, а также осуществлять обмен ими. Репозиторий обладает высокой масштабируемостью и поддерживает систему безопасности, основанную на роли пользователя, контроль версий, поиск и возможности составления отчетов.

На рисунке представлена схема преобразования моделей, которые можно подготовить средствами, предлагаемыми данной средой моделирования последней версии. Как и ранние версии, обновленный PowerDesigner 12.5 является интегрированной средой моделирования и проектирования современных бизнес-приложений, в которой объединены одновременно средства анализа и моделирования бизнес-процессов, структур баз данных, объектно-ориентированного моделирования на базе UML-диаграмм. Кроме того, в PowerDesigner 12.5 расширены возможности по созданию модели требований (Requirement Model) пользователя, а также добавлен мастер подготовки моделей репликационных потоков данных (Liquidity Model).

Таким образом, PowerDesigner является мощным инструментальным средством, позволяющим поддерживать весь процесс проектирования, начиная от построения свободных моделей на этапе анализа предметной области и заканчивая этапом поддержки информационной системы.

О методологии разработки информационных систем

В общем случае разработка информационных систем предполагает процесс создания ИС и обеспечивает управление данным процессом для того чтобы гарантировать выполнение требований как к самой ИС, так и к характеристикам процесса разработки. Для успешного создания сложных систем, к которым относятся различные корпоративные ИС, недостаточно иметь только современные платформы и средства, но следует также значительное внимание уделять методологии создания ИС.

На наш взгляд, в качестве составляющих методологии можно выделить следующие аспекты, которые в той или иной мере используются современными разработчиками ИС:

- временная итерационная спиральная модель жизненного цикла ИС;
- методология анализа ИС на основе бизнес-процессов;

- методология временных функциональных моделей;
- методология проектирования данных;
- комплекс инструментальных средств разработки, создания и поддержки ИС;
- комплекс личностно-ориентированной мотивации и взаимодействия с ИС.

В основу временной итерационной спиральной модели жизненного цикла ИС положена спиральная модель жизненного цикла ИС, которая активно развивается с 90-х гг. XX в. Методология описывает процесс создания и сопровождения информационных систем в виде жизненного цикла (ЖЦ) ИС, представляя его в виде последовательности стадий, каждая из которых разбита на этапы, и выполняемых на них процессов. Для каждого этапа определяются последовательность выполняемых работ, получаемые результаты, методы и средства, необходимые для выполнения работ, роли и ответственность участников и т. д. Временная спиральная модель включает стадии анализа, проектирования, разработки, тестирования и интеграции, внедрения, сопровождения и развития ИС. Причем добавление временной компоненты предлагает в данном случае отражение всех аспектов, которые связаны с временным развитием ИС, и предполагает взаимовлияние и соответственно корректировку всех этапов ЖЦ ИС даже после введения ИС в эксплуатацию. Именно этот момент накладывает также требование дополнительного рассмотрения комплекса личностно-ориентированной мотивации и взаимодействия с ИС.

Методология анализа ИС на основе бизнес-процессов предполагает тщательное изучение предметной области ИС. На начальном этапе необходимо точно отразить цели и задачи организации. Кроме того, нужно выяснить все требования заказчиков к ИС, а затем преобразовать их на языке моделей в требования к разработке проекта ИС так, чтобы обеспечить соответствие целям и задачам организации. В процессе описания организации и ее деятельности, как правило, создаются три основные модели организации: стратегическая, укрупненная и детальная. Все эти модели отображают основные аспекты деятельности организации и базируются на бизнес-процессах.

Комплекс временных функциональных моделей определяет процесс создания корпоративных информационных систем как процесс построения и последовательного развития всех моделей, начиная от моделей, описывающих деятельность организации, и заканчивая готовой информационной системой. Все создаваемые модели должны контролироваться, преобразовываться, согласовываться с помощью соответствующих CASE-средств и сохраняться в репозитории.

В настоящее время данные для любой организации являются наиболее стабильной ее составляющей. В силу этого проектированию данных отводится значительное место при построении корпоративной ИС. В процессе проектирования модели данных развиваются от начальной наиболее простой версии в законченную спецификацию приложения, используемую для генерации кода под соответствующий сервер базы данных. Как правило, на основе корректной модели данных можно в дальнейшем построить и реорганизовать многие клиентские приложения, создаваемые для организации.

Комплекс согласованных инструментальных средств обеспечивает непрерывный цикл автоматизации процессов, выполняемых на всех этапах ЖЦ ИС. Комплекс средств такого рода позволяет строить модели, описывающие деятельность организации, формировать требования к ИС, быстро переходить от моделей требований к ИС к проекту приложений и баз данных. Таким образом, данные средства обеспечивают поддержку при разработке приложений, их тестирование и интеграцию в систему.

И наконец, комплекс личностно-ориентированной мотивации и взаимодействия с ИС предполагает заинтересованность каждого работающего в организации в надлежащем функционировании ИС, выявлении слабых мест системы и во внесении своевременных предложений по улучшению и реорганизации ИС.

Итак, рассмотренные выше составляющие методологии обеспечивают:

- коллективную разработку системы;
- переход от одного этапа разработки к другому;
- контроль результатов разработки;
- итеративный характер разработки (возможность пересмотра полученных результатов и возврата на любой из предыдущих этапов);
- управление конфигурацией;
- личностную мотивацию по развитию и расширению ИС.

Моделирование данных в PowerDesigner

Как указывалось выше, данные являются основной составляющей организации. В PowerDesigner предусмотрены соответствующие средства, поддерживающие объектно-реляционное моделирование данных. Так, для создания схемы базы данных используется двухуровневая модель данных.

Концептуальная модель данных (CDM) представляет собой формализованное представление общей структуры данных информационной системы без привязки к конкретной реализации на базе СУБД. Модель строится в виде графической схемы, в основе которой лежит диаграмма «сущность-связь». Как правило,

при работе на концептуальном уровне создается абстрактная схема данных; выделяются сущности из предметной области; устанавливаются связи между ними; задаются бизнес-правила работы с данными. Для построения концептуальной модели, в Power Designer используются нотации Entity/Relationship, Merise, Entity/Relationship+Merise и IDEF1X. Следует напомнить, что нотация Entity/Relationship+Merise (E/R+Merise) является смешанной и содержит объекты для моделирования по нотациям Entity/Relationship и Merise.

Физическая модель данных (PDM) соответствует реализации концептуальной модели на конкретной выбранной платформе реализации базы данных (СУБД) и реально представляет собой логическую модель данных (также – физическая схема). Работая на логическом уровне, уточняют и задают типы данных; проводят денормализацию; создают индексы и триггеры на SQL-диалекте выбранной СУБД и т. п. Физическая схема имеет однозначное отображение в DDL-скрипт (набор инструкций на языке SQL для создания базы данных). Графическая интерпретация модели может быть представлена в следующих нотациях: Relational (стрелка связи указывает на первичный ключ родительской таблицы), CODASYL (стрелка указывает на внешний ключ дочерней таблицы), Conceptual (аналогично CDM) и IDEF1X.

Двухуровневая модель данных с возможностью прямого и обратного инжиниринга, а также слияние моделей (merge) позволяет вести итерационную разработку структуры данных. Так, например, системный аналитик подготовил концептуальную модель (CDM), которая преобразована в физическую (PDM). Далее, физическая модель уточняется программистом и отлаживается на СУБД. Затем проводится обратный инжиниринг и обновление существующей CDM-модели, которая дорабатывается системным аналитиком уже на следующей итерации.

Что же касается объектно-реляционного подхода, то он реализуется только на этапе создания концептуальной модели и поддерживает наследование. PowerDesigner поддерживает одинарное наследование двух типов: обычное наследование, когда одной записи родительской сущности могут соответствовать наборы атрибутов всех дочерних сущностей, и исключаяющее наследование, когда только одна дочерняя запись связана с записью родительской сущности. При переходе от концептуальной модели к физической родительские и дочерние сущности, участвующие в наследовании, преобразуются различным способом во множество таблиц одного уровня и связей между ними.

Следует отметить, что реализованное в PowerDesigner объектно-реляционное моделирование данных совмещает преимущества реляционной модели и объектно-ориентированную направленность архитектуры современных приложений. Как видно из рисунка, можно простой генерацией получить из концептуальной модели объектно-ориентированную модель данных.

Однако нельзя не отметить и следующие недостатки, которые присущи объектно-реляционному подходу. *Во-первых*, структура класса жестко привязана к структуре таблицы, а, *во-вторых*, отсутствие полноценной реализации объектно-ориентированной среды.

Объектное моделирование в PowerDesigner

PowerDesigner поддерживает также и графические нотации языка UML. Однако в новой версии расширены возможности по построению тех или иных диаграмм, что, непосредственно, способствует разработке и документированию моделей сложных информационных систем различного назначения.

Так, разработчик может использовать четыре вида диаграмм: структурные, вариантов использования, динамические и реализации.

Структурные диаграммы позволяют пользователю моделировать статическую структуру его информационной системы. PowerDesigner предоставляет четыре типа диаграмм для такого моделирования систем, каждая из которых предлагает различное представление объектов и их связей. *Диаграмма классов* (class diagram) представляет статическую структуру классов, составляющих систему. *Диаграмма классов* предназначена для определения видов объектов, которые будут составлять систему, и отношений, которыми они будут связаны между собой. *Составная структурная диаграмма* (composite structural diagram) позволяет определить более детально внутреннюю структуру классов и отношений, которыми они будут связаны друг с другом. В частности, составная структурная диаграмма используется для моделирования комплексных форм структуры, которая может быть очень объемной при моделировании диаграммы классов. *Диаграмма объектов* (object diagram) подобна диаграмме классов, за исключением того, что диаграмма объектов показывает отдельные экземпляры классов. *Диаграмма объектов* используется для представления моментального снимка отношений между действующими сущностями классов. *Диаграмма пакетов* (package diagram) отображает структуру пакетов, которые составляют приложение, и отношения между ними.

Диаграмма вариантов использования (use case diagram) является диаграммой высокого уровня для исследования требований и поведения системы. Данная диаграмма является частью объектно-ориентированной модели, графически конструирует набор функциональных возможностей системы в форме вариантов

использования и показывает как внешние пользователи (актеры) могут взаимодействовать с системой. Диаграмма вариантов использования хорошо подходит для задачи описания всех ситуаций, которые могут происходить в системе баз данных при использовании ее всеми пользователями. Однако данная диаграмма плохо подходит для описания сетевого протокола TCP/IP, т. к. в данном случае существует много исключительных ситуаций, ветвящихся действий и условной функциональности (что случится, когда соединение разорвется; что случится, если пакет потеряется и т. п.).

В PowerDesigner предоставлены 5 видов *динамических диаграмм*. *Диаграмма коммуникации* (communication diagram) предназначена для представления функциональности системы с точки зрения взаимодействия между объектами, при акценте на структуру объекта. *Диаграмма последовательности* (sequence diagram) отображает функциональности системы с точки зрения взаимодействий объектов, при акценте на хронологию поступления сообщений между объектами системы. *Диаграмма деятельности* (activity diagram) представляет функциональности системы с точки зрения действий или активности выполнения и переходов, вызванных завершением, какого-либо из этих действий, на ней можно также представлять условные выполняемые переходы. *Диаграмма состояний* (statechart diagram) предназначена для представления функциональности системы с точки зрения состояний и взаимодействий между ними. *Обзорная диаграмма взаимодействий* (interaction overview diagram) описывает взаимодействие в системе на высоком уровне.

Диаграммы реализаций предназначены для моделирования физической среды системы и представлены двумя диаграммами. *Диаграмма компонентов* (component diagram) представляет систему в виде отдельно разделенных компонент или подсистем. *Диаграмма развертывания* (deployment diagram) позволяет представить среду выполнения для проекта, на ней отображаются аппаратные средства, на которых будут выполнены описанные компоненты системы, а также связи данных аппаратных средств между собой.

Итак, легко видеть, что в распоряжении разработчика PowerDesigner дополнительно представлены четыре диаграммы, три из которых относятся к структурным диаграммам, а одна – к динамическим.

Основные выводы

В статье рассмотрены основные возможности интегрированной среды разработки PowerDesigner 12.5. Приведена методология, лежащая в основе создания информационных систем. Кроме того, дана детальная характеристика имеющихся средств, с помощью которых проводится моделирование данных, а также – объектное моделирование.

Отметим также, что в PowerDesigner одновременно можно решать несколько различных задач моделирования, не ограничивая проектировщиков жесткими рамками. Например, разрабатывать проект базы данных и/или объектно-ориентированную модель бизнес-логики. Таким образом, PowerDesigner представляет собой единую среду разработки, объединяющую как проектировщиков баз данных, так и разработчиков приложений.

Для демонстрации возможностей PowerDesigner разработаны концептуальная и физическая модели для базы данных «Туристическое агентство», а также проведено объектное моделирование для предметной области, связанной с поддержкой учебного процесса на любом факультете вуза.

Литература

1. Калянов, Г. Н. Case-технологии. Консалтинг в автоматизации бизнес-процессов / Г. Н. Калянов. 3-е изд. – М. : Горячая линия-Телеком, 2002.
2. Рудикова, Л. В. Некоторые аспекты, касающиеся проектирования данных и возможность использования PowerDesigner / Л. В. Рудикова // Современные информационные компьютерные технологии в учебном процессе, научных исследованиях и управлении университетом: материалы открытой науч.-практ. конф. преподавателей и студентов математического факультета. 26–27 апреля 2005 года. – Гродно : ГрГУ, 2005. – С. 183–188.
3. Рудикова, Л. В. Использование средств PowerDesigner для поддержки задач проектирования / Л. В. Рудикова // Управление в социальных и экономических системах: материалы XV Междунар. науч.-практ. конф. (6–4 июня 2006 г., г. Минск) / редкол.: Н. В. Суша (пред.) и др.; Минский ин-т управления. – Минск : Изд-во МИУ, 2006. – С. 211–212.

Рудикова Лада Владимировна, доцент кафедры информатики и вычислительной техники факультета математики и информатики Гродненского государственного университета имени Янки Купалы, кандидат физико-математических наук, доцент, rudikowa@grsu.by