МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное автономное бюджетное образовательное учреждение высшего образования

Севастопольский государственный университет

кафедра Информационных систем

Институт информационных технологий и управления в технических системах

курс 1 группа ИC/б-11-о

Лисянский Александр Игоревич

09.04.02 Информационные системы и технологии (уровень магистра)

**РЕФЕРАТ**

По дисциплине Технологии Проектирования ИС

«Методология разработки DATARUN»

Отметка о зачёте \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Руководитель практикума

Профессор Доронина Ю. В.

(должность) (подпись) (инициалы, фамилия)

Севастополь

2016

Оглавление

[Введение 2](#_Toc472280222)

[1. Характеристики современных проектов. Определение проектирования. Объект проектирования 3](#_Toc472280223)

[2. Программные средства поддержки жизненного цикла ПО 4](#_Toc472280224)

[3. Методология DATARUN 6](#_Toc472280225)

[4. Основные этапы методологии DATARUN 8](#_Toc472280226)

[5. CASE-система SILVERRUN 10](#_Toc472280227)

[6. Подход к представлению проектной информации 12](#_Toc472280228)

# Введение

Основные особенности современных проектов ИС. Этапы создания ИС: формирование требований, концептуальное проектирование, спецификация приложений, разработка моделей, интеграция и тестирование информационной системы. Методы программной инженерии в проектировании ИС. Проектирование (от латинского projectus, что означает “брошенный вперед”) – это процесс составления описания, необходимого для создания в заданных условиях еще не существующего объекта по первичному описанию этого объекта путем его детализации, дополнения, расчетов и оптимизации. Описание объекта может быть задано по-разному: в виде текста, алгоритма, программы, чертежа, таблицы или, что чаще всего, комбинировано в традиционно бумажном или электронном виде.

Таким образом, главной особенностью проектирования является работа с еще не существующим объектом. В этом отличие проектирования от моделирования, где объект не может не существовать.

# 1. Характеристики современных проектов. Определение проектирования. Объект проектирования

Проектирование можно рассматривать с одной стороны, как заключительную фазу исследований, а с другой как начальную фазу производства.

Проектирование ИС охватывает три основные области:

\* проектирование объектов данных, которые будут реализованы в базе данных;

\* проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;

\* учет конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры (файл-сервер или клиент-сервер), параллельной обработки, распределенной обработки данных и т.п.

Проектирование информационных систем всегда начинается с определения цели проекта. В общем виде цель проекта можно определить, как решение ряда взаимосвязанных задач, включающих в себя обеспечение на момент запуска системы и в течение всего времени ее эксплуатации:

\* требуемой функциональности системы и уровня ее адаптивности к изменяющимся условиям функционирования;

\* требуемой пропускной способности системы;

\* требуемого времени реакции системы на запрос;

\* безотказной работы системы;

\* необходимого уровня безопасности;

\* простоты эксплуатации и поддержки системы.

# 2. Программные средства поддержки жизненного цикла ПО

Современные методологии и реализующие их технологии поставляются в электронном виде вместе с CASE-средствами и включают библиотеки процессов, шаблонов, методов, моделей и других компонент, предназначенных для построения ПО того класса систем, на который ориентирована методология. Электронные методологии включают также средства, которые должны обеспечивать их адаптацию для конкретных пользователей и развитие методологии по результатам выполнения конкретных проектов.

Процесс адаптации заключается в удалении ненужных процессов, действий ЖЦ и других компонентов методологии, в изменении неподходящих или в добавлении собственных процессов и действий, а также методов, моделей, стандартов и руководств. Настройка методологии может осуществляться также по следующим аспектам: этапы и операции ЖЦ, участники проекта, используемые модели ЖЦ, поддерживаемые концепции и др.

Электронные методологии и технологии (и поддерживающие их CASE-средства) составляют ядро комплекса согласованных инструментальных средств среды разработки ИС.

Методология DATARUN опирается на две модели или на два представления:

\* модель организации;

\* модель ИС.

Методология DATARUN базируется на системном подходе к описанию деятельности организации. Построение моделей начинается с описания процессов, из которых затем извлекаются первичные данные, которые описывают продукты или услуги организации, выполняемые операции (транзакции) и потребляемые ресурсы. Они становятся основой для проектирования архитектуры ИС. Архитектура ИС будет более стабильной, если она основана на первичных данных, тесно связанных с основными деловыми операциями, определяющими природу бизнеса, а не на традиционной функциональной модели.

# 3. Методология DATARUN

Высокая динамичность рынка требует от организаций быстрого развития информационно- технологической инфраструктуры. Одной из ее наиболее важных и дорогостоящих составляющих является информационная система, для реализации которой применяются современные технологии: архитектура клиент/сервер, распределенные базы данных, сложные сети коммуникаций, развитые интерфейсы пользователя. Все это ставит перед разработчиком проблему выбора инструментальных средств и технологий для ведения проекта.

Создание сложной информационной системы невозможно без единого интегрированного подхода к процессу разработки. Такой подход часто оформляется в виде коммерчески доступной методологии проектирования. Методология служит двум целям: 1) обеспечивает концептуальную основу для всего процесса разработки; 2) предоставляет технологию руководства проектом.  
Многие методологии применялись в течение ряда лет с разной степенью успеха. Часто разнообразие используемых в них моделей приводит к получению огромного количества документации, не сосредоточенной на результатах. Множественные перекрывающиеся модели процессов и данных создают избыточность, которая преподносится как перекрестный контроль.

DATARUN - уникальная концепция в ряду методов. Эта методология гарантирует, что на каждой стадии выполняется только существенная для целей проекта работа, облегчающая быстрое создание приложений. Повторения и избыточность в спецификациях исключаются, создается управляемая, основанная на моделях форма итеративной разработки. Исходные версии объектов доступны для непосредственного использования на следующих фазах проектного цикла. Создаваемая информационная система описывается рядом последовательных моделей, каждая из которых является развитием предыдущей и наследует правила и данные, определенные в более ранних моделях. Наследование свойств позволяет многократно использовать различные спецификации на всех уровнях прикладного проекта.

Методология DATARUN ведет заказчика и разработчика информационной системы по всем этапам жизненного цикла проекта, от стадии первоначальной экономической оценки затрат на проект до выхода реального приложения. Она позволяет координировать и контролировать работу всех групп лиц, занятых в работе над проектом.

Методология DATARUN обеспечена средствами автоматизированной поддержки:

Для управления проектной деятельностью имеется система Software Engineering Companion, позволяющая детально расписывать ведение проекта, распределять проектные роли среди исполнителей, контролировать выполнение заданий.

Детальное изложение техник моделирования данных и бизнес-функций, проектирования баз данных, создания приложений содержится в гипертекстовой системе Software Engineering Guidelines.

Автоматизация проведения проектных работ обеспечивается CASE-системой SILVERRUN.

Предоставляемая этими средствами среда проектирования дает возможность руководителю проекта контролировать выполнение работ. Каждый участник проекта, подключившись к системе, может уточнить содержание и сроки выполнения порученной ему работы, изучить технику ее выполнения в гипертексте по технологиям, и вызвать инструмент (модуль SILVERRUN) для реального выполнения работы.

Такой автоматизированный комплекс поддержки выполнения проектов, основанный на современной методологии проектирования и эффективном CASE-средстве, создает все необходимые условия для быстрого создания сложных информационных систем с высоким качеством.

# 4. Основные этапы методологии DATARUN

**Построение бизнес-модели предметной области**

Строится функциональная (DFD) и информационная (ER) модели предметной области. При построениии функциональной модели выявляются первичные структуры данных, которые преобразуются в сущности ER-модели. Результат: Модель бизнес-процессов, содержащая Первичные структуры данных, и Концептуальная модель данных.

**Построение архитектуры информационной системы**

Принимается решение, из каких приложений (подсистем) будет состоять система. Анализируются существующие системы. Архитектура системы документируется в виде DFD, где функции представляют компоненты приложений с указанием используемой информации (путем ссылки на сущности и связи ER- модели). ER-модель также делится на группы сущностей, обрабатываемых приложениями. Результат: Архитектура информационной системы.

**Проектирование приложений (подсистем)**

На основе концептуальной ER модели строится реляционная модель данных. Части ER-модели, соответствующие различным приложениям, оформляются как подсхемы базы данных. Для каждого приложения создается (возможно, разными группами разработчиков) детальный проект. Строится модель системных процессов (программных функций) и подсхемы базы данных для каждой функции (спецификация интерфейса). Поскольку все приложения работают с подсхемами одной базы данных, обеспечивается их совместная работа. Результат: Реляционная модель данных. Для каждого приложения - Модель данных приложения, Модель системных процессов. Для каждого интерфейса в приложении - Спецификация интерфейса.

**Создание приложений**

Из модели базы данных генерируется код для ее создания на сервере. Программируются системные процессы. При этом возможно разнесение процессов по узлам распределенной системы (часть процессов реализуется как хранимые процедуры на сервере, часть как сервисы монитора транзакций, часть - как программы клиентской части). Интерфейс приложений (обычно составляющий до 70-80% всей системы) может быть быстро создан перенесением соответствующей ему подсхемы базы данных в среду языка 4-го поколения).

**Интеграция приложений**

Созданные приложения объединяются в единую среду и тестируются на совместимость. Поскольку все приложения строились на основе общей глобальной модели данных, достигается высокая степень интеграции.

# 5. CASE-система SILVERRUN

CASE-система SILVERRUN американской фирмы Сomputer Systems Advisers, Inc. (CSA) используется для инструментального обеспечения анализа и проектирования информационных систем бизнес-класса. Она применима для поддержки любой методологии, основанной на раздельном построении функциональной и информационной моделей (диаграмм потоков данных и диаграмм "сущность-связь").

Настройка на конкретную методологию обеспечивается выбором требуемого графического изображения символов моделей и набора правил проверки проектных спецификаций. В системе имеются готовые настройки для наиболее распространенных методологий: Gane/Sarson, Yourdon/DeMarco, Merise, Ward/Mellor, Information Engineering. Для каждого проектного понятия имеется возможность добавления собственных описателей.

SILVERRUN состоит из трех основных подсистем: модуля построения диаграмм потоков данных BPM (Business Process Modeler) и двух модулей построения диаграмм "сущность-связь": концептуальных моделей - модуль ERX (Entity Relationship eXpert) и реляционных моделей - модуль RDM (Relational Data Modeler). Каждый модуль является самостоятельным продуктом и может приобретаться и использоваться отдельно. Для интеграции подсистем в единое целое служит менеджер репозитория WRM (Workgroup Repository Manager).

Встроенный в модуль RDM генератор схем баз данных позволяет получить операторы определения данных (DDL) для 16 СУБД. Но для полного использования специфики каждой СУБД следует использовать отдельно приобретаемые мосты, позволяющие как получить схему базы данных из модели, так и построить модель существующей базы данных. SILVERRUN имеет мосты к следующим СУБД: DB2, Informix, Ingres, Oracle, Progress, SQL Base, SQL Server, Sybase.

Для обмена данными с языками разработки приложений также используются соответствующие мосты. В настоящее время существуют мосты к следующим средствам разработки приложений: Object Studio, Omnis 7, PowerBuilder, Progress, SQL Windows, Synon 2/E, Uniface.

Таким образом, из модулей можно собрать систему требуемого масштаба: от диаграммера до среды разработки приложений для конкретного языка программирования и СУБД. Заменяемость мостов позволяет использовать единые модели в ситуациях, когда разные подразделения или филиалы организации используют разные СУБД и средства разработки приложений.

Сама система SILVERRUN функционирует на четырех платформах: Windows, OS/2, Macintosh, Solaris. Коллективная разработка в стандартной версии поддерживается разделением и слиянием моделей. В версии SILVERRUN Enterprise поддерживается одновременный коллективный доступ к репозиторию. Между версиями разных платформ обеспечен обмен данными, что позволяет вести одновременную разработку в разнородной среде как в сетевом, так и в одиночном режимах.

# 6. Подход к представлению проектной информации

Проектные данные могут быть представлены множеством способов. Это спецификации функций, файлов баз данных, экранов ввода информации, бланков документов и картотек. Что же общего между этими столь разными представлениями? Это содержащиеся в них данные. В конечном счете, информационная система - это система хранения и обработки информации. Поэтому в SILVERRUN основой всех представлений являются данные. Они являются общей частью всех формализмов (типов описаний) и универсально обрабатываются всеми модулями.

Один и тот же элемент данных может в разных представлениях иметь разные названия и форматы. В SILVERRUN обеспечена возможность связывать элементы данных различных представлений с общим элементом (common item), выражающим смысл представленной этими элементами данных информации. Этот общий элемент может стать столбцом таблицы базы данных в новой системе или просто использоваться как универсальный термин для связи разных представлений.

**Моделирование процессов**

Диаграммы потоков данных, создаваемые в SILVERRUN, соответствуют современному развитию принципов структурного анализа. В системе имеется возможность изменять внешний вид символов диаграмм и выбирать отображаемые в них дескрипторы, включая определяемые пользователем. На рис. 1 показан экран определения нотации модуля BPM. Также можно выбирать набор правил, проверяемых процедурой анализа корректности модели. В SILVERRUN встроено несколько предопределенных нотаций и наборов правил, соответствующих наиболее известным методам построения DFD: Gane/Sarson, Yourdon/DeMarco, Merise, Ward/Mellor.

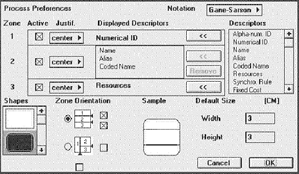


Рисунок 1 - Экран определения нотации модуля BPM

К специфическим особенностям системы относится возможность использования символьных иерархических идентификаторов процессов вместо числовых, а также удобные функции автоматической перенумерации объектов диаграмм в порядке расположения на схеме или в указанной пользователем последовательности. Система может строить дерево процессов и дает возможность переопределять их иерархическую вложенность перемещением процессов из одних узлов дерева в другие.

Для анализа и реинжениринга бизнес-процессов (BPR) имеется возможность определить используемые ресурсы и задать их удельную стоимость. При построении моделей можно указывать, какие ресурсы в каком объеме используются процессами. Система автоматически подсчитывает стоимость каждого процесса с учетом стоимости его подпроцессов, а также общую стоимость каждого ресурса по всей модели.

**Моделирование данных**

В процессе первичного анализа данных, собранных из разных источников, необходимо выявить основные информационные понятия (сущности) и их взаимосвязи. Эта деятельность поддерживается модулем ERX, в который встроена экспертная система, помогающая реструктурировать "сырую" информацию и привести ее к виду, допускающему реализацию в реляционной СУБД. На рис. 2 - модель "сущность-связь" в третьей нормальной форме, автоматически построенная модулем ERX из этой структуры и нормализованная в процессе ответов пользователя на задаваемые системой вопросы.

На рис. 3 показана созданная в ERX модель, перенесенная в модуль реляционного моделирования RDM и доработанная для непосредственной реализации в реляционной базе данных. В нижней части символов таблиц показаны действия над записями. В генерируемой мостом схеме базы данных это будут соответствующие триггеры. А три колонки справа от имен столбцов таблиц отражают действия, производимые над этими столбцами при операциях считывания, присваивания и модификации значений. Также можно определять глобальные действия на уровне всей модели (в схеме базы данных - хранимые процедуры). Таким образом достигается не только определение конструкций реляционной СУБД, но и объектно-ориентированное расширение реляционной модели.

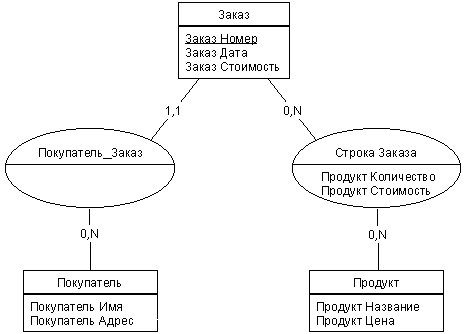


Рисунок 2 - Модель "сущность-связь", автоматически построенная модулем ERX

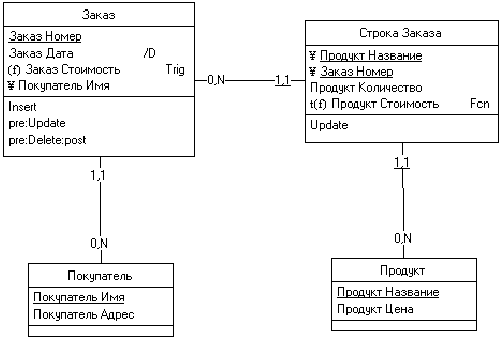


Рисунок 3 - Модель базы данных в модуле RDM

Структура данных, на основе которой будет строится ER-модель

Любые комбинации столбцов таблиц и связей можно выделить для генерации индексов. SILVERRUN сама может сгенерировать индексы для первичных и альтернативных ключей, а также операторы контроля ссылочной целостности на основе характеристик связей.

В модели имеются конструкции для подтипов и альтернативных связей. Причем отсутствует требование принадлежности только к одному подтипу, что позволяет моделировать встречающиеся на практике ситуации, не попадающие под определение категоризации в таких нотациях, как, например, IDEF1.

Для поддержки различных методологий модуль RDM предоставляет возможность переопределения нотации. На рис. 4 изображен экран выбора графических символов для представления различных характеристик связей.

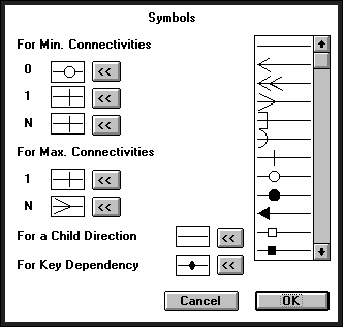


Рисунок 4 - Экран выбора символов для характеристик связей

Различные группы пользователей имеют доступ к разным подмножествам базы данных и к ограниченному набору операций над ними. Для моделирования пользовательских (внешних по терминологии ANSI SPARC) представлений в модуле RDM используется механизм подсхем. Подсхема - это подмножество модели данных, доступное конкретному приложению или группе пользователей. SILVERRUN позволяет управлять "прозрачностью" границы между схемой и подсхемой, а также по требованию интерактивно переносить изменения из схемы в подсхему и наоборот. Число уровней подсхем не ограничено: можно создавать подсхемы подсхем.

**Генерация приложений и схем баз данных**

Спецификации схемы (или подсхемы) можно перенести через соответствующий мост в среду разработки приложений или сгенерировать схему базы данных для СУБД.

SILVERRUN не подменяет среду разработки и не содержит средств моделирования видов экранов или форматов отчетов. Эти функции специфичны для каждого языка разработки. SILVERRUN передает данные о форматах ввода, правилах редактирования, формах представления прямо в репозиторий среды разработки. Программисту остается расставить поля на экранах, подправить, если нужно, генерируемые большинством языков четвертого поколения автоматически SQL-запросы для этих экранов, определить систему меню - и основа работающего приложения создана.

Генерация схемы базы данных происходит путем создания файла с SQL-операторами, в которые переводятся конструкции модели данных. Этот файл используется для физического создания базы данных на сервере.

**Развитие системы**

SILVERRUN постоянно развивается. Регулярно выпускаются новые версии системы и постоянно обновляются мосты, что позволяет работать с самыми последними версиями поддерживаемых продуктов. Динамичность развития системы показывает, что SILVERRUN находится на передовом рубеже среди высокотехнологичных систем.

Дистрибутором методологии DATARUN и CASE-системы SILVERRUN в СНГ является фирма АРГУССОФТ. Она обеспечивает техническую поддержку и обучение пользователей.

# Заключение

Самым важным этом является проектирование информационной системы, поскольку именно от этого этапа зависит ее существования то, с чего, собственно, должна начинаться её жизнь. Например, для того, чтобы спроектировать сеть, нужно понять, какие задачи она будет решать, какими будут основные потоки трафика, как физически будут расположены пользователи и ресурсы, нужно ли задание приоритетов видов трафика, как будут решаться вопросы защиты информации внутри сети, как сеть будет подключена к Интернет, как решить задачи управления правами доступа пользователей.

Необходимо понять процесс создания ИС, что позволит гарантировать достижение целей разработки и соблюдение различных ограничений, например, бюджетных, временных и пр., а также привело к широкому использованию в этой сфере методов и средств программной инженерии: структурного анализа, объектно-ориентированного моделирования, CASE-систем. В работе рассмотрены цели создания ИС, задачи формирования требований к ИС, главная цель проектирования.

# Список литературы

1. Вендров А.М. Один из подходов к выбору средств проектирования баз данных и приложений. "СУБД", 1995, №3.
2. Зиндер Е.З. Бизнес-реинжиниринг и технологии системного проектирования. Учебное пособие. М., Центр Информационных Технологий, 1996.
3. Марка Д.А., МакГоуэн К. Методология структурного анализа и проектирования. М., "МетаТехнология", 1993.
4. Международные стандарты, поддерживающие жизненный цикл программных средств. М., МП "Экономика", 1996