

**ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ**  
**Федеральное государственное образовательное бюджетное**  
**учреждение высшего профессионального образования**  
**«САНКТ-ПЕТЕРБУРГСКИЙ**  
**ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ**  
**им. проф. М. А. БОНЧ-БРУЕВИЧА»**

---

**А.Н.Губин**

**Современные методы проектирования**  
**информационных систем**

**Краткий конспект лекций**

**Раздел 4. Типизация проектных решений.**

**САНКТ-ПЕТЕРБУРГ**

**2015**

## Содержание

### **Раздел 1. Общая характеристика процессов проектирования ИС.**

- 1.1. Информационные системы как объекты проектирования.*
- 1.2. Методологические основы проектирования ИС.*
- 1.3. Нормативная база проектирования ИС.*
- 1.4. Стадии проектирования ИС.*
- 1.5. Состав и содержание проектной документации.*

### **Раздел 2. Основные технологии проектирования ИС.**

- 2.1. Структурный подход к проектированию ИС.*
- 2.2. Особенности современных методов и средств проектирования ИС, основанных на CASE-технологии.*

### **Раздел 3. Основные принципы бездефектного проектирования ИС.**

- 3.1. Задачи моделирования ИС.*
- 3.2. Высокоуровневое моделирование ИС..*
- 3.3. Низкоуровневое моделирование средств реализации ИС.*
- 3.4. Имитационное моделирование ИС.*
- 3.5. Анализ и оценка производительности ИС.*

#### **Раздел 4. Типизация проектных решений.**

*4.1. Особенности технологии типового проектирования ИС.*

*4.2. Основные методы типового проектирования ИС.*

*4.3. RAD – технология проектирования.*

#### **Раздел 5. Управление проектами ИС.**

*5.1. Жизненный цикл ИС.*

*5.2. Расширение и обновление ИС.*

*5.3. Сопровождение, контроль эффективности и качества ИС.*

*5.4. Мониторинг безопасности ИС.*

*5.5. Перспективы и основные направления развития ИС и средств их проектирования.*

## Раздел 4. Типизация проектных решений.

### 4.1. Особенности технологии типового проектирования ИС

**Типовое проектирование ИС** предполагает создание системы из готовых типовых элементов. основополагающим требованием для применения методов *типового проектирования* является возможность декомпозиции проектируемой ИС на множество составляющих компонентов (подсистем, комплексов задач, программных модулей и т.д.). Для реализации выделенных компонентов выбираются имеющиеся  *типовые проектные решения*, которые настраиваются на особенности конкретного предприятия.

**Типовое проектное решение (ТПР)**- это тиражируемое (пригодное к многократному использованию) проектное решение.

Существующая классификация *ТПР* основана на уровне декомпозиции проектируемой ИС.

Можно выделить следующие классы *ТПР*:

- элементные *ТПР*- типовые решения по задаче или по отдельному виду обеспечения задачи (информационному, программному, техническому, математическому, организационному);
- подсистемные *ТПР*- в качестве элементов типизации выступают отдельные подсистемы, разработанные с учетом функциональной полноты и минимизации внешних информационных связей;
- объектные *ТПР*- типовые отраслевые проекты, которые включают полный набор функциональных и обеспечивающих подсистем ИС.

Основные особенности приведенных выше классов *ТПР* приведены в таблице 4.1.

Особенности классов ТПР

Таблица 4.1

Класс ТПР. Реализация ТПР	Достоинства	Недостатки
Элементные <i>ТПР</i> Библиотеки методо-ориентированных программ	Обеспечивается применение модульного подхода к проектированию и документированию ИС	1. Большие затраты времени на сопряжение разнородных элементов вследствие информационной, программной и технической несовместимости 2. Большие затраты времени на доработку <i>ТПР</i> отдельных элементов
Подсистемные <i>ТПР</i> Пакеты	3. достигается высокая	6. адаптивность ТПР

прикладных программ	<p>степень интеграции элементов ИС</p> <p>4. позволяют осуществлять: модульное проектирование; параметрическую настройку программных компонентов на различные объекты управления</p> <p>5. обеспечивают: сокращение затрат на проектирование и программирование взаимосвязанных компонентов; хорошее документирование отображаемых процессов обработки информации</p>	<p>недостаточна с позиции непрерывного инжиниринга деловых процессов</p> <p>7. возникают проблемы в комплексировании разных функциональных подсистем, особенно в случае использования решений нескольких производителей программного обеспечения</p>
Объектные ТПР Отраслевые проекты ИС	<p>8. комплексирование всех компонентов ИС за счет методологического единства и информационной, программной и технической совместимости</p> <p>9. открытость архитектуры — позволяет устанавливать ТПР на разных программно-технических платформах</p> <p>10. масштабируемость — допускает конфигурацию ИС для переменного числа рабочих мест</p> <p>11. конфигурируемость — позволяет выбрать необходимое подмножество компонентов</p>	<p>12. проблемы привязки типового проекта к конкретному объекту управления, что вызывает в некоторых случаях даже необходимость изменения организационно-экономической структуры объекта автоматизации</p>

#### 4.2. Основные методы типового проектирования ИС.

Для реализации *типового проектирования* используются два подхода - *параметрически-ориентированное* и *модельно-ориентированное проектирование*.

**Параметрически-ориентированное проектирование** включает следующие этапы: определение критериев оценки пригодности пакетов прикладных программ (ППП) для решения поставленных задач, анализ и оценка доступных ППП по сформулированным критериям, выбор и закупка

наиболее подходящего пакета, настройка параметров (доработка) закупленного ППП.

Критерии оценки ППП делятся на следующие группы:

- назначение и возможности пакета;
- отличительные признаки и свойства пакета;
- требования к техническим и программным средствам;
- документация пакета;
- факторы финансового порядка;
- особенности установки пакета;
- особенности эксплуатации пакета;
- помощь поставщика по внедрению и поддержанию пакета;
- оценка качества пакета и опыт его использования;
- перспективы развития пакета.

Внутри каждой группы критериев выделяется некоторое подмножество частных показателей, детализирующих каждый из десяти выделенных аспектов анализа выбираемых ППП.

Числовые значения показателей для конкретных ППП устанавливаются экспертами по выбранной шкале оценок (например, 10-балльной). На их основе формируются групповые оценки и комплексная оценка пакета (путем вычисления средневзвешенных значений). Нормированные взвешивающие коэффициенты также получают экспертным путем.

**Модельно-ориентированное проектирование** заключается в адаптации состава и характеристик типовой ИС в соответствии с моделью объекта автоматизации.

Технология проектирования в этом случае должна обеспечивать единые средства для работы как с моделью типовой ИС, так и с моделью конкретного предприятия.

Типовая ИС в специальной базе метаданных - репозитории - содержит модель объекта автоматизации, на основе которой осуществляется конфигурирование программного обеспечения. Таким образом, *модельно-ориентированное проектирование ИС* предполагает, прежде всего, построение модели объекта автоматизации с использованием специального программного инструментария (например, SAP Business Engineering Workbench (BEW), VAAN Enterprise Modeler). Возможно также создание системы на базе *типовой модели ИС* из репозитория, который поставляется вместе с программным продуктом и расширяется по мере накопления опыта проектирования информационных систем для различных отраслей и типов производства.

Репозиторий содержит *базовую (ссылочную) модель ИС, типовые (референтные) модели* определенных классов ИС, модели конкретных ИС предприятий.

**Базовая модель ИС** в репозитории содержит описание бизнес-функций, бизнес-процессов, бизнес-объектов, бизнес-правил, организационной структуры, которые поддерживаются программными модулями типовой ИС.

**Типовые модели** описывают конфигурации информационной системы для определенных отраслей или типов производства.

Модель конкретного предприятия строится либо путем выбора фрагментов основной или *типовой модели* в соответствии со специфическими особенностями предприятия (BAAN Enterprise Modeler), либо путем автоматизированной адаптации этих моделей в результате экспертного опроса (SAP Business Engineering Workbench).

Построенная модель предприятия в виде метаописания хранится в репозитории и при необходимости может быть откорректирована. На основе этой модели автоматически осуществляется конфигурирование и настройка информационной системы.

Бизнес-правила определяют условия корректности совместного применения различных компонентов ИС и используются для поддержания целостности создаваемой системы.

Модель бизнес-функций представляет собой иерархическую декомпозицию функциональной деятельности предприятия.

Модель бизнес-процессов отражает выполнение работ для функций самого нижнего уровня модели бизнес-функций. Для отображения процессов используется модель управления событиями (EPC - Event-driven Process Chain). Именно модель бизнес-процессов позволяет выполнить настройку программных модулей - приложений информационной системы в соответствии с характерными особенностями конкретного предприятия.

Модели бизнес-объектов используются для интеграции приложений, поддерживающих исполнение различных бизнес-процессов.

Модель организационной структуры предприятия представляет собой традиционную иерархическую структуру подчинения подразделений и персонала.

Внедрение типовой информационной системы начинается с анализа требований к конкретной ИС, которые выявляются на основе результатов предпроектного *обследования* объекта автоматизации. Для оценки соответствия этим требованиям программных продуктов может использоваться описанная выше методика оценки ППП. После выбора программного продукта на базе имеющихся в нем референтных моделей строится предварительная модель ИС, в которой отражаются все особенности реализации ИС для конкретного предприятия. Предварительная модель является основой для выбора *типовой модели* системы и определения перечня компонентов, которые будут реализованы с использованием других программных средств или потребуют разработки с помощью имеющихся в составе типовой ИС инструментальных средств (например, АВАР в SAP, Tools в BAAN).

Реализация типового проекта предусматривает выполнение следующих операций:

- установку глобальных параметров системы;

- задание структуры объекта автоматизации;
- определение структуры основных данных;
- задание перечня реализуемых функций и процессов;
- описание интерфейсов;
- описание отчетов;
- настройку авторизации доступа;
- настройку системы архивирования.

#### 4.3. RAD – технология проектирования.

Одним из условий обеспечения высокого качества создаваемых ИС является активное вовлечение конечных пользователей проектируемых ИС в процесс разработки предназначенных для них интерактивных систем, что нашло отражение в методологии прототипного проектирования.

Ядром этой методологии является быстрая разработка приложений RAD (Rapid Application Development).

При создании более сложных корпоративных ИС пользователям необходимо работать совместно с проектировщиками на протяжении всего периода разработки. Одним из путей повышения качества и эффективности создаваемых таким образом систем является применение технологии прототипного проектирования.

Данная технология обеспечивает создание на ранней стадии реализации действующей интерактивной модели системы, так называемой *системы-прототипа*, позволяющей наглядно продемонстрировать пользователю будущую систему, уточнить его требования, оперативно модифицировать интерфейсные элементы: формы ввода сообщений, меню, выходные документы, структуру диалога, состав реализуемых функций.

В процессе работы с системой-прототипом пользователь реально осознает возможности будущей системы и определяет наиболее удобный для него режим обработки данных, что значительно повышает качество создаваемых систем. Осуществляются проверка принципиальных проектных решений по составу и структуре ИС и оценка основных ее эксплуатационных характеристик.

Вовлечение пользователей в процесс проектирования и конструирования приложения позволяет получать замечания и дополнения к требованиям непосредственно в процессе проектирования приложения, сокращая время разработки. Представители заказчика получают возможность контролировать процесс создания системы и влиять на ее функциональное наполнение. Результатом является сдача в эксплуатацию системы, учитывающей большинство потребностей заказчиков.

Все приемы RAD служат одновременно для обеспечения высокого качества продукта и низкой стоимости разработки.



К числу этих приемов относятся:

- 1) разработка приложения итерациями;
- 2) необязательность полного завершения работ на каждом из этапов жизненного цикла для начала работ на следующем;
- 3) обязательное вовлечение пользователей в процесс проектирования и построения системы;
- 4) высокая параллельность работ;
- 5) повторное использование частей проекта;
- 6) необходимое применение CASE-средств, обеспечивающих техническую целостность на этапах анализа и проектирования;
- 7) применение средств управления конфигурациями, облегчающее внесение изменений в проект и сопровождение готовой системы;
- 8) использование автоматических генераторов (мастеров);
- 9) использование прототипирования, позволяющего полнее выяснить и удовлетворить потребности конечного пользователя;
- 10) тестирование и развитие проекта, осуществляемые одновременно с разработкой нескольких версий прототипа.

Каждое из перечисленных положений в отдельности способствует повышению скорости, улучшению качества, но только их совместное применение вызывает качественные изменения в процессе разработки.

Неполное завершение работ на каждом этапе позволяет переходить на следующий этап, не дожидаясь полного завершения работы на текущем этапе. При итеративном способе разработки ИС недостающую работу можно будет выполнить на следующей итерации. Главная же задача - как можно быстрее показать пользователям системы работоспособный продукт, тем самым активизируя процесс уточнения и дополнения требований.

Основная проблема процесса разработки ИС по RAD-технологии заключается в определении момента перехода на следующий этап. Для ее решения необходимо ввести временные ограничения на каждый из этапов жизненного цикла. Переход осуществляется в соответствии с планом, даже если не вся запланированная работа закончена. План составляется на основе статистических данных, полученных в предыдущих проектах, и личного опыта разработчиков с использованием инструментов автоматизации процесса планирования.

Средства автоматизации планирования являются важным элементом при разработке приложений по методологии RAD. Они применяются для определения состава и объема работ, необходимого числа разработчиков, распределения работ между участниками проекта, контроля за ходом выполнения работ и динамической корректировкой плана.

Для реализации технологии прототипного проектирования необходимо применять высокоуровневые инструментальные средства, которые позволяют

быстро преобразовать прототип системы в функционирующую версию и внести в нее в дальнейшем необходимые изменения.

Такие инструментальные средства можно условно разделить на два класса: инструменты быстрой разработки приложения в развитых СУБД - класс DEVELOPER и интегрированные инструменты быстрой разработки приложений - класс BUILDER.

К инструментам этих классов можно отнести средства 4GL (генераторы компонентов приложений):

- генераторы таблиц базы данных;
- генераторы форм ввода-вывода;
- генераторы запросов;
- генераторы отчетов;
- генераторы меню.

Такие генераторы существуют почти во всех СУБД, как персональных Access, FoxPro, Paradox, так и в окружении промышленных серверов БД (Oracle, Informix, Adabas D и др.).

Отличительной чертой класса BUILDER является то, что инструменты данного класса легко интегрируются с CASE-средствами и представляют собой единую среду быстрой разработки приложения. К интегрированным инструментам класса BUILDER можно отнести такие, как Power Builder Enterprise (Power Soft), Delphi (Borland), Builder Си ++ и др.

Жизненный цикл создания ИС на основе RAD-технологии предполагает после формирования технического задания и декомпозиции системы независимую разработку подсистем с последующей сборкой, тестированием и внедрением комплексной ИС (рис. 4.1).

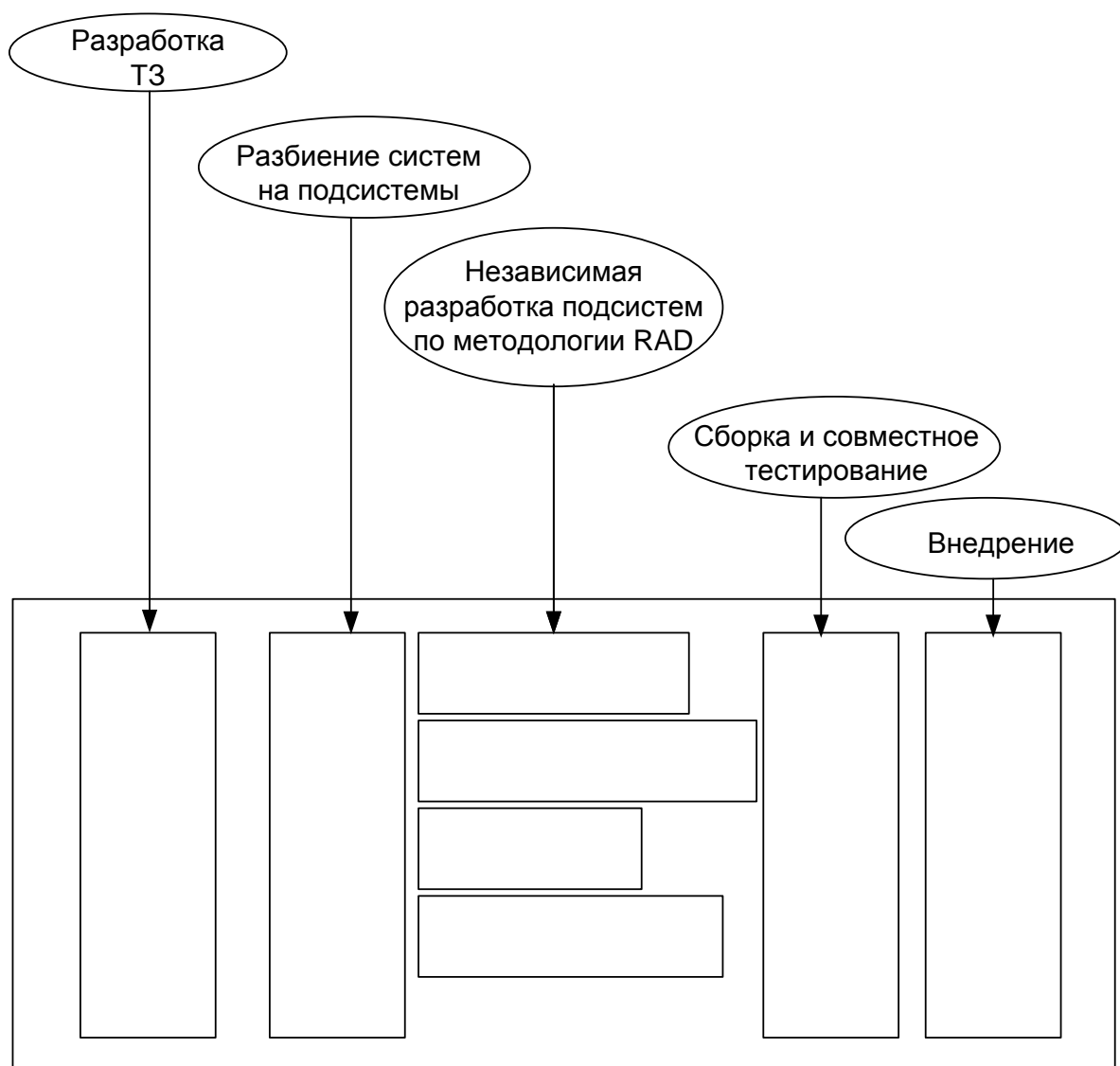


Рис. 4.1. Жизненный цикл создания ИС на основе RAD-технологии

Опыт использования RAD-технологии показывает, что существуют два базовых варианта организации технологического процесса проектирования с использованием систем-прототипов.

В первом варианте создание системы-прототипа используется для лучшей спецификации требований к разработке ИС, после разработки которых сам прототип оказывается ненужным. В этом случае традиционно разрабатывается этап «Постановка задачи», который является спецификацией системы-прототипа. После демонстрации пользователю и доработки прототипа разрабатывается новый вариант «Постановки задачи», который служит основой создания действующей ИС. Технологическая сеть проектирования (ТСП) данного варианта на стадии техно-рабочего проектирования ИС представлена на рис. 4.2.

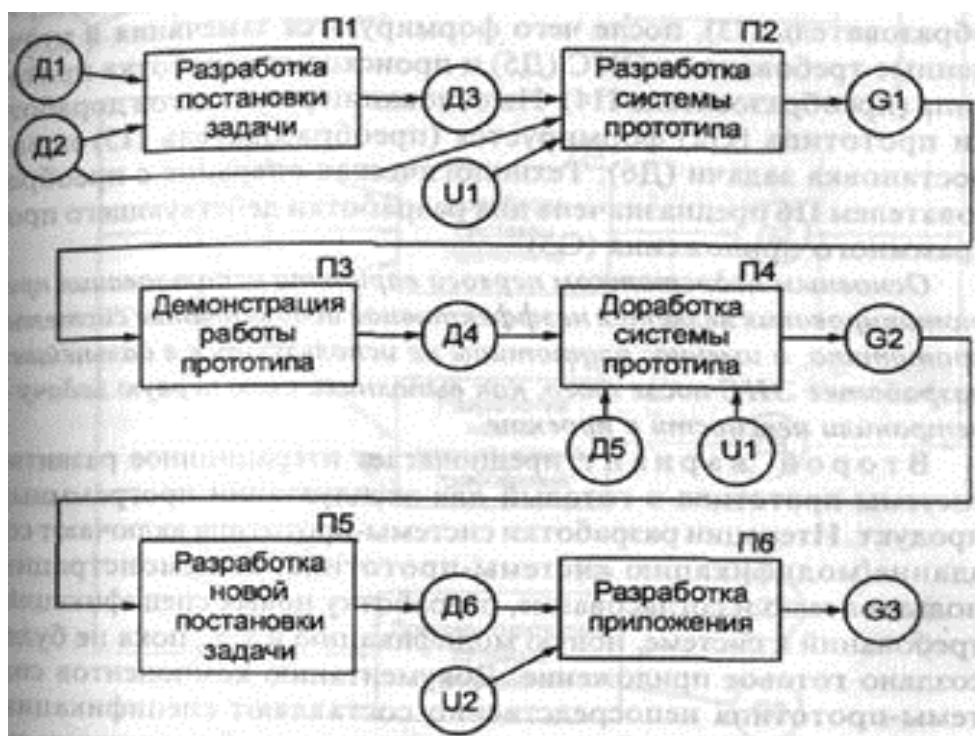


Рис.4.2. Технологическая сеть проектирования ИС с использованием блоков "Постановка задачи".

Д1 - техническое задание на разработку; Д2 - описание предметной области; Д3 - постановка задачи; U1 - универсум средств быстрой разработки приложений; G1 - приложение-прототип; Д4 - результаты работы приложения-прототипа; Д5 - замечания и уточненные требования к ЭИС; G2 - доработанный прототип; Д6 - новая постановка задачи; U2 - универсум средств разработки приложений; G3 - готовое приложение.

Основным недостатком этого варианта использования прототипирования является неэффективное использование системы-прототипа, а именно: прототипы не используются в дальнейшей разработке ИС после того, как выполнили свою первую задачу - устранили неясности в проекте.

Второй вариант предполагает итерационное развитие системы-прототипа в готовый для эксплуатации программный продукт. Итерации разработки системы-прототипа включают создание /модификацию системы-прототипа, ее демонстрацию пользователю и согласование, разработку новых спецификаций-требований к системе, новую модификацию и т.д., пока не будет создано готовое приложение.

Документацию компонентов системы-прототипа непосредственно составляют спецификации, которые являются требованиями к программной

реализации системы и определяют характер взаимоотношений с заказчиком на этапе сдачи готовой системы. Технологическая сеть проектирования данного варианта на стадии техно-рабочего проектирования ИС представлена на рис. 4.3.

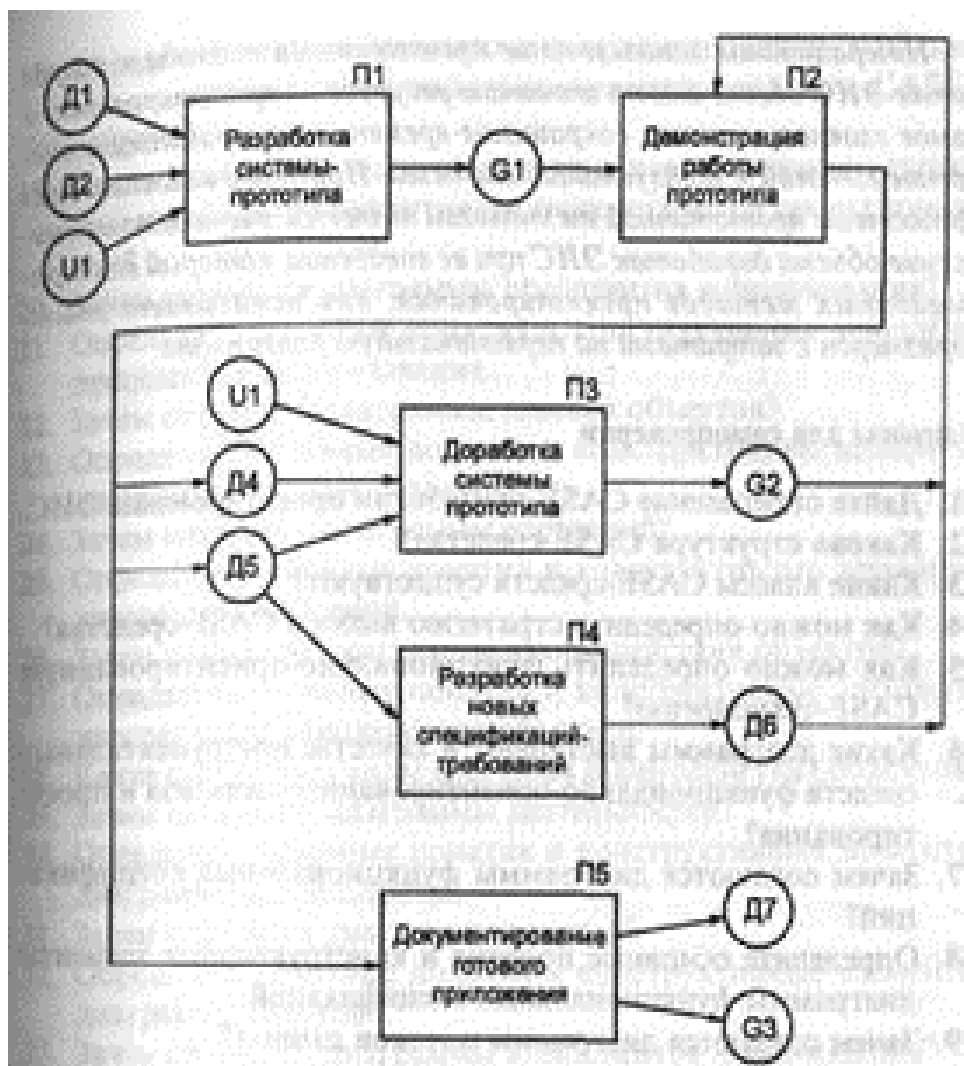


Рис.4.3. Технологическая сеть проектирования ИС с итерационным использованием системы-прототипа ИС.

Д1 - техническое Задание на разработку; Д2 - описания предметной области; U1 - универсум средств быстрой разработки приложений; G1 - приложение-прототип; Д4 - результаты работы приложения-прототипа; Д5 - замечания и уточненные требования к ЭИС; G2 - доработанный прототип; Д6 - новые спецификации-требования; G3 - готовое приложение, Д7 – готовая документация.

Итерационное использование прототипного подхода к разработке ИС обеспечивает экономию ресурсов на проектирование, а самое главное, - резкое

сокращение времени на разработку и внедрение готовой к эксплуатации системы. При этом основным достоинством прототипной технологии является значительное снижение объема доработок ИС при ее внедрении, который для традиционных методов проектирования, как показывает опыт, соразмерен с затратами на первоначальную реализацию.

