



Московский авиационный институт
(национальный исследовательский университет)



Факультет №8 Прикладная математика и физика

Направление подготовки: Прикладная математика и информатика

Специализация подготовки: Информатика

Дисциплина: **ПРОЕКТИРОВАНИЕ БАЗ
ДААННЫХ**

СЕРПУХОВ 2016



Информационные ресурсы представляют собой один из обязательных элементов, необходимых для осуществления любого вида человеческой деятельности: производства, управления, научных исследований, проектирования новой техники и технологии, подготовки и переподготовки кадров и т.д..

Информационная система (ИС) в целом - автоматизированная система, предназначенная для организации, хранения, пополнения, поддержки и представления пользователям информации в соответствии с их запросами.



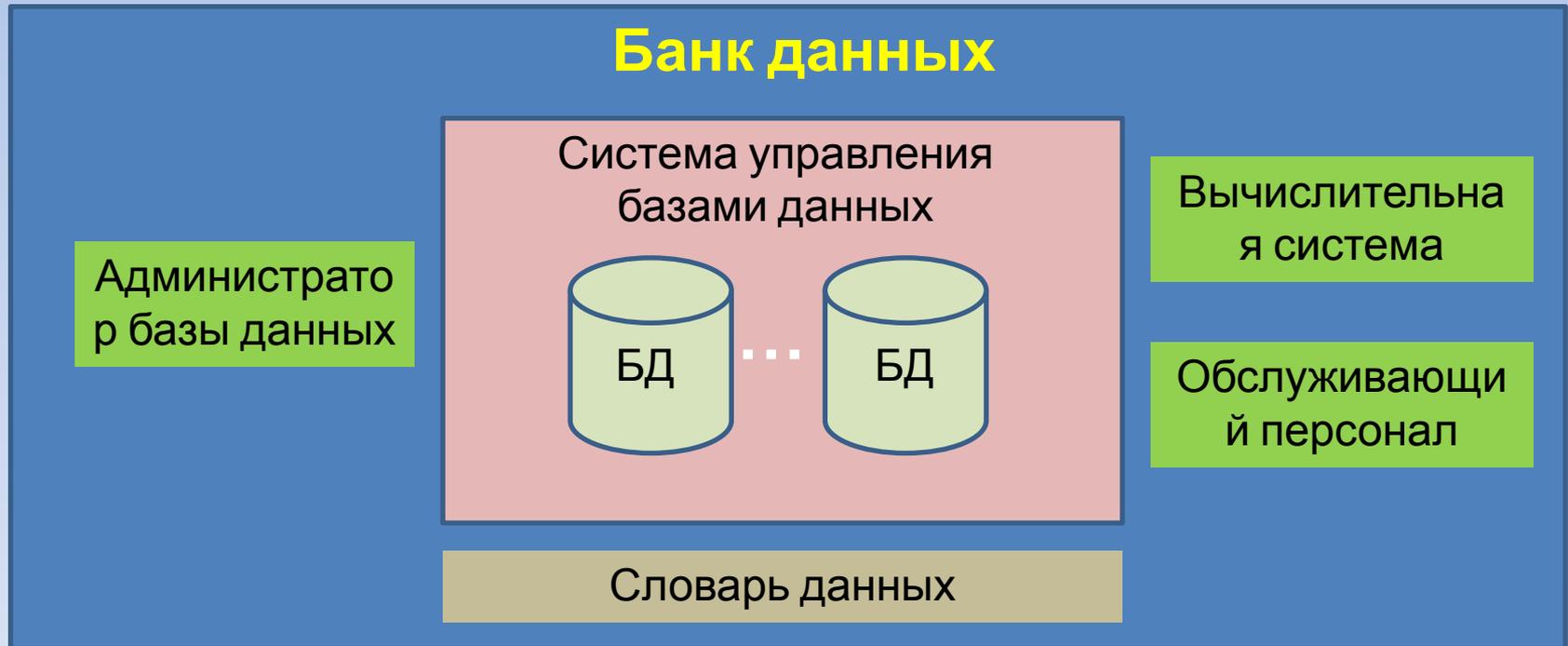
Укрупненная функциональная схема информационной системы



В большинстве случаев информационные системы реализуются в виде банков данных (БНД)



БНД – разновидность ИС, в которой реализованы функции централизованного хранения и накопления обрабатываемой информации, организованной в одну или несколько баз данных





База данных (БД) – совокупность специальным образом организованных данных, хранимых в памяти вычислительной системы и отображающих состояние объектов и их взаимосвязей в рассматриваемой предметной области.

Система управления базами данных (СУБД) – это комплекс языковых и программных средств, предназначенный для создания, ведения и совместного использования БД многими пользователями.

Словарь данных – подсистема БД, предназначенная для централизованного хранения информации о структуре данных, взаимосвязях файлов БД друг с другом, типах данных и форматах их представления, принадлежности данных пользователям, кодах защиты и разграничения доступа и т.д.

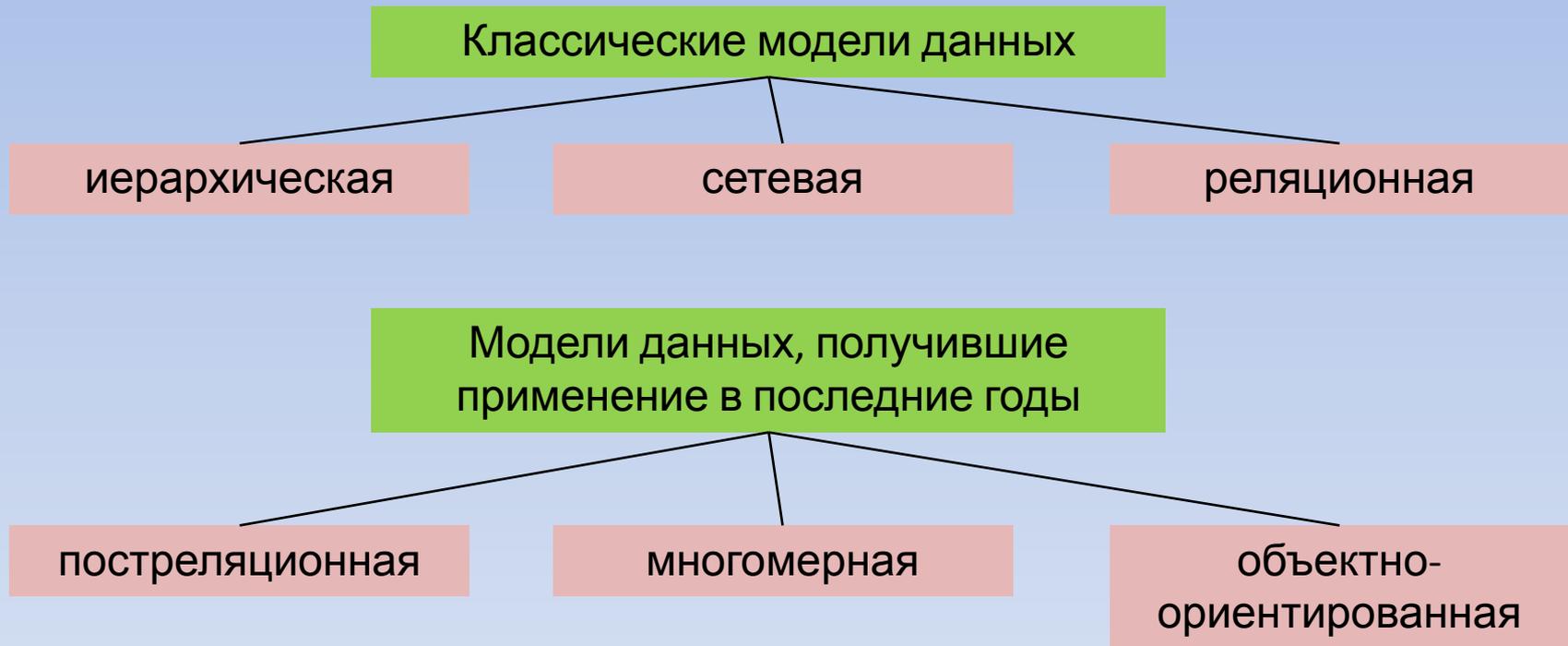
Администратор баз данных – лицо или группа лиц, ответственных за выработку требований к БД, её проектирование, создание, эффективное использование и сопровождение.

Вычислительная система – совокупность взаимосвязанных и согласованно действующих ЭВМ или процессоров и других устройств, обеспечивающих автоматизацию процесса приёма, обработки и выдачи информации потребителям.

Обслуживающий персонал выполняет функции поддержания технических и программных средств в работоспособном состоянии.



Хранимые в базе данные имеют определённую логическую структуру – иными словами, описываются некоторой **моделью представления данных** (моделью данных), поддерживаемой СУБД.





Лекция 1 Основные понятия технологии проектирования информационных систем (ИС)



Учебные вопросы:

- 1.1 Технологии проектирования информационных систем
- 1.2 Организация разработки ИС

Литература:

1. Хоменко А.Д., Цыганков И.М., Мальцев М.Г. Базы данных. Учебник для высших учебных заведений / Под ред. Проф. А.А. Хоменко. – 4-е изд. доп. и перераб. – СПб: КОРОНА принт, 2004.
2. Швецов В.И., Визгунов А.Н., Мееров И.Б. Базы данных. Учебное пособие. Нижний Новгород: Изд-во ННГУ, 2004.



1.1 Технологии проектирования информационных систем



Индустрия разработки автоматизированных информационных систем зародилась в 1950-х - 1960-х годах и к концу века приобрела вполне законченные формы

Этапы развития подходов к разработке информационных систем:

I этап:

На данном этапе основным подходом в проектировании ИС был метод «снизу-вверх»:

- система создавалась как набор приложений, наиболее важных в данный момент для поддержки деятельности предприятия. Основной целью этих проектов было не создание тиражируемых продуктов, а обслуживание текущих потребностей конкретного учреждения. Такой подход отчасти сохраняется и сегодня. В рамках «Лоскутной автоматизации» достаточно хорошо обеспечивается поддержка отдельных функций, но практически полностью отсутствует стратегия развития комплексной системы автоматизации, а объединение функциональных подсистем превращается в самостоятельную и достаточно сложную проблему.



Достоинство:

достаточно хорошо обеспечивается поддержка отдельных функций

Недостатки:

сложности, связанные с разными представлениями пользователей об одних и тех же данных, приводили к непрерывным доработкам программных продуктов для удовлетворения все новых и новых пожеланий отдельных работников. Как следствие - и работа программистов, и создаваемые ИС вызывали недовольство руководителей и пользователей системы.



Этапы развития подходов к разработке информационных систем:

II этап:

Следующий этап связан с осознанием того факта, что существует потребность в достаточно стандартных программных средствах автоматизации деятельности различных учреждений и предприятий. Из всего спектра проблем разработчики выделили наиболее заметные: автоматизацию ведения бухгалтерского аналитического учета и технологических процессов. Системы начали проектироваться «сверху-вниз», т.е. в предположении, что одна программа должна удовлетворять потребности многих пользователей.



Достоинство:

универсальность данных

Недостатки:

заложенные «сверху» жесткие рамки не дают возможности гибко адаптировать систему к специфике деятельности конкретного предприятия: учесть необходимую глубину аналитического и производственно-технологического учета, включить необходимые процедуры обработки данных, обеспечить интерфейс каждого рабочего места с учетом функций и технологии работы конкретного пользователя. Решение этих задач требует серьезных доработок системы. Таким образом, материальные и временные затраты на внедрение системы и ее доводку под требования заказчика обычно значительно превышают запланированные показатели.



Этапы развития подходов к разработке информационных систем:

III этап:

Таким образом, возникла насущная необходимость формирования новой методологии построения информационных систем.

Цель такой методологии заключается в регламентации процесса проектирования ИС и обеспечении управления этим процессом с тем, чтобы гарантировать выполнение требований как к самой ИС, так и к характеристикам процесса разработки.



Основные задачи, решению которых должна способствовать методология проектирования корпоративных ИС:

- обеспечивать создание корпоративных ИС, отвечающих целям и задачам организации, а также предъявляемым требованиям по автоматизации деловых процессов заказчика;
- гарантировать создание системы с заданным качеством в заданные сроки и в рамках установленного бюджета проекта;
- поддерживать удобную дисциплину сопровождения, модификации и наращивания системы;
- обеспечивать преемственность разработки, т.е. использование в разрабатываемой ИС существующей информационной инфраструктуры организации (задела в области информационных технологий).

Внедрение методологии должно приводить к снижению сложности процесса создания ИС за счет полного и точного описания этого процесса, а также применения современных методов и технологий создания ИС на всем жизненном цикле ИС - от замысла до реализации.



Проектирование ИС охватывает три основные области:

- проектирование объектов данных, которые будут реализованы в базе данных;
- проектирование программ, экранных форм, отчётов, которые будут обеспечивать выполнение запросов к данным;
- учёт конкретной среды или технологии, а именно: топологии сети; конфигурации аппаратных средств; используемой архитектуры (файл-сервер или клиент-сервер); параллельной или распределённой обработки данных и т.д.



Проектирование ИС всегда начинается с определений цели проекта.

В общем виде цель проекта можно определить как решение ряда взаимосвязанных задач, включавших 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.



Процесс создания ИС делится на ряд этапов (стадий), ограниченных некоторыми временными рамками и заканчивающихся выпуском конкретного продукта (моделей, программных продуктов, документации и пр.).

Обычно выделяют следующие этапы создания ИС:

- формирование требований к системе;
- проектирование;
- реализация;
- тестирование;
- ввод в эксплуатацию;
- эксплуатация и сопровождение.

Совокупность данных этапов называется **жизненным циклом ИС**



В настоящее время известны и используются следующие модели жизненного цикла:

- **Каскадная модель** предусматривает последовательное выполнение всех этапов проекта в строго фиксированном порядке. Переход на следующий этап означает полное завершение работ на предыдущем этапе.
- **Поэтапная модель с промежуточным контролем.** Разработка ИС ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют учитывать реально существующее взаимовлияние результатов разработки на различных этапах; время жизни каждого из этапов растягивается на весь период разработки.
- **Спиральная модель.** На каждом витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество и планируются работы следующего витка.



Каскадная модель ЖЦ ИС





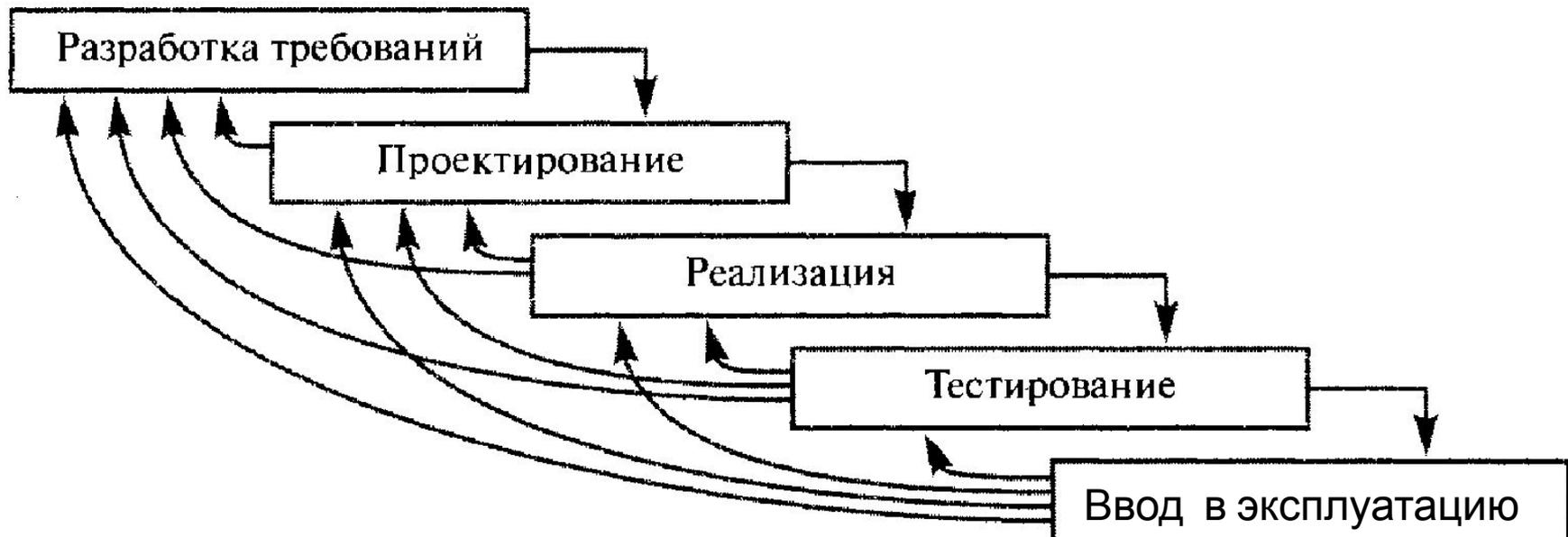
Можно выделить следующие положительные стороны применения

каскадного подхода:

- на каждом этапе формируется законченный набор проектной документации, отвечающий критериям полноты и согласованности;
- выполняемые в логической последовательности этапы работ позволяют планировать сроки завершения всех работ и соответствующие затраты.

Основным недостатком этого подхода является то, что реальный процесс создания системы никогда полностью не укладывается в такую жесткую схему, постоянно возникает потребность в возврате к предыдущим этапам и уточнении или пересмотре ранее принятых решений.

Поэтапная модель с промежуточным контролем





Однако и эта схема не позволяет оперативно учитывать возникающие изменения и уточнения требований к системе. Согласование результатов разработки с пользователями производится только в точках, планируемых после завершения каждого этапа работ, а общие требования к ИС зафиксированы в виде технического задания на все время ее создания. Таким образом, пользователи зачастую получают систему, не удовлетворяющую их реальным потребностям.



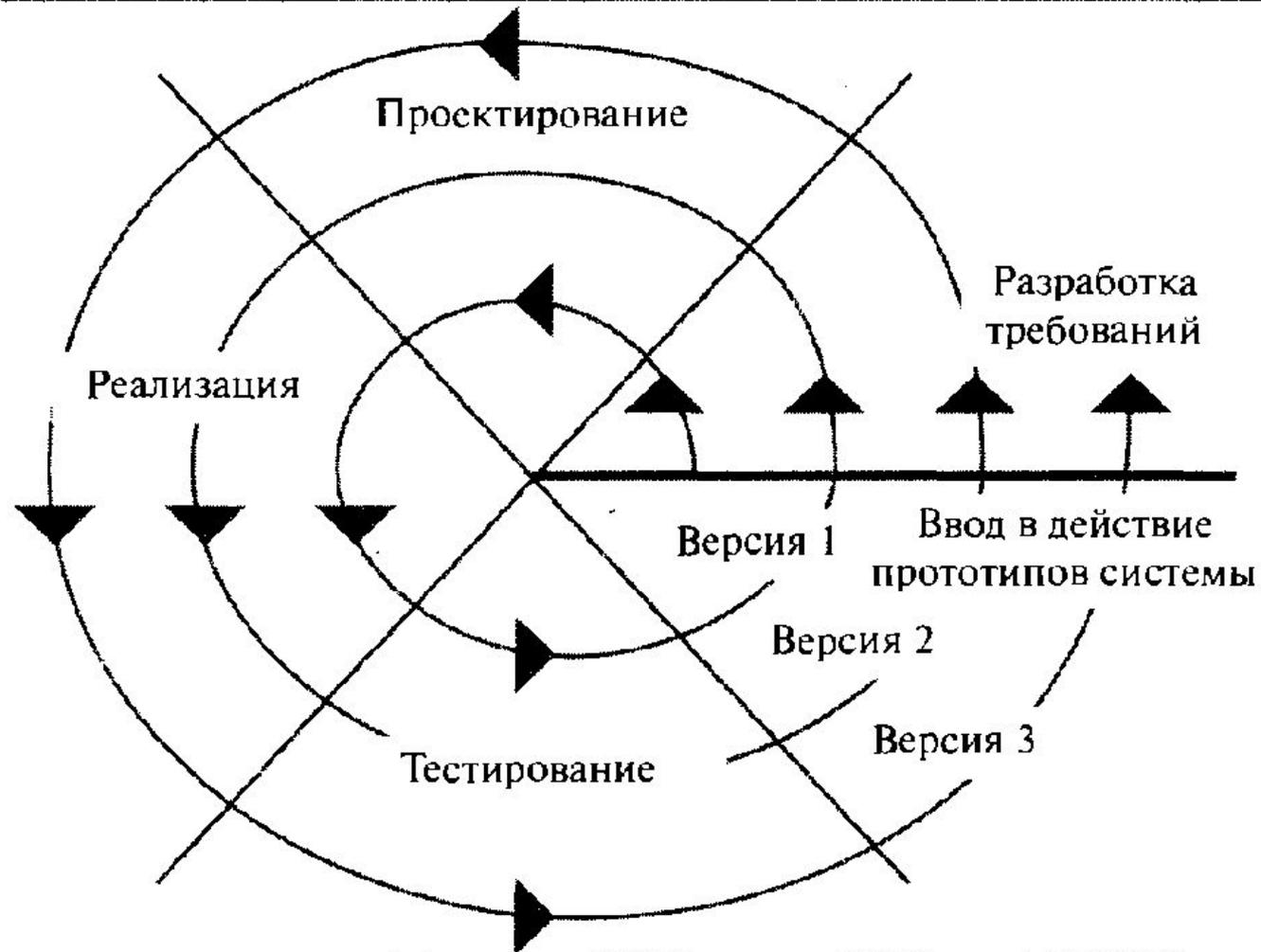
Спиральная модель ЖЦ была предложена для преодоления перечисленных проблем.

На этапах анализа и проектирования реализуемость технических решений и степень удовлетворения потребностей заказчика проверяется путем создания прототипов.

Каждый виток спирали соответствует созданию работоспособного фрагмента или версии системы. Это позволяет уточнить требования, цели и характеристики проекта, определить качество разработки, спланировать работы следующего витка спирали.

Таким образом углубляются и последовательно конкретизируются детали проекта и в результате выбирается обоснованный вариант, который удовлетворяет действительным требованиям заказчика и доводится до реализации.

Спиральная модель ЖЦ ИС





Основная проблема спирального цикла – определение момента перехода на следующий этап.

Для ее решения вводятся временные ограничения на каждый из этапов жизненного цикла, и переход осуществляется в соответствии с планом, даже если не вся запланированная работа закончена. Планирование производится на основе статистических данных, полученных в предыдущих проектах, и личного опыта разработчиков.

Существует целый ряд стандартов, регламентирующих ЖЦ ПО, а в некоторых случаях и процессы разработки.

Но практически все они в своей основе имеют методологию, разработанную фирмой IBM в середине 70-х годов BSP (Business System Planning - методология организационного планирования).



Среди наиболее известных стандартов можно выделить следующие:

- **ГОСТ 34.601-90** - распространяется на автоматизированные системы и устанавливает стадии и этапы их создания. Кроме того, в стандарте содержится описание содержания работ на каждом этапе. Стадии и этапы работы, закрепленные в стандарте, в большей степени соответствуют каскадной модели жизненного цикла.
- **ISO/IEC 12207:1995** - стандарт на процессы и организацию жизненного цикла. Распространяется на все виды заказного ПО. Стандарт не содержит описания фаз, стадий и этапов.
- **Custom Development Method** (методика Oracle) по разработке прикладных информационных систем - технологический материал, детализированный до уровня заготовок проектных документов, рассчитанных на использование в проектах с применением Oracle. Применяется СОМ для классической модели ЖЦ (предусмотрены все работы/задачи и этапы), а также для технологий «быстрой разработки» (Fast Track) или «облегченного подхода», рекомендуемых в случае малых проектов.



- **Rational Unified Process (RUP)** предлагает итеративную модель разработки, включавшую четыре фазы: начало, исследование, построение и внедрение. Каждая фаза может быть разбита на этапы (итерации), в результате которых выпускается версия для внутреннего или внешнего использования. Прохождение через четыре основные фазы называется циклом разработки, каждый цикл завершается генерацией версии системы. Если после этого работа над проектом не прекращается, то полученный продукт продолжает развиваться и снова минует те же фазы. Суть работы в рамках RUP - это создание и сопровождение моделей на базе UML (Unified Modeling Language — унифицированный язык моделирования) .
- **Microsoft Solution Framework (MSF)** сходна с RUP, так же включает четыре фазы: анализ, проектирование, разработка, стабилизация, является итерационной, предполагает использование объектно-ориентированного моделирования. MSF в сравнении с RUP в большей степени ориентирована на разработку бизнес-приложений.
- **Extreme Programming (XP)**. Экстремальное программирование (самая новая среди рассматриваемых методологий) сформировалось в 1996 году. В основе методологии командная работа, эффективная коммуникация между заказчиком и исполнителем в течение всего проекта по разработке ИС, а разработка ведется с использованием последовательно дорабатываемых прототипов.



1.2 Организация разработки ИС



Каноническое проектирование ИС

Организация канонического проектирования ИС ориентирована на использование главным образом каскадной модели жизненного цикла ИС. Стадии и этапы работы описаны в стандарте ГОСТ 34.601-90.

Стадии и этапы создания ИС, выполняемые организациями-участниками, прописываются в договорах и технических заданиях на выполнение работ:

Стадия 1. Формирование требований к ИС

- обследование объекта и обоснование необходимости создания ИС;
- формирование требований пользователей к ИС;
- оформление отчета о выполненной работе и тактика-технического задания на разработку.

Стадия 2. Разработка концепции ИС

- изучение объекта автоматизации;
- проведение необходимых научно-исследовательских работ;
- разработка вариантов концепции ИС, удовлетворяющих требованиям пользователей;
- оформление отчета и утверждение концепции.



Стадия 3. Техническое задание

- разработка и утверждение технического задания на создание ИС.

Стадия 4. Эскизный проект

- разработка предварительных проектных решений по системе и ее частям;
- разработка эскизной документации на ИС и ее части.

Стадия 5. Технический проект

- разработка проектных решений по системе и ее частям;
- разработка документации на ИС и ее части;
- разработка и оформление документации на поставку комплектующих изделий;
- разработка заданий на проектирование в смежных частях проекта.

Стадия 6. Рабочая документация

- разработка рабочей документации на ИС и ее части;
- разработка и адаптация программ.



Стадия 7. Ввод в действие

- подготовка объекта автоматизации;
- подготовка персонала;
- комплектация ИС поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями);
- строительско-монтажные работы;
- пусконаладочные работы;
- проведение предварительных испытаний;
- проведение опытной эксплуатации;
- проведение приемочных испытаний.

Стадия 8. Сопровождение ИС

- выполнение работ в соответствии с гарантийными обязательствами;
- послегарантийное обслуживание.



Типовое проектирование ИС

Типовое проектирование ИС предполагает создание системы из готовых типовых элементов. Основопологающим 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 на рынке типовые проектные решения, которые настраиваются на особенности конкретного предприятия.

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

Выделяются 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.



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.



Спасибо за внимание!