

# Тема №4

## Жизненный цикл программного обеспечения ИС

# Что такое жизненный цикл ИС?

**Жизненный цикл (ЖЦ) ИС** - некоторую последовательность стадий и выполняемых на них процессов.

*Для каждого этапа определяются состав и последовательность выполняемых работ, получаемые результаты, методы и средства, необходимые для выполнения работ, роли и ответственность участников и т.д.*

Жизненный цикл ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования.

# Модель жизненного цикла

Модель жизненного цикла отражает различные состояния системы, начиная с момента возникновения необходимости в данной ИС и заканчивая моментом ее полного выхода из употребления.

**Модель жизненного цикла** - структура, содержащая процессы, действия и задачи, которые осуществляются в ходе разработки, функционирования и сопровождения программного продукта в течение всей жизни системы, от определения требований до завершения ее использования.

# Виды моделей жизненного цикла

В настоящее время известны и используются следующие модели жизненного цикла:

- Каскадная модель
- Поэтапная модель с промежуточным контролем
- Спиральная модель

# Каскадная модель



- 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

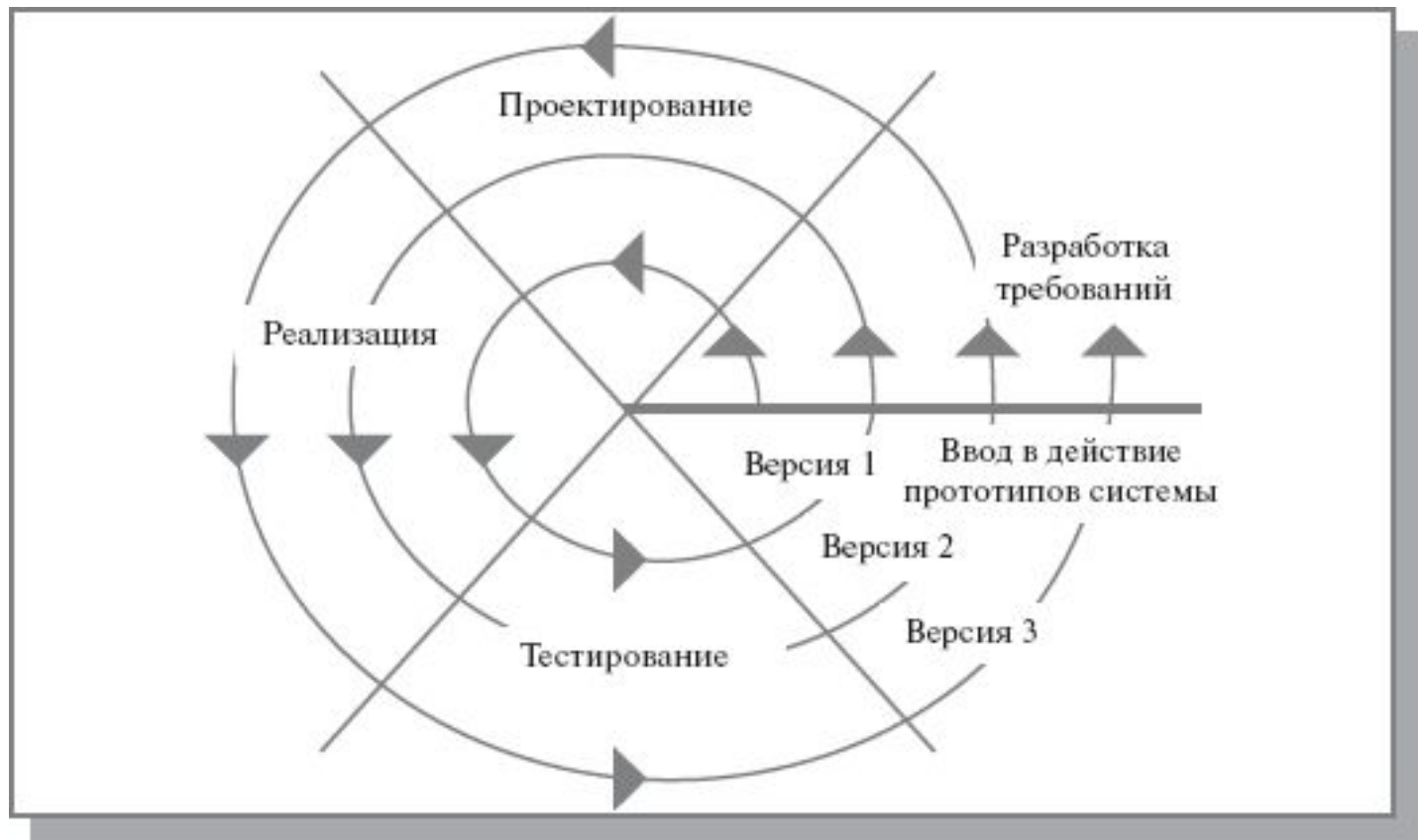


- Разработка ИС ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют учитывать реально существующее взаимовлияние результатов разработки на различных этапах; время жизни каждого из этапов растягивается на весь период разработки.

# Спиральная модель

- На каждой витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество и планируются работы следующего витка.
- **Особое внимание уделяется начальным этапам разработки** - анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов (макетирования).

# Спиральная модель





# Исторические особенности

На практике наибольшее распространение получили две основные модели жизненного цикла: **каскадная модель** (характерна для периода **1970-1985 гг.**); **спиральная модель** (характерна для периода **после 1986.г.**).

В ранних проектах достаточно простых ИС каждое приложение представляло собой единый, функционально и информационно независимый блок. Для разработки такого типа приложений эффективным оказался каскадный способ.

# Плюсы каскадного подхода

Можно выделить следующие положительные стороны применения каскадного подхода:

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

*Каскадный подход хорошо зарекомендовал себя при построении относительно простых ИС, когда в самом начале разработки можно достаточно точно и полно сформулировать все требования к системе.*

# Минусы каскадного подхода

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.

**В результате реальный процесс создания ИС оказывается соответствующим поэтапной модели с промежуточным контролем.**

# Необходимость учета оперативных изменений

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.

Следовательно: пользователи зачастую получают систему, не удовлетворяющую их реальным потребностям.

# Спиральная модель – как средство решения проблем

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.*

# Основная проблема спирального цикла

**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.

# Первая причина популярности каскадной модели

1. **Привычка** - многие ИТ-специалисты получали образование в то время, когда изучалась только каскадная модель, поэтому она используется ими и в наши дни.

# Вторая причина популярности каскадной модели

**Иллюзия снижения рисков** участников проекта (заказчика и исполнителя). Каскадная модель предполагает разработку законченных продуктов на каждом этапе: технического задания, технического проекта, программного продукта и пользовательской документации.

Разработанная документация позволяет не только определить требования к продукту следующего этапа, но и определить обязанности сторон, объем работ и сроки.



# Основа иллюзии

Если требования к информационной системе меняются в ходе реализации проекта, а качество документов оказывается невысоким (требования неполны и/или противоречивы), то в действительности использование каскадной модели создает лишь иллюзию определенности и на деле увеличивает риски, уменьшая лишь ответственность участников проекта.

*При формальном подходе менеджер проекта реализует только те требования, которые содержатся в спецификации, опирается на документ, а не на реальные потребности бизнеса.*

# Типы контрактов

Есть два основных типа контрактов на разработку ИС:

- первый тип предполагает выполнение определенного объема работ за определенную сумму в определенные сроки (fixed price);
- второй тип предполагает повременную оплату работы (time work).

*Выбор того или иного типа контракта зависит от степени определенности задачи. Каскадная модель с определенными этапами и их результатами лучше приспособлена для заключения контракта с оплатой по результатам работы.*

# Контракт с повременной оплатой

**Целесообразно заключение контракта с повременной оплатой на небольшую систему, с относительно небольшим весом в структуре затрат предприятия.**

# Внедрение интегрированной ИС

Разработка и внедрение интегрированной информационной системы требует существенных финансовых затрат, поэтому *используются контракты с фиксированной ценой и каскадная модель разработки и внедрения.*

Спиральная модель чаще применяется при разработке информационной системы силами собственного отдела ИТ предприятия.

# Третья причина популярности каскадной модели

**Проблемы внедрения** при использовании итерационной модели. В некоторых областях спиральная модель не может применяться, поскольку невозможно тестирование продукта, обладающего неполной функциональностью (например, военные разработки, атомная энергетика и т.д.).

Поэтапное итерационное внедрение информационной системы для бизнеса возможно, но сопряжено с организационными сложностями (перенос данных, интеграция систем, изменение бизнес-процессов, учетной политики).

# Понятие процесса жизненного цикла проектирования

Каждая из стадий создания системы предусматривает выполнение определенного объема работ, которые представляются в виде процессов ЖЦ.

*Процесс определяется как совокупность взаимосвязанных действий, преобразующих входные данные в выходные.*

Описание каждого процесса включает в себя перечень решаемых задач, исходных данных и результатов.

# Методология Business System Planning

Компания IBM предложила в середине 1970-х годов методологию **BSP** (Business System Planning - **методология организационного планирования**).

Метод структурирования информации с использованием *матриц пересечения бизнес-процессов, функциональных подразделений, функций систем обработки данных (информационных систем), информационных объектов, документов и баз данных*, предложенный в BSP, используется сегодня не только в ИТ-проектах, но и проектах по реинжинирингу бизнес-процессов, изменению организационной структуры.

# Стандарты, регламентирующие жизненный цикл ИС

- **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.



# Стандарты, регламентирующие жизненный цикл ИС

- **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.

# Стандарты, регламентирующие жизненный цикл ИС

- **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 CDM 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.

# Стандарты, регламентирующие жизненный цикл ИС

- **Rational Unified Process (RUP)** предлагает итеративную модель разработки, включающую 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.
- Суть работы в рамках RUP - это создание и сопровождение моделей на базе UML.

# Стандарты, регламентирующие жизненный цикл ИС

- **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.

# Стандарты, регламентирующие жизненный цикл ИС

- **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.

# Группы процессов жизненного цикла ИС (ISO/IEC 12207 )

## **Основные процессы:**

приобретение;  
поставка;  
разработка;  
эксплуатация;  
сопровождение.

## **Организационные**

**процессы:**  
создание  
инфраструктуры;  
управление;  
обучение;  
усовершенствование.

## **Вспомогательные процессы:**

документирование;  
управление  
конфигурацией;  
обеспечение  
качества;  
разрешение  
проблем;  
аудит;  
аттестация;  
совместная оценка;  
верификация.

# Содержание основных процессов ЖЦ (ISO/IEC 12207)

Процесс (исполнитель процесса)	Действия	Вход	Результат
Приобретение (заказчик)	Инициирование •Подготовка заявочных предложений •Подготовка договора •Контроль деятельности поставщика •Приемка ИС	Решение о начале работ по внедрению ИС •Результаты обследования деятельности заказчика •Результаты анализа рынка ИС/ тендера •План поставки/ разработки •Комплексный тест ИС	Технико-экономическое обоснование внедрения ИС •Техническое задание на ИС •Договор на поставку/ разработку •Акты приемки этапов работы •Акт приемно-сдаточных испытаний

# Содержание основных процессов ЖЦ (ISO/IEC 12207)

Процесс (исполнитель процесса)	Действия	Вход	Результат
Поставка (разработчик ИС)	Инициирование • Ответ на заявочные предложения • Подготовка договора • Планирование исполнения • Поставка ИС	Техническое задание на ИС • Решение руководства об участии в разработке • Результаты тендера • Техническое задание на ИС • План управления проектом • Разработанная ИС и документация	Решение об участии в разработке • Коммерческие предложения/ конкурсная заявка • Договор на поставку/ разработку • План управления проектом • Реализация/ корректировка • Акт приемно-сдаточных испытаний



# Содержание основных процессов ЖЦ (ISO/IEC 12207)

Процесс (исполнитель процесса)	Действия	Вход	Результат
Разработка (разработчик ИС)	<ul style="list-style-type: none"> <li>•Подготовка</li> <li>•Анализ требований к ИС</li> <li>•Проектирование архитектуры ИС</li> <li>•Разработка требований к ПО</li> <li>•Проектирование архитектуры ПО</li> <li>•Детальное проектирование ПО</li> <li>•Кодирование и тестирование ПО</li> <li>•Интеграция ПО и квалификационное тестирование ПО</li> <li>•Интеграция ИС и квалификационное тестирование ИС</li> </ul>	<ul style="list-style-type: none"> <li>•Техническое задание на ИС</li> <li>•Техническое задание на ИС, модель ЖЦ</li> <li>•Техническое задание на ИС</li> <li>•Подсистемы ИС</li> <li>•Спецификации требования к компонентам ПО</li> <li>•Архитектура ПО</li> <li>•Материалы детального проектирования ПО</li> <li>•План интеграции ПО, тесты</li> <li>•Архитектура ИС, ПО, документация на ИС, тесты</li> </ul>	<ul style="list-style-type: none"> <li>•Используемая модель ЖЦ, стандарты разработки</li> <li>•План работ</li> <li>•Состав подсистем, компоненты оборудования</li> <li>•Спецификации требования к компонентам ПО</li> <li>•Состав компонентов ПО, интерфейсы с БД, план интеграции ПО</li> <li>•Проект БД, спецификации интерфейсов между компонентами ПО, требования к тестам</li> <li>•Тексты модулей ПО, акты автономного тестирования</li> <li>•Оценка соответствия комплекса ПО требованиям ТЗ</li> <li>•Оценка соответствия ПО, БД, технического комплекса и комплекта документации требованиям ТЗ</li> </ul>

# Стандарт на процессы (ISO/IEC 15288)

В 2002 г. опубликован стандарт на процессы жизненного цикла систем (ISO/IEC 15288 System life cycle processes).

Был учтен практический опыт создания систем в правительственных, коммерческих, военных и академических организациях.

*Стандарт применим для широкого класса систем, но его основное предназначение - поддержка создания компьютеризированных систем.*

# Группы процессов жизненного цикла ИС (ISO/IEC 15288)

## **Договорные процессы:**

приобретение (внутренние решения или решения внешнего поставщика);  
поставка (внутренние решения или решения внешнего поставщика).

## **Процессы предприятия:**

управление окружающей средой предприятия;  
инвестиционное управление;  
управление ЖЦ ИС;  
управление ресурсами;  
управление качеством.

# Группы процессов жизненного цикла ИС (ISO/IEC 15288)

## **Проектные процессы:**

планирование проекта;

оценка проекта;

контроль проекта;

управление рисками;

управление конфигурацией;

управление информационными потоками;

принятие решений.

# Группы процессов жизненного цикла ИС (ISO/IEC 15288)

## **Технические процессы:**

определение требований; анализ требований;  
разработка архитектуры; внедрение;  
интеграция; верификация;  
переход; аттестация;  
эксплуатация; сопровождение;  
утилизация.

## **Специальные процессы:**

определение и установка взаимосвязей  
исходя из задач и целей.

# Стадии создания системы по стандарту ISO/IEC 15288

№ п/п	Стадия	Описание
1	Формирование концепции	Анализ потребностей, выбор концепции и проектных решений
2	Разработка	Проектирование системы
3	Реализация	Изготовление системы
4	Эксплуатация	Ввод в эксплуатацию и использование системы
5	Поддержка	Обеспечение функционирования системы
6	Снятие с эксплуатации	Прекращение использования, демонтаж, архивирование системы