

# **ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ**

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Информационная система (ИС) – основной компонент автоматизированных систем управления.
- Проектирование информационных систем (ИС) – процесс создания и сопровождения систем в виде жизненного цикла (ЖЦ) информационной системы.
- Жизненный цикл ИС – некоторая последовательность стадий и выполняемых на них процессов.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

Для каждого этапа ЖЦ определяются:

- состав и последовательность выполняемых работ,
  - получаемые результаты,
  - методы и средства, необходимые для выполнения работ,
  - роли и ответственность участников и т.д.
- 
- Такое *формальное описание* ЖЦ ИС позволяет спланировать и организовать процесс коллективной разработки и обеспечить управление этим процессом.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Жизненный цикл ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования.
- *Модель* жизненного цикла отражает различные состояния системы, начиная с момента возникновения необходимости в данной ИС и заканчивая моментом ее полного выхода из употребления.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

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

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

В настоящее время известны и  
используются

три модели жизненного цикла:

- каскадная модель;
- поэтапная модель с промежуточным контролем;
- спиральная модель.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

*Каскадная* модель (рис. 1) –  
предусматривает последовательное  
выполнение всех этапов проекта в  
строго фиксированном порядке.

- Переход на следующий этап означает полное завершение работ на предыдущем этапе.

# 8. ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

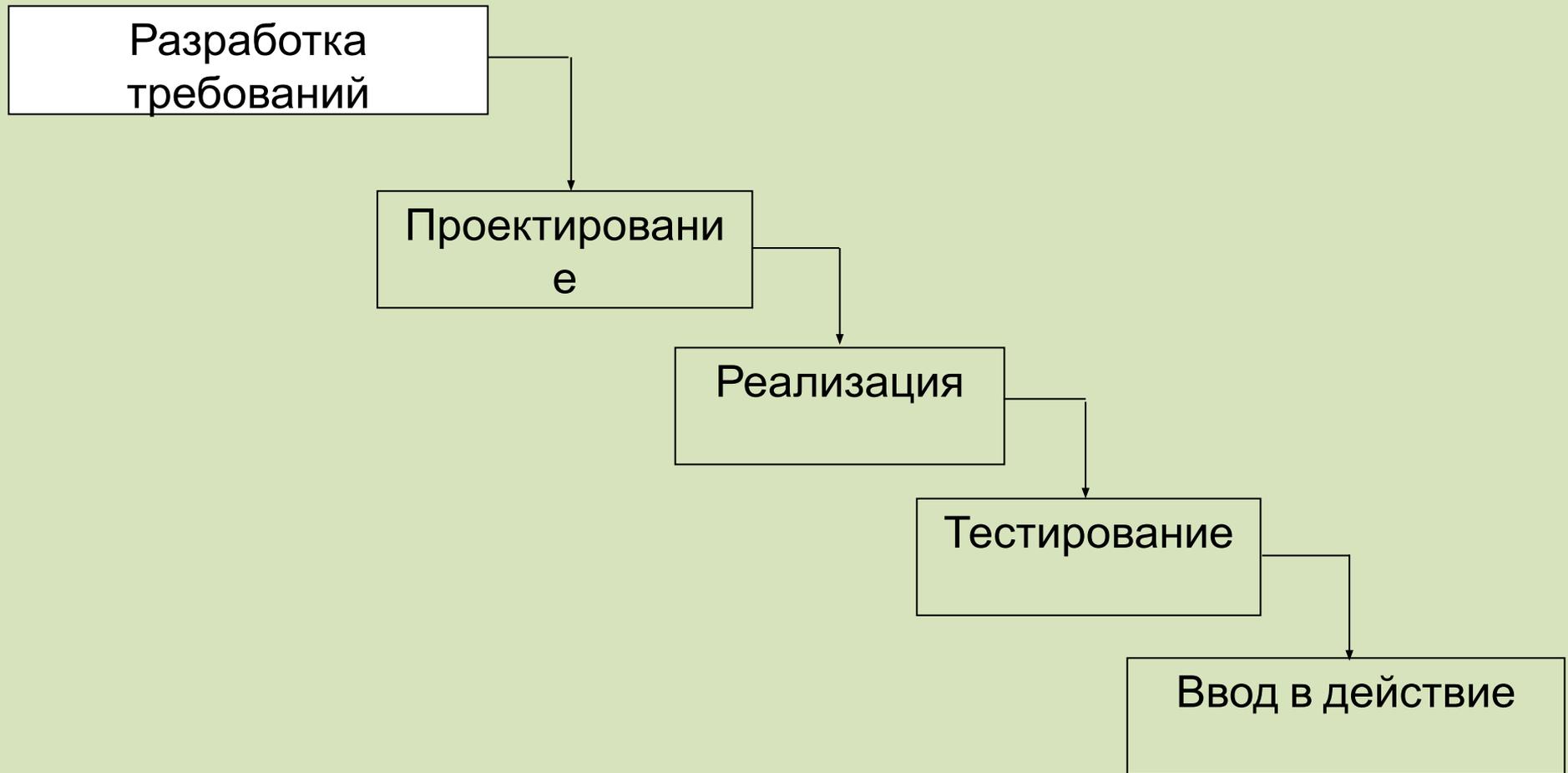
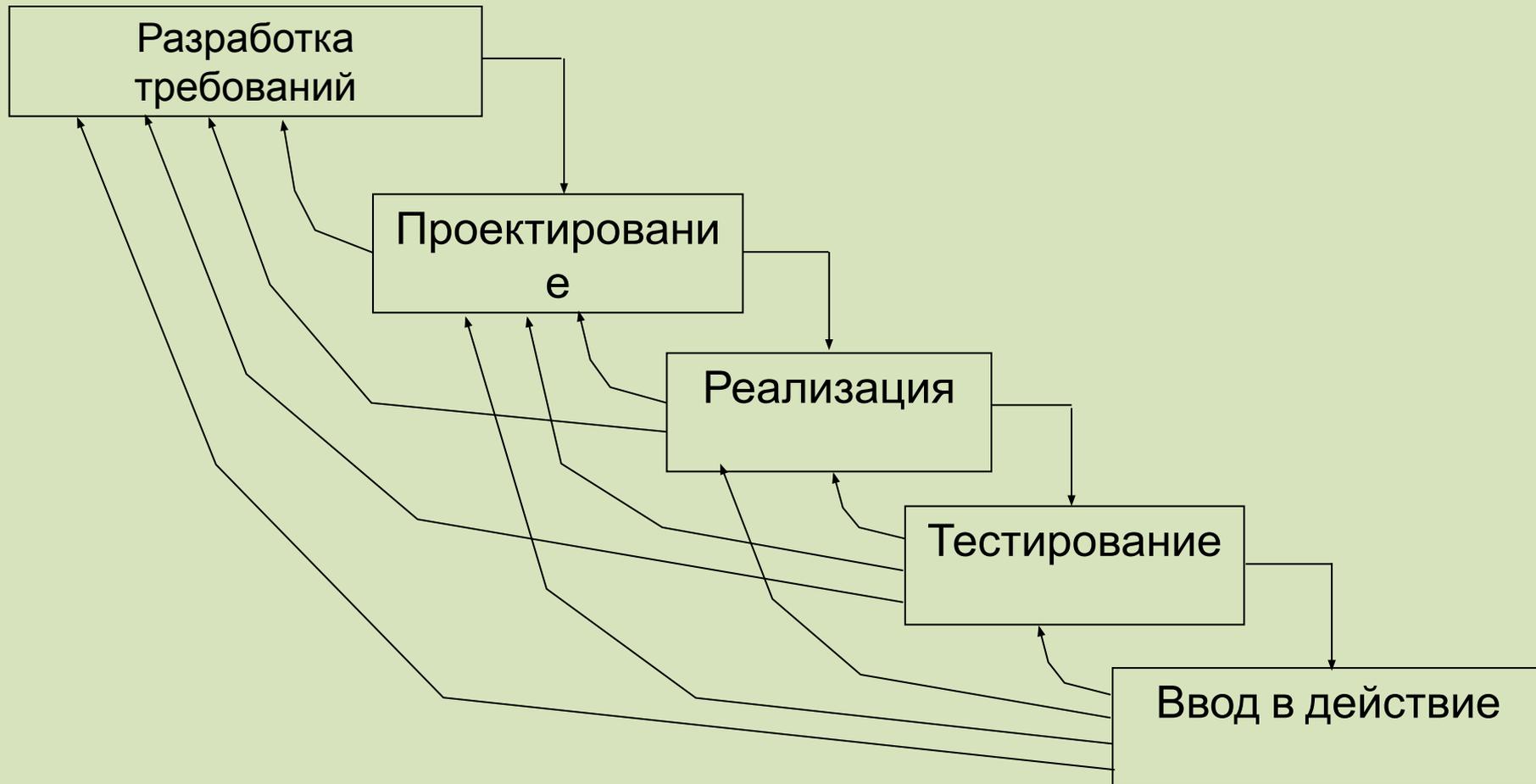


Рис. 1. Каскадная модель ЖЦ ИС

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- *Поэтапная модель с промежуточным контролем*  
(рис.2) – 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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

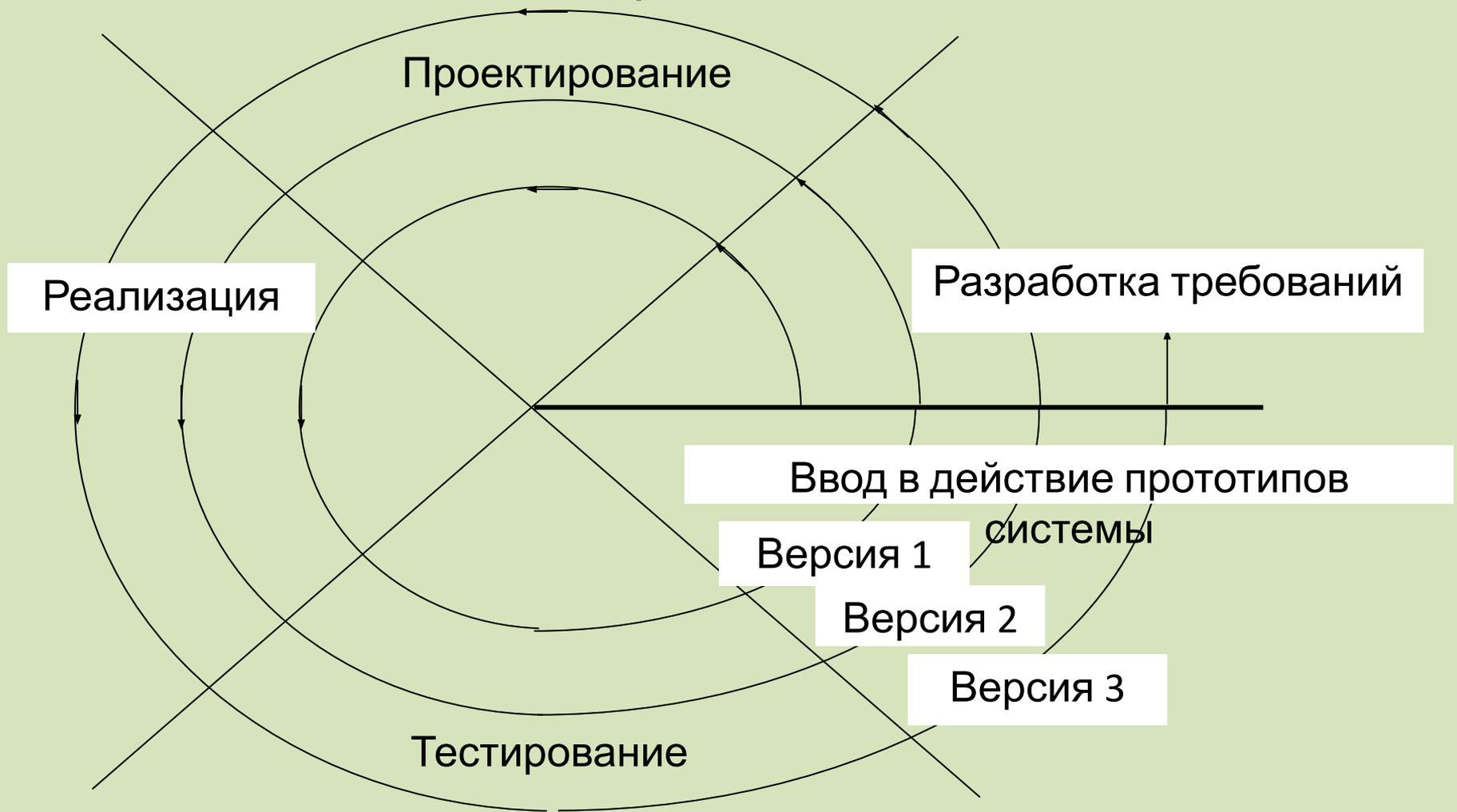


- Рис. 2. Поэтапная модель с промежуточным контролем

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- *Спиральная модель* (рис. 3) – на каждом витке спирали:
- выполняется создание очередной версии продукта,
- уточняются требования проекта,
- определяется его качество,
- 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).

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ



- Рис. 3. Спиральная модель ЖЦ ИС

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- 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.).

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

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.
- Каждый этап завершался после полного выполнения и документального оформления всех предусмотренных работ.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

*Преимущества* 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.
- 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. 15

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Основной *недостаток* каскадного подхода –  
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

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Однако и эта схема (поэтапная модель с промежуточным контролем) 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 получают систему, не удовлетворяющую их реальным потребностям.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Каскадная модель предполагает разработку законченных продуктов на каждом этапе:
- технического задания,
- технического проекта,
- программного продукта
- пользовательской документации.

Разработанная документация позволяет не только определить требования к продукту следующего этапа, но и определить обязанности сторон, объем работ и сроки, при этом окончательная оценка сроков и стоимости проекта производится на начальных этапах, после завершения обследования.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Если требования к ИС 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 пользователя.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- 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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

На каждом витке спирали:

- уточняются требования, цели и характеристики проекта,
  - определяется качество разработки,
  - планируются 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

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Итеративная разработка объективно отражает существующий спиральный цикл создания сложных систем.
- Она позволяет переходить на следующий этап, не дожидаясь полного завершения работы на текущем этапе.
- Кроме того, решается *главная задача* – как можно быстрее показать пользователям системы работоспособный продукт, тем самым активизируя процесс уточнения и дополнения требований.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Основная *проблема* спирального цикла – определение момента перехода на следующий этап.
- Для решения этой проблемы – 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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- В некоторых областях спиральная модель не может применяться, т.к. невозможно использование/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, изменение роли сотрудников).

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Трудозатраты при поэтапном итерационном внедрении оказываются значительно выше, а управление проектом требует настоящего искусства.
- Предвидя указанные сложности, заказчики выбирают каскадную модель, чтобы "внедрять систему один раз".

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Каждая из стадий создания системы предусматривает выполнение определенного объема работ, которые представляются в виде процессов ЖЦ.
- *Процесс* – совокупность взаимосвязанных действий, преобразующих входные данные в выходные.
- *Описание* каждого процесса включает в себя перечень
  - решаемых задач,
  - исходных данных,
  - результатов.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Существует целый ряд *стандартов*, регламентирующих ЖЦ программного обеспечения ИС, а в некоторых случаях и процессы разработки.
- Значительный вклад в теорию проектирования и разработки информационных систем внесла компания IBM, предложив еще в середине 1970-х годов методологию BSP (Business System Planning – *методология организационного планирования*).

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Наиболее известные стандарты:
- ГОСТ 34.601-90
- ISO/IEC 12207:1995
- Custom Development Method (CDM) по разработке прикладных информационных систем (методика Oracle)
- Rational Unified Process (RUP)
- Microsoft Solution Framework (MSF)
- Extreme Programming (XP).

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- **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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- **ISO/IEC 12207:1995** – стандарт на процессы и организацию жизненного цикла. Распространяется на все виды заказного программного обеспечения.
- Стандарт не содержит описания фаз, стадий и этапов.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Custom Development Method (CDM) по разработке прикладных информационных систем (методика Oracle) – 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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- Rational Unified Process (RUP) – итеративная модель разработки, содержит 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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- 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.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

- 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 последовательного подхода к разработке.

# ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

В соответствии с базовым международным стандартом ISO/IEC 12207 все процессы ЖЦ программного обеспечения ИС делятся на три группы:

- *Основные* процессы: приобретение; поставка; разработка; эксплуатация; сопровождение.
- *Вспомогательные* процессы: документирование; управление конфигурацией; обеспечение качества; разрешение проблем; аудит; аттестация; совместная оценка; верификация.
- *Организационные* процессы: создание инфраструктуры; управление; обучение; совершенствование.