

# ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ.

**ОСНОВНЫЕ ПОНЯТИЯ И ПОДХОДЫ**

# Технология программирования

совокупность методов, средств и процедур,  
используемых в процессе разработки  
программного обеспечения.

# Структура описания технологической операции



# Типы программ

- Автономное ПО - устанавливается на одиночный компьютер, не связано с другим программным и аппаратным обеспечением.
- Встроенное ПО – часть уникального приложения с привлечением аппаратного обеспечения.
- ПО реального времени - должно выполнять функции в течении малого интервала времени с минимальным временем отклика
- Сетевое ПО – состоит из частей взаимодействующих через сеть.

# Жизненный цикл программного обеспечения

период от момента появления идеи создания некоторого программного обеспечения до момента завершения его поддержки фирмой-разработчиком или фирмой, выполнявшей сопровождение.

Состав процессов жизненного цикла регламентируется международным стандартом ISO/IEC 12207: 1995 «Information Technology - Software Life Cycle Processes»

# Структура процессов ЖЦПО

## **Основные процессы**

- Приобретение
- Поставка
- Разработка
- Эксплуатация
- Сопровождение

## **Организационные процессы**

- Управление
- Усовершенствование
- Создание инфраструктуры
- Обучение

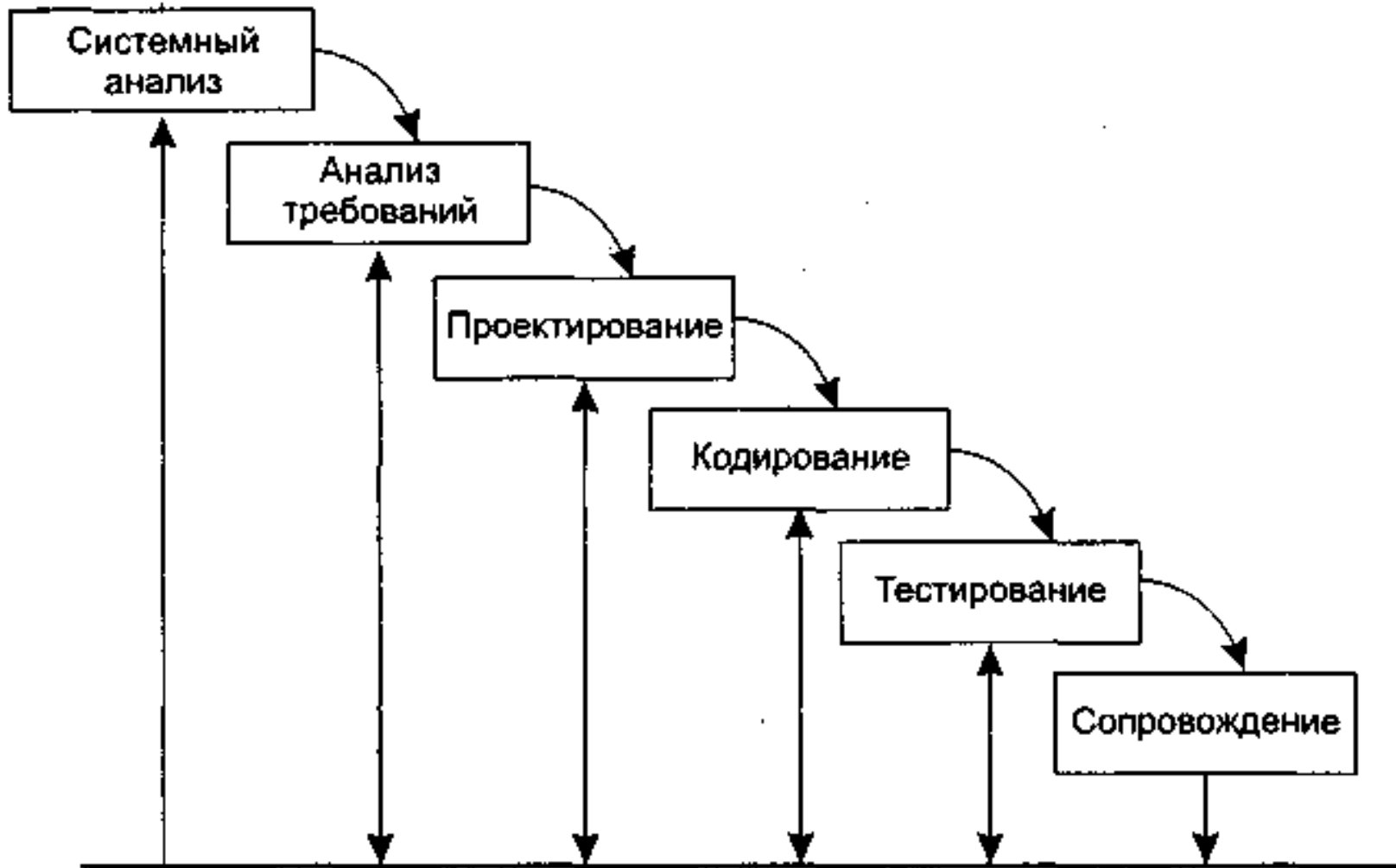
## **Вспомогательные процессы**

- Документирование
- Управление конфигурацией
- Обеспечение качества
- Верификация
- Аттестация
- Аудит
- Разрешение проблем

# Стратегии конструирования ПО

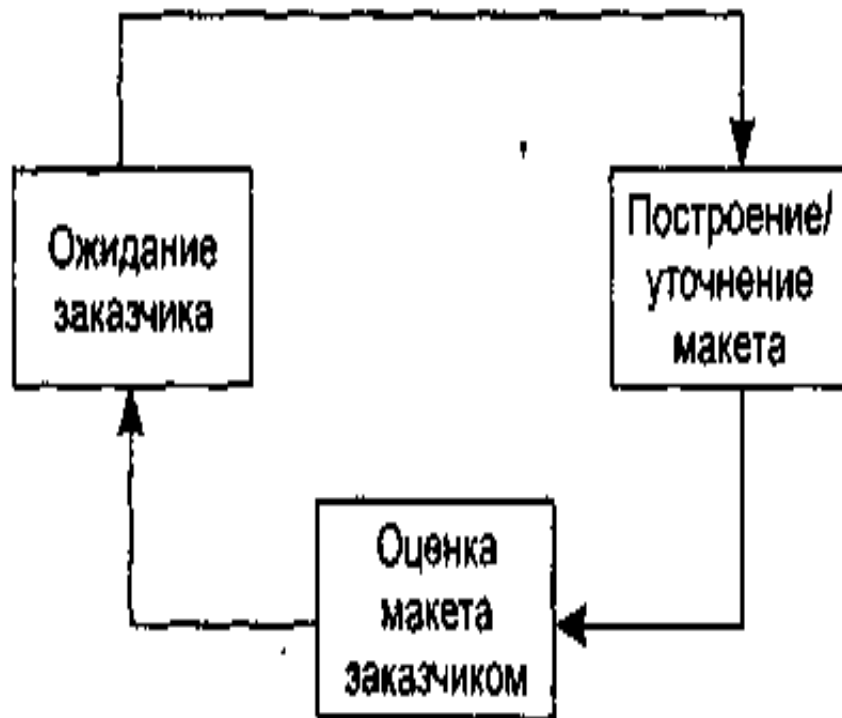
- *однократный проход (водопадная стратегия) ;*
- *инкрементная стратегия;*
- *эволюционная стратегия.*

# Водопадная модель ЖЦПО





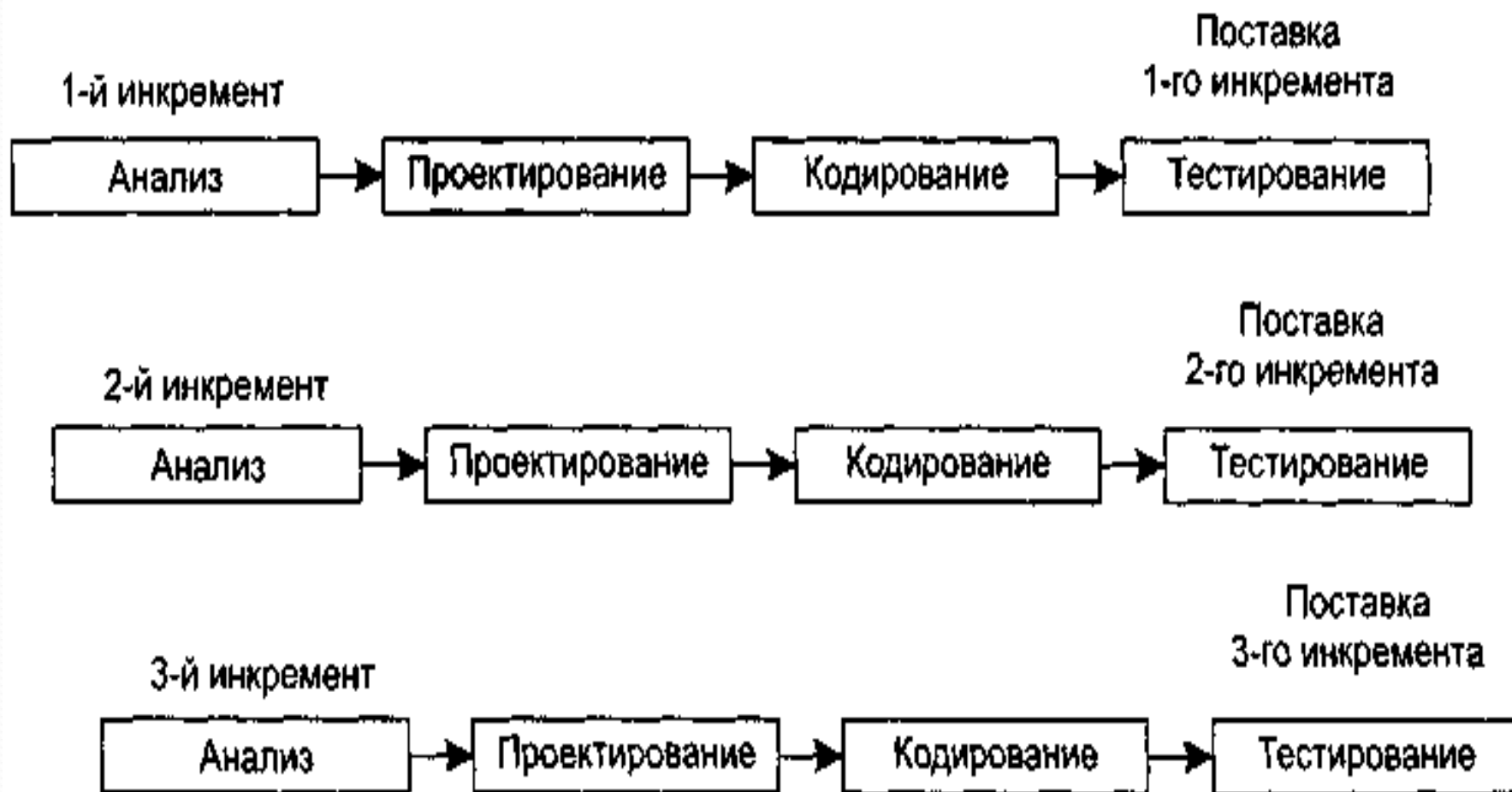
# Макетирование



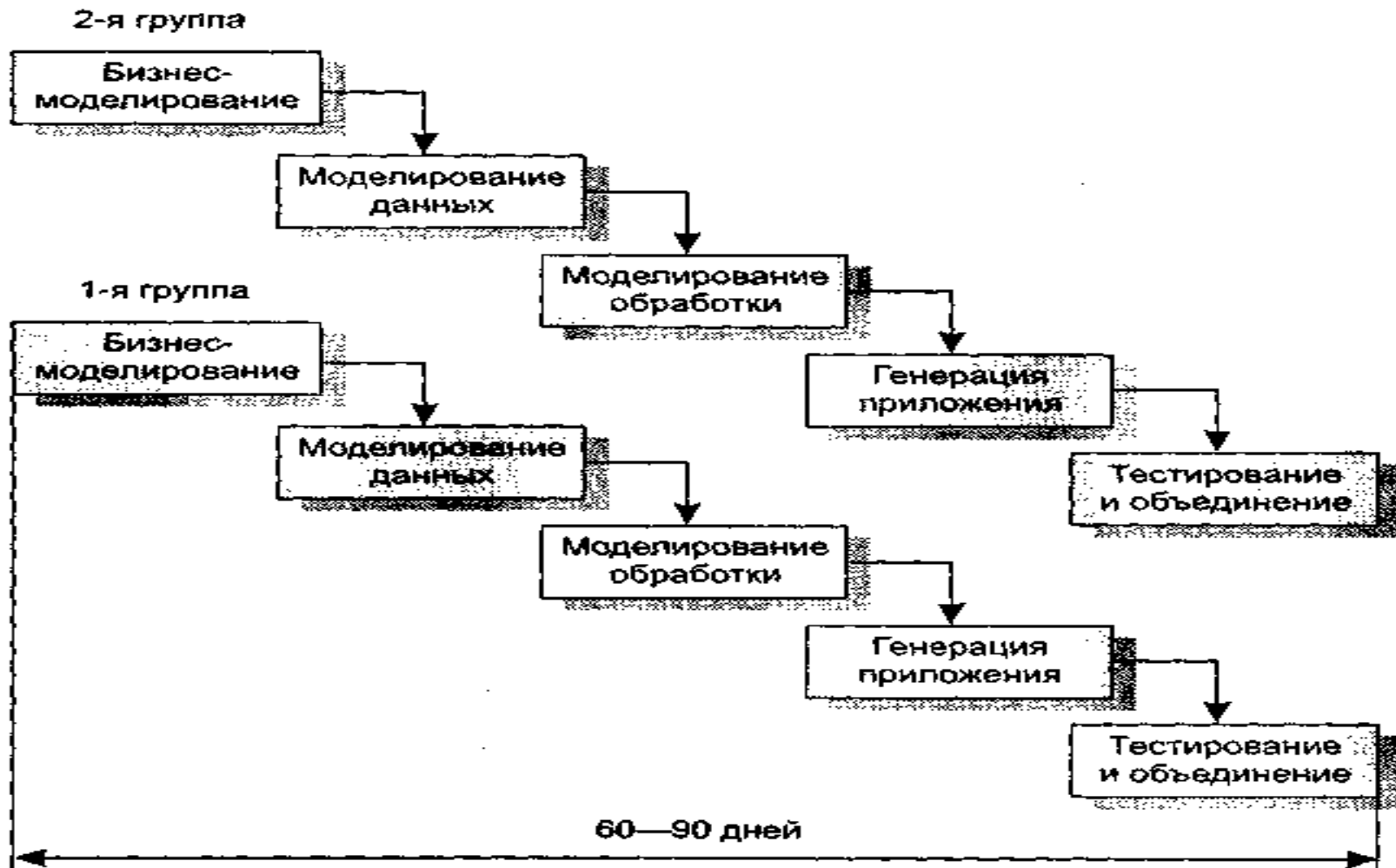
# Спиральная модель БОЭМА



# Инкрементальная модель



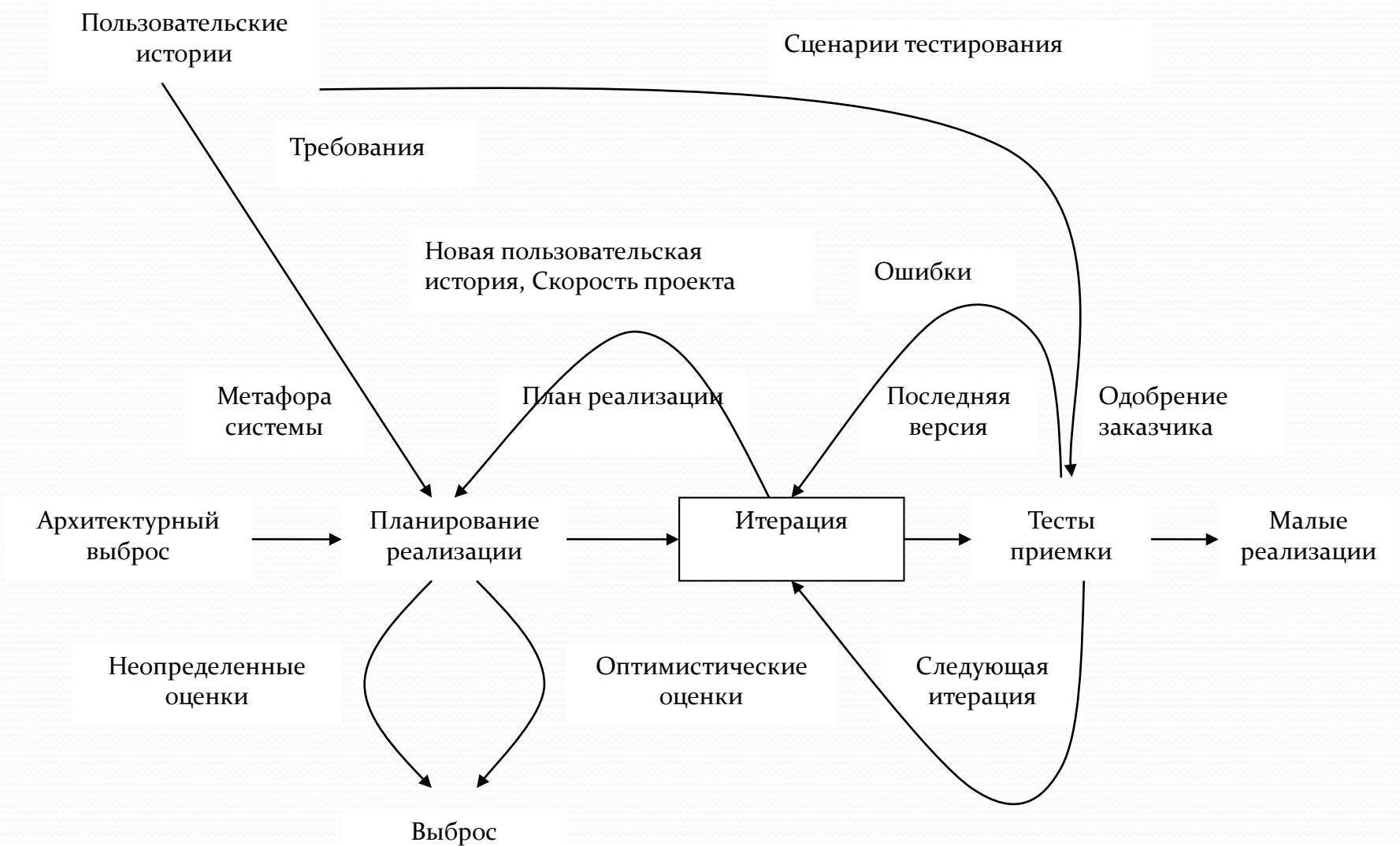
# Модель быстрой разработки приложений (RAD)



# Основные возможности и преимущества RAD-технологий



# Схема процесса XR.



	Достоинства	Недостатки
Классический ЖЦ	План и временной график	Реальные проекты требуют отклонения от стандартного плана, требуется точная формулировка исходных требований, результаты доступны только по окончании всех работ
Макетирование	Обеспечивает определение полных требований	Заказчик и разработчик могут принять макет за продукт.
Эволюционная модель	Реально отображает разработку ПО, Учитывает риск на каждом витке, использует моделирование для уменьшения риска.	Повышенные требования к заказчику, Трудности контроля времени разработки.
Инкрементная модель	На каждом инкременте рабочий продукт, разбиение проекта на части	Требует существенных людских ресурсов, не применима в условиях высоких технических рисков

**Экстремальное  
программирование**

**Короткие сроки  
разработки версии,  
контроль времени  
разработки,  
отсутствие строгой  
архитектуры  
системы, постоянная  
качественная  
обратная связь с  
заказчиком**

**Невыполнимость  
сложных проектов,  
невозможность  
предсказать  
результаты  
длительного проекта ,  
неприспособленность  
ХР для задач где  
решения не находятся  
сразу на основе ранее  
полученного опыта, а  
требуют проведения  
предварительных  
исследований.**



# Классификация программных продуктов по назначению



# Эксплуатационные требования к программным продуктам

- правильность;
- универсальность;
- надежность (помехозащищенность) ;
- проверяемость;
- точность результатов;
- защищенность;
- программная совместимость;
- аппаратная совместимость;
- эффективность;
- адаптируемость;
- повторная входимость;
- реентерабельность.

# «Техническое задание. Требования к содержанию и оформлению».

## ГОСТ 19.201-78

- введение;
- основания для разработки;
- назначение разработки;
- требования к программе или программному изделию;
- требования к программной документации;
- технико-экономические показатели;
- стадии и этапы разработки;
- порядок контроля и приемки.

- ПРИМЕР. Разработать техническое задание на программный продукт, предназначенный для наглядной демонстрации школьникам графиков функций одного аргумента  $y = f(x)$ . Разрабатываемая программа должна рассчитывать таблицу значений и строить график функций на заданном отрезке по заданной формуле и менять шаг аргумента и границы отрезка. Кроме этого, программа должна запоминать введенные формулы.

## 1. ВВЕДЕНИЕ

Настоящее техническое задание распространяется на разработку программы построения графиков и таблиц значений функций одной переменной, предназначенной для использования школьниками старших классов. В школьном курсе элементарной алгебры тема анализа функций является одной из самых сложных. При изучении данной темы школьники должны научиться исследовать и строить графики функций одной переменной, используя все известные характеристические точки функции, включая корни, точки разрыва первого и второго рода и т. д.

Существующее программное обеспечение, которое может решать подобные задачи, является универсальным, например Eurica или MathCad. Оно имеет сравнительно сложный пользовательский интерфейс, ориентированный на пользователя, прослушавшего, как минимум, институтский курс высшей математики, что делает использование подобных средств школьниками невозможным.

## 2. ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ

Программа разрабатывается на основе учебного плана кафедры «Электронные вычислительные машины» и в соответствии с договором кафедры со школой № ... от 5.09.2011.

## 3. НАЗНАЧЕНИЕ

Основным назначением программы является помощь школьникам при изучении раздела «Исследование функций одного аргумента» школьного курса элементарной алгебры.

## 4. ТРЕБОВАНИЯ К ПРОГРАММЕ ИЛИ ПРОГРАММНОМУ ИЗДЕЛИЮ

### 4.1. Требования к функциональным характеристикам

#### 4.1.1. Программа должна обеспечивать возможность выполнения следующих функций:

- ввод аналитического представления функции одной переменной и длительное хранение его в системе;
- ввод и изменение интервала определения функции;
- ввод и корректировку шага аргумента;
- построение таблицы значений функции на заданном интервале и изображение графика функции на заданном интервале при условии, что на указанном интервале она не имеет точек разрыва.

#### 4.1.2. Исходные данные:

- аналитическое задание функции;
- интервал определения функции;
- шаг изменения аргумента, определяющий количество точек на интервале.

#### 4.2. Требования к надежности

4.2.1. Предусмотреть контроль вводимой информации.

4.2.2. Предусмотреть блокировку некорректных действий пользователя при работе с системой.

#### 4.3. Требования к составу и параметрам технических средств

4.3.1. Система должна работать на IBM совместимых персональных компьютерах.

4.3.2. Минимальная конфигурация:

- тип процессора .Pentium и выше;
- объем оперативного запоминающего устройств ...32 Мб и более.

#### 4.4. Требования к информационной и программной совместимости

Система должна работать под управлением семейства операционных систем Win 32 (Windows 2000, Windows NT и т. п.).

### 5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

5.1. Разрабатываемые программные модули должны быть самодокументированы, т. е. тексты программ должны содержать все необходимые комментарии.

5.2. Разрабатываемая программа должна включать справочную информацию об основных терминах соответствующего раздела математики и подсказки учащимся.

5.3. В состав сопровождающей документации должны входить:

5.3.1. Пояснительная записка на 25-30 листах, содержащая описание разработки.

5.3.2. Руководство пользователя.