

# Программная инженерия

# Метрики объектно-ориентированных программных систем

# Метрические особенности ООП систем

Объектно-ориентированные метрики вводятся с целью:

- ▶ улучшить понимание качества продукта;
- ▶ оценить эффективность процесса разработки;
- ▶ улучшить качество работы на этапе проектирования.

С точки зрения метрик выделяют пять характеристик объектно-ориентированных систем:

- ▶ Локализация;
- ▶ Инкапсуляция;
- ▶ Информационная закрытость;
- ▶ Наследование;
- ▶ Способы абстрагирования объектов.

# Характеристики ООС

- ▶ **Локализация** фиксирует способ группировки информации в программе.

Основной механизм локализации в классических методах - функция, программные метрики ориентированы на внутреннюю структуру или сложность функций или на способ, которым функции связываются друг с другом. В ООС базовым элементом является класс, то локализация здесь основывается на объектах.

- ▶ **Инкапсуляция** - это связывание совокупности элементов.

Для классических ПС примерами низкоуровневой инкапсуляции является записи и массивы. В ООС инкапсулируются обязанности класса, представляемые его атрибутами, операциями и состояниями.

- ▶ **Информационная закрытость** делает невидимыми операционные детали программного компонента.
- ▶ **Наследование** - это механизм, обеспечивающий тиражирование обязанностей одного класса в другие классы.
- ▶ **Абстракция** - это механизм, который позволяет проектировщику выделять 4  
главное в программном компоненте, без учета второстепенных деталей.

# Эволюция мер связи для ООП систем

## Связность объектов:

- 1) Связность по совпадению;
- 2) Логическая связность;
- 3) Временная связность;
- 4) Процедурная связность;
- 5) Коммуникативная связность;
- 6) Информационная связность;
- 7) Функциональная связность;
- 8) Объектная связность.

# Метрики для объектно-ориентированного тестирования

## Метрики инкапсуляции:

- Метрика 1. Недостаток связности в методах LCOM;
- Метрика 2. Процент публичных и защищенных PPR;
- Метрика 3. Публичный доступ к компонентным данным PAD;

## Метрики наследования:

- Метрика 4. Количество корневых классов NOR;
- Метрика 5. Коэффициент объединения по входу FIN;
- Метрика 6. Количество детей NOC;
- Метрика 7. Высота дерева наследования DIT.

# Метрики для объектно-ориентированного тестирования

## Метрики полиморфизма:

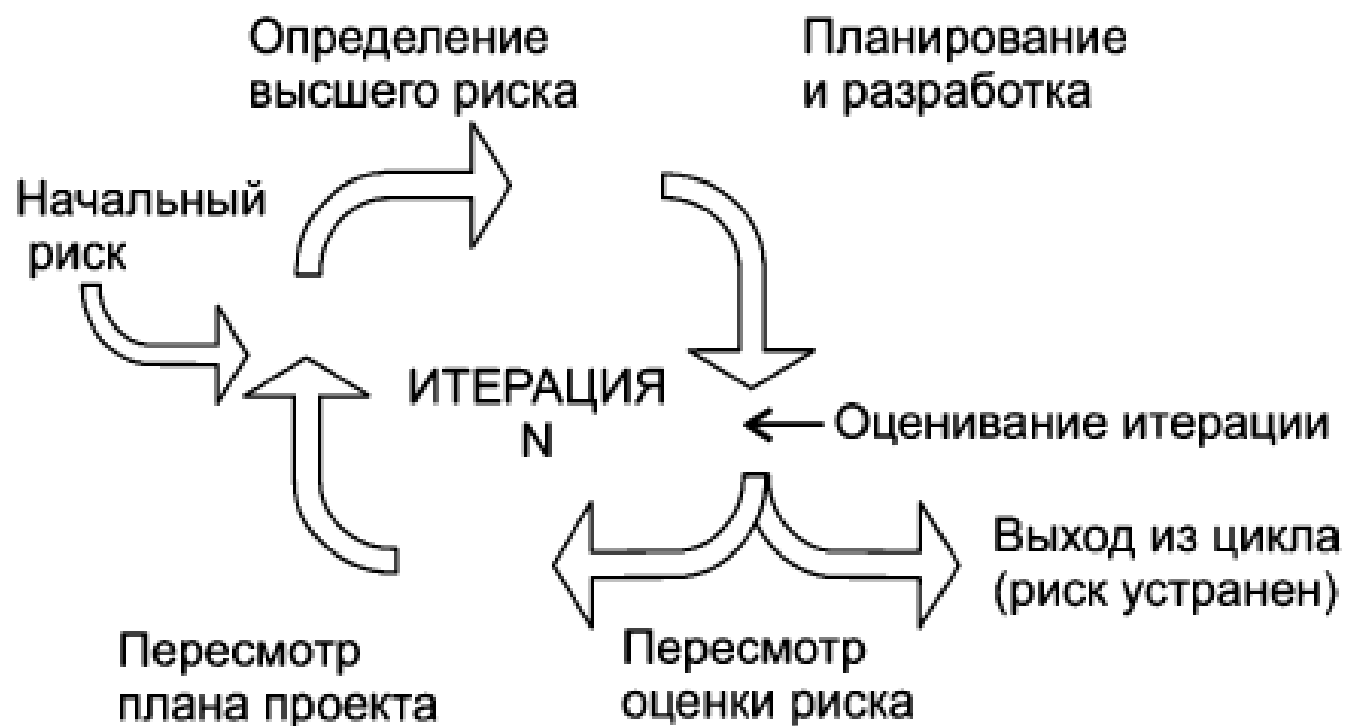
- Метрика 8. Процентное количество не переопределенных запросов OVR;
- Метрика 9. Процентное количество динамических запросов DYN;
- Метрика 10. Скачок класса Bounce-C;
- Метрика 11. Скачок системы Bounce-S;

# Примеры объектно-ориентированных процессов разработки

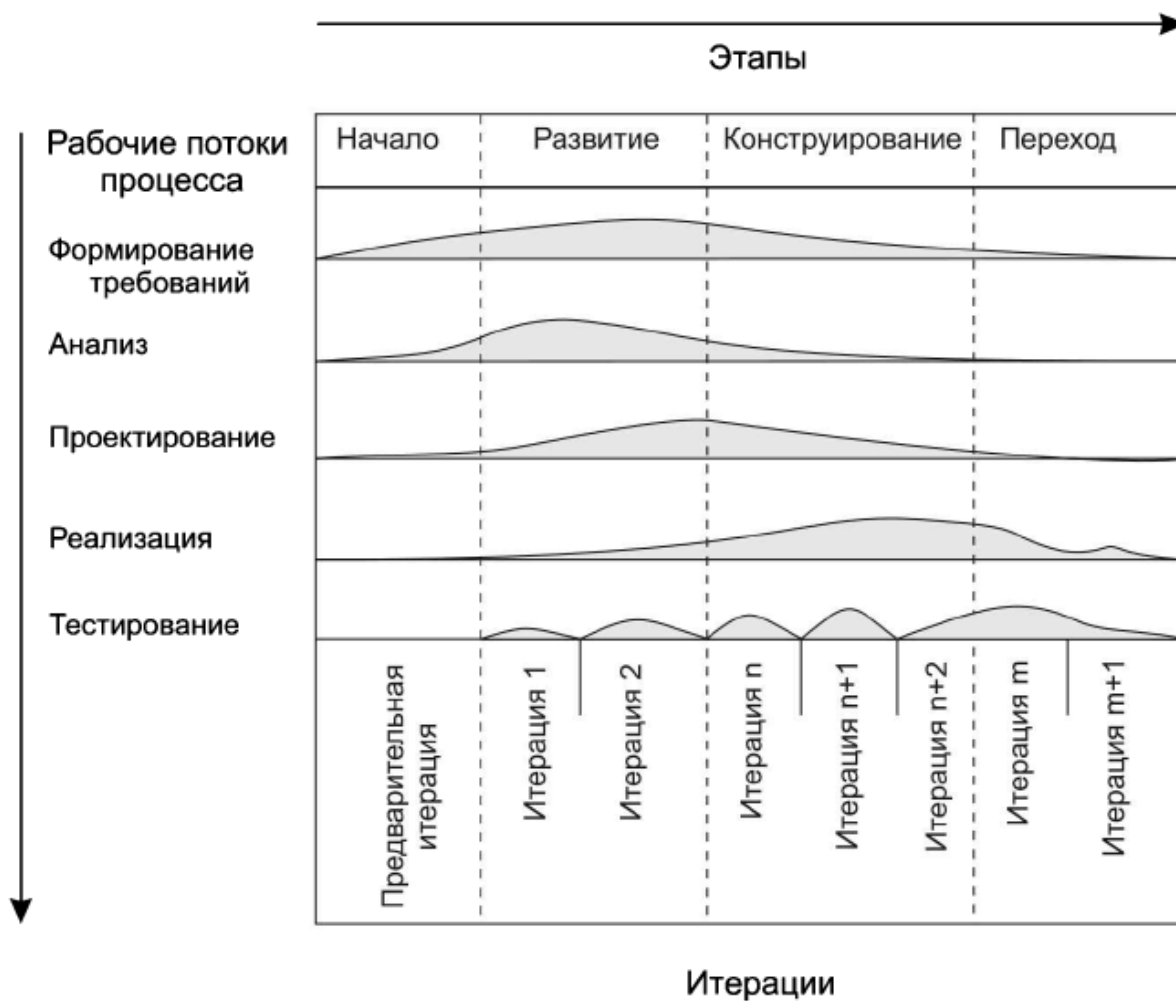


# Основные понятия унифицированного процесса разработки

Процесс разработки программной системы организуется в виде эволюционно-инкрементного жизненного цикла.



# Измерения унифицированного процесса разработки



# Этапы унифицированного процесса разработки

## Этап НАЧАЛО (Inception)

Главное назначение этапа - запустить проект

## Этап РАЗВИТИЕ (Elaboration)

Главное назначение этапа - создать архитектурный базис системы

## Этап КОНСТРУИРОВАНИЕ (Construction)

Главное назначение этапа - создать программный продукт, который обеспечивает начальные операционные возможности

## Этап ПЕРЕХОД (Transition)

Главное назначение этапа - применить программный продукт в среде пользователей и завершить реализацию продукта.

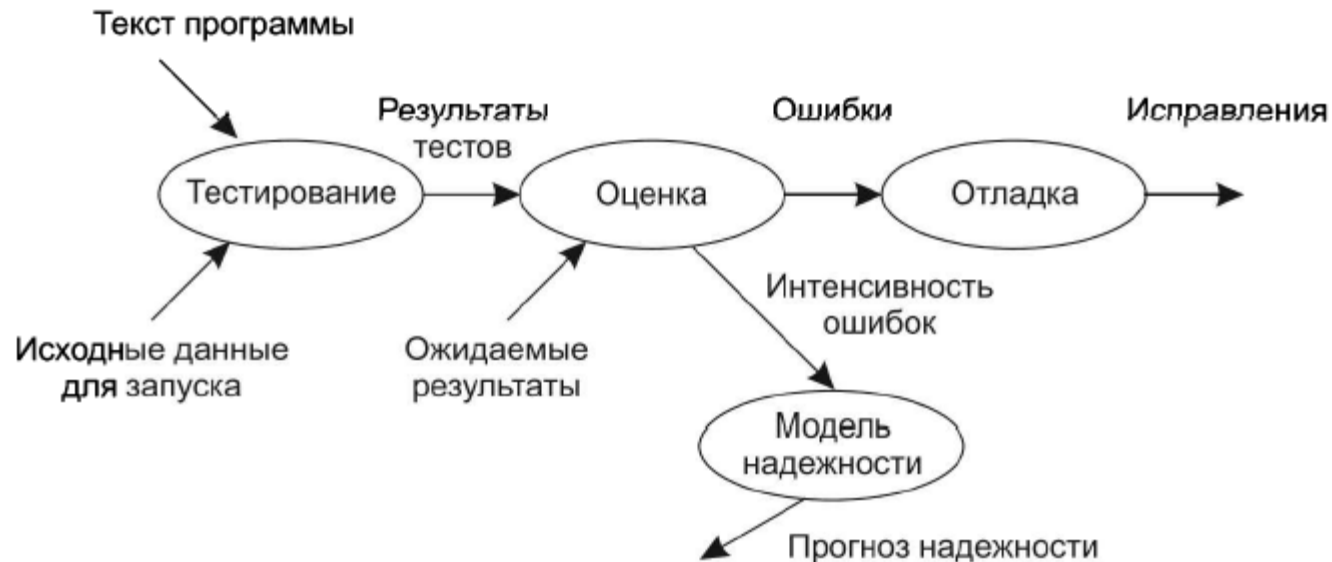
# Тестирование программного обеспечения

# Основные понятия и принципы тестирования ПО

**Тестирование** - это процесс выполнения программы с целью обнаружения ошибок.

Полную проверку программы гарантирует исчерпывающее тестирование.

Целью проектирования тестовых вариантов является систематическое обнаружение различных классов ошибок при минимальных затратах времени и стоимости.



# Методика тестирования программных систем

Тестирование - это процесс выполнения программы с целью обнаружения ошибок.

Полную проверку программы гарантирует исчерпывающее тестирование.

Целью проектирования тестовых вариантов является систематическое обнаружение различных классов ошибок при минимальных затратах времени и стоимости.



# Процесс тестирования программных систем

## 1. Тестирование элементов

Объект тестирования = модуль

- ❖ интерфейс модуля;
- ❖ внутренние структуры данных;
- ❖ независимые пути;
- ❖ пути обработки ошибок;
- ❖ граничные условия.

## 2. Тестирование интеграции

- нисходящее тестирование интеграции;
- восходящее тестирование интеграции.

## 3. Тестирование правильности

Цель - подтвердить, что функции, описанные в спецификации требований к ПС, соответствуют ожиданиям заказчика.

## 4. Системное тестирование

- тестирование восстановления;
- тестирование безопасности;
- стрессовое тестирование;
- Тестирование производительности.

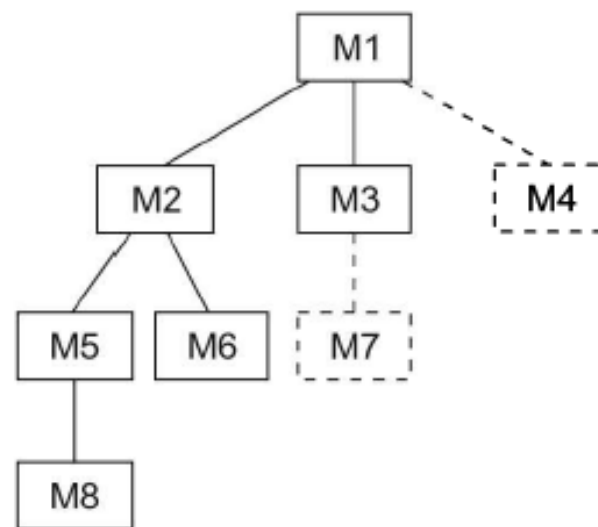
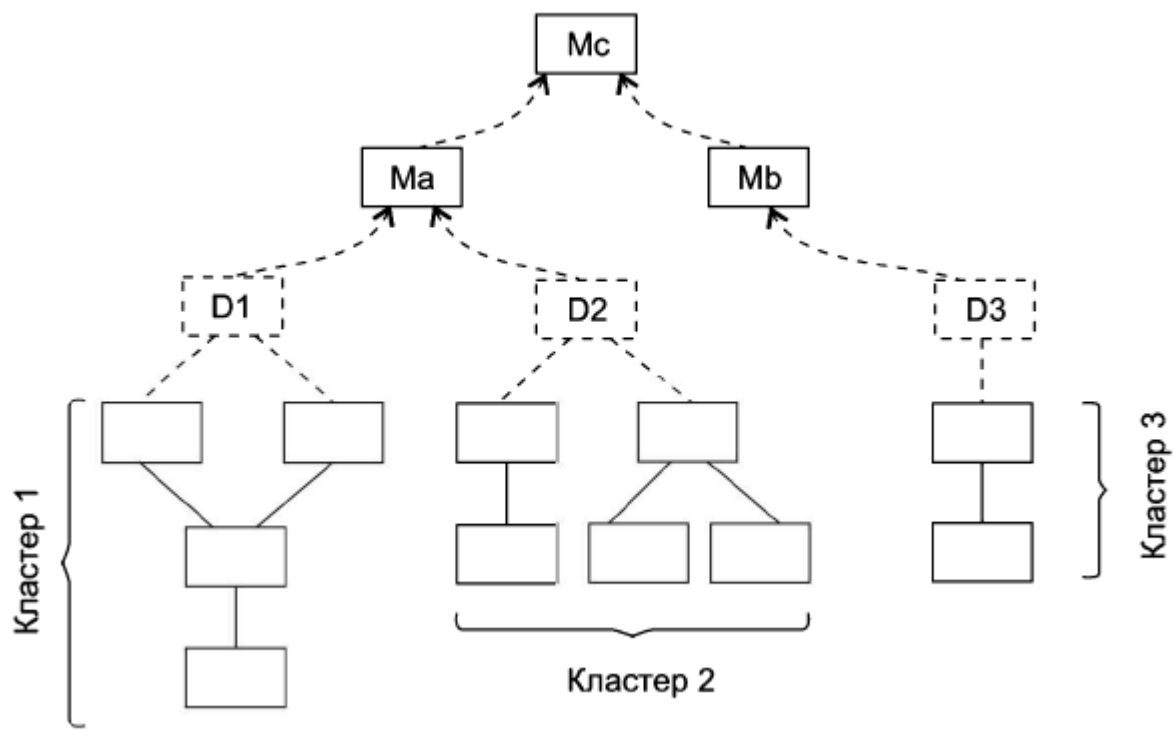
## 5. Отладка

Отладка - следствие успешного тестирования

# Тестирование интеграции

Восходящая интеграция системы

Нисходящая интеграция системы





# СПАСИБО ЗА ВНИМАНИЕ!

Если остались вопросы, задавайте их  
в ЭИОС МТУСИ или пишите на почту