

Соловьев С.Ю.

**Дополнительные главы
программной инженерии**

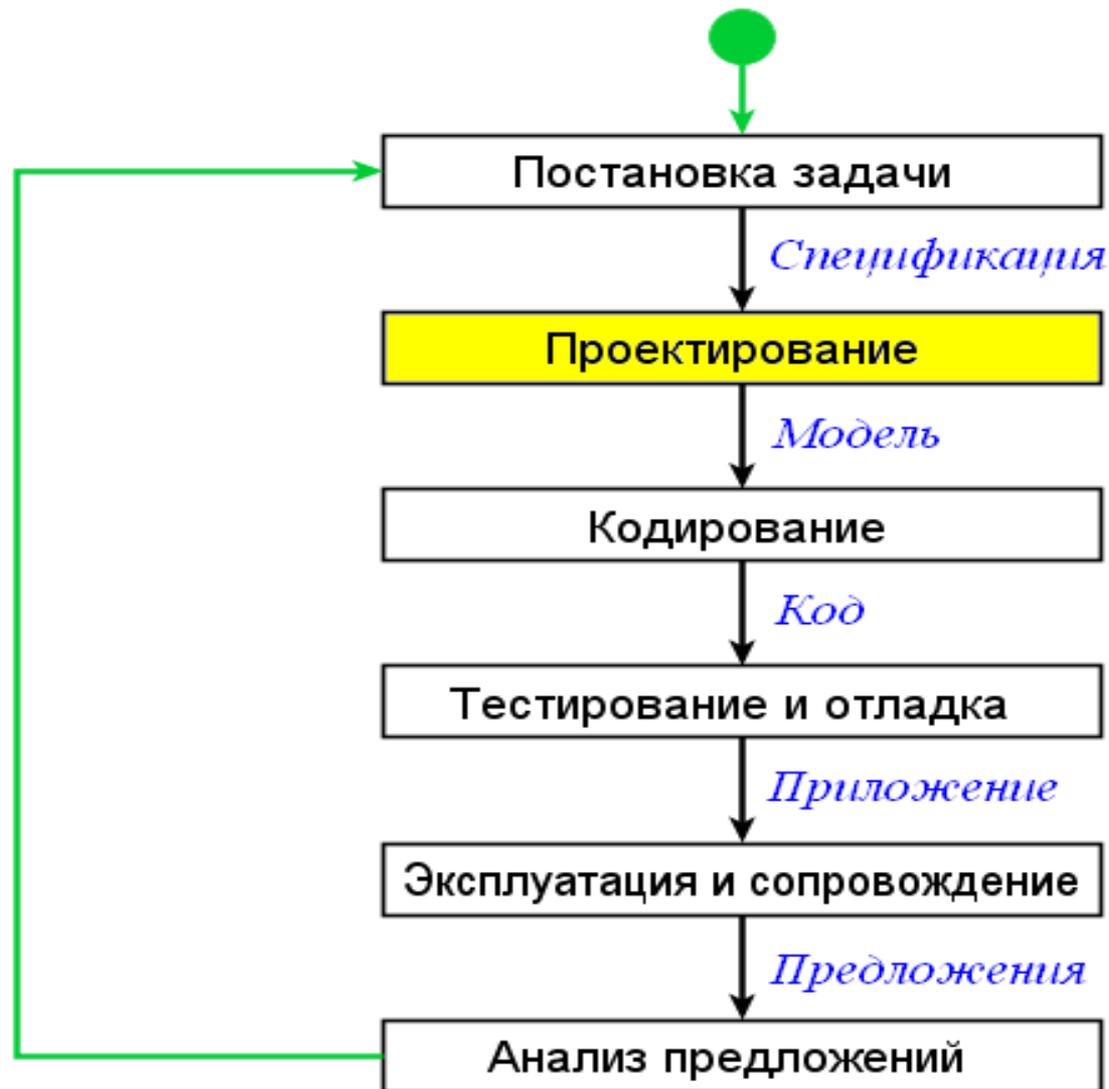
(5)

Детальное проектирование

2021

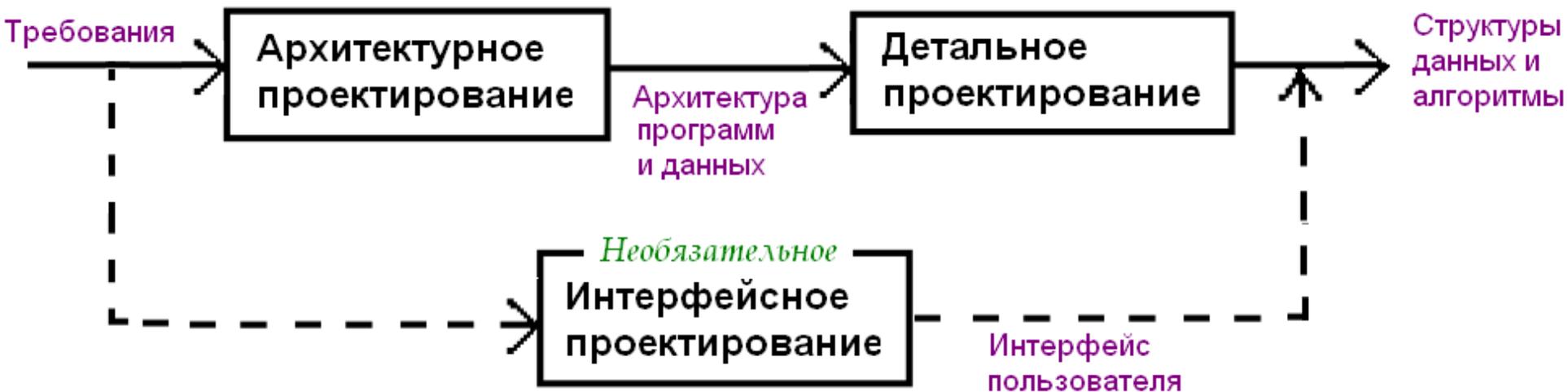
Напоминание

Жизненный цикл

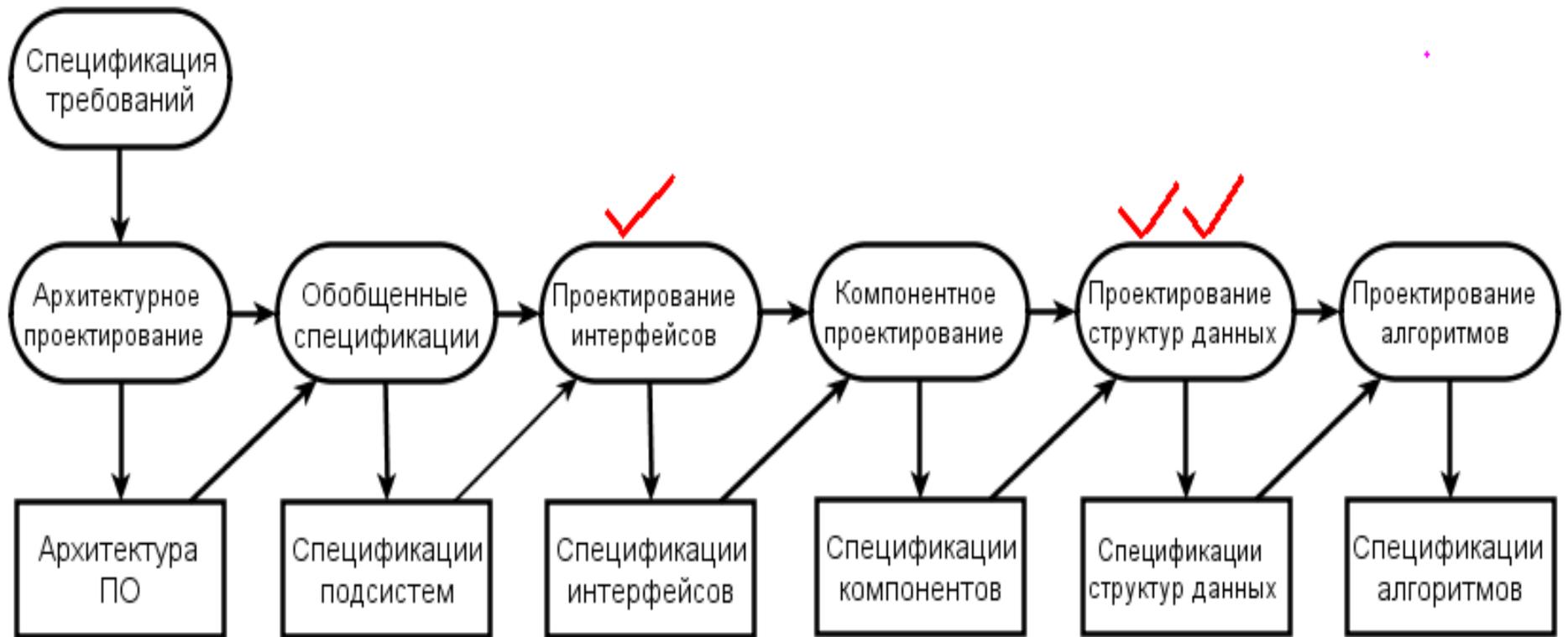


Проектирование ПО – преобразование высокоуровневых требований к ПО в инженерное представление этого ПО.

Различают архитектурное, детальное и интерфейсное проектирования.



Этапы проектирования



Детальное проектирование ПО

- описание компонентов ПО и интерфейсов между ними; а также
- разработка и документирование требований к тестам и общего плана тестирования.

Детальное проектирование состоит в декомпозиции компонентов нижнего уровня архитектурного проекта до модулей на выбранном языке программирования.

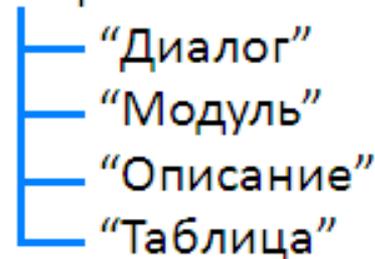
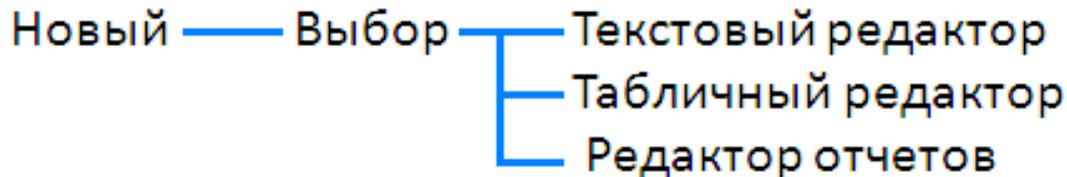
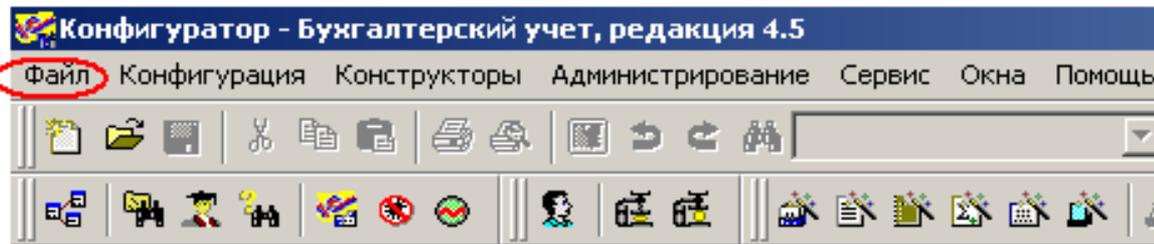
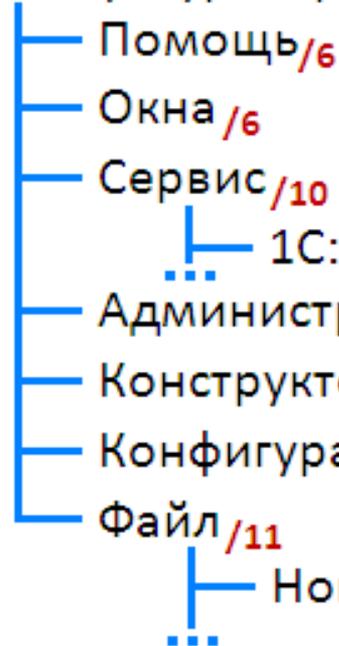
Структурная модель: Структурная схема → Меню

Объектно-ориентированная модель: Паттерны

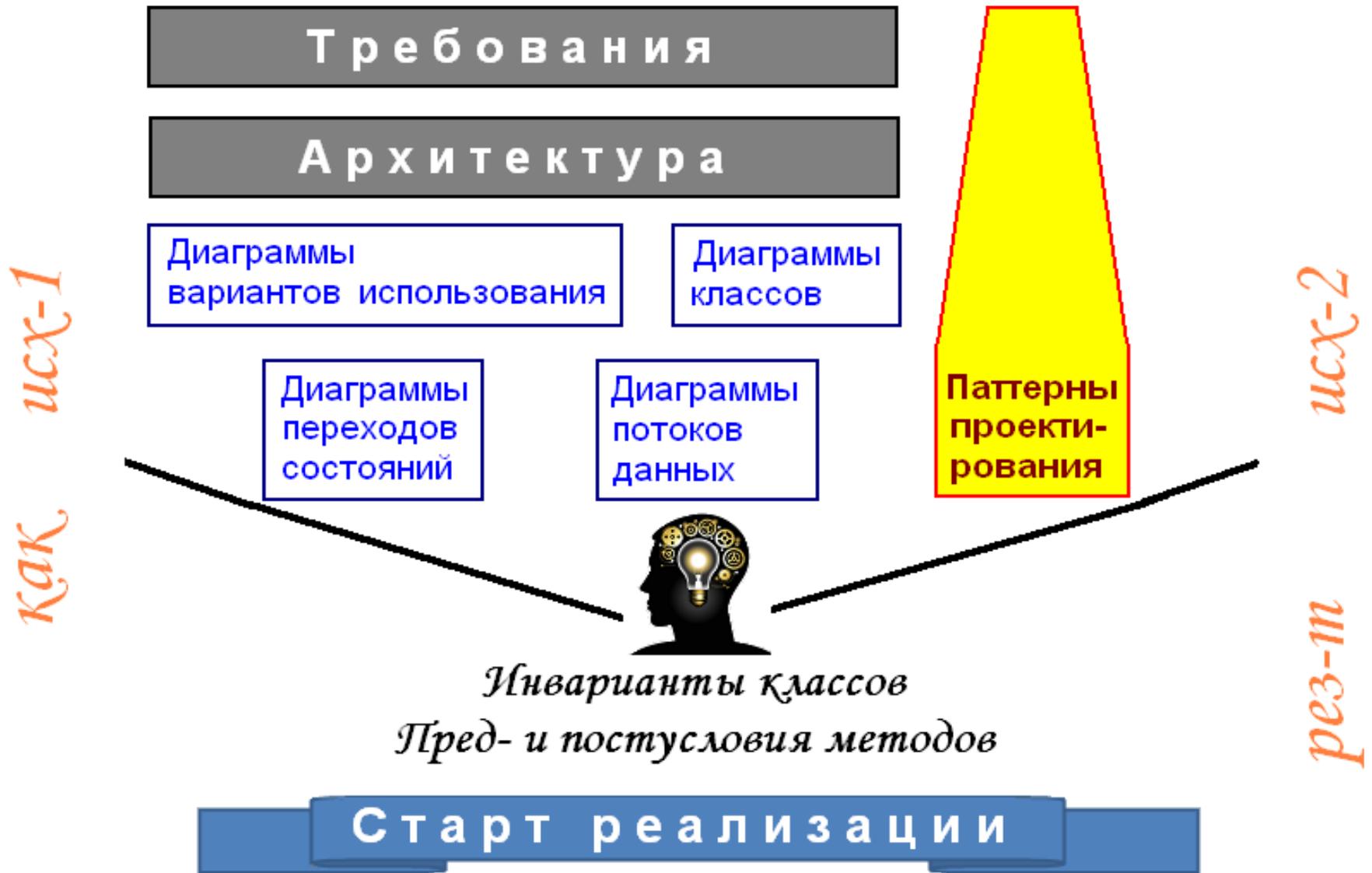
Структурная схема ПО

отражает состав и взаимодействие-по-управлению частей ПО

1С: Конфигуратор

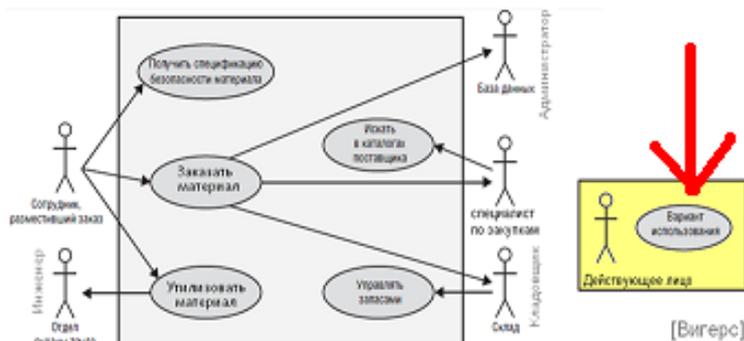


ООП: Детальное проектирование



исх-1: Напоминание

Диаграмма вариантов использования



Сотрудник → Вариант → Сотрудник'

Объектно-ориентированная модель
пример диаграммы классов



STD: Диаграммы переходов состояний

Состояние ≈ устойчивая характеристика объекта, свойственная определенному отрезку времени-его-существования.



Пример DFD: заказ + предоплата



Варианты применения

Тип варианта: **Основной** ✓ пользовательские требования

Наименование:

Цель:

Действующие лица:

Краткое описание:

Типичный ход событий: ✓ диалог действ.лица с ПО;
таблицы для выбора вариантов

Возможные альтернативы: ✓ нарушения типичного хода

Диagramмы
вариантов
использования

Тип варианта: **Вспомогательный** ✓ инсталляция и др.

Наименование:

Цель:

Действующие лица:

Краткое описание:

Тип варианта: **Дополнительный** ✓

Наименование:

Цель:

Действующие лица:

Краткое описание:



и др.

исх-2: Образцы = Шаблоны = Паттерны

- **архитектурные паттерны:**
 - GRASP = General Responsibility Assignment Software Pattern \approx паттерны распределения ответственности и др.
- **паттерны проектирования:**
 - GoF_{/23},
 - параллельного программирования и др.
- **паттерны анализа,**
- **паттерны тестирования,**
- **паттерны реализации.**

vs. Антипаттерны

– неэффективные подходы к решению

задач.

GoF = Gang of Four

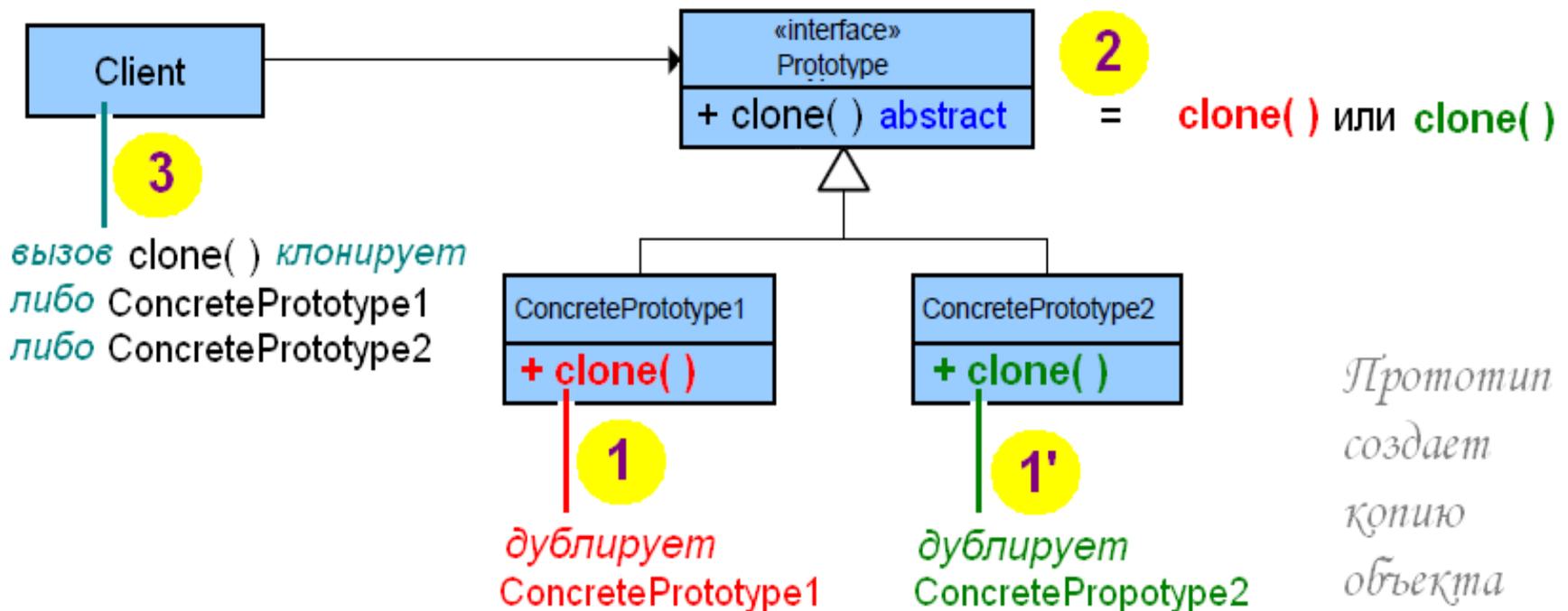
Э.Гамма, Р.Хелм, Р. Джонсон, Дж.Влиссидес (1994)

1. **Порождающие** паттерны проектирования /5
определяют способы создания объектов
2. **Структурные** паттерны проектирования /7
описывают способы конструирования сложных структур из классов и объектов
3. **Поведенческие** паттерны проектирования /11
описывают способы взаимодействия между объектами

Порождающие паттерны проектирования

1. Фабричный метод
2. Абстрактная фабрика
3. Одиночка
4. Строитель
5. Прототип ✓

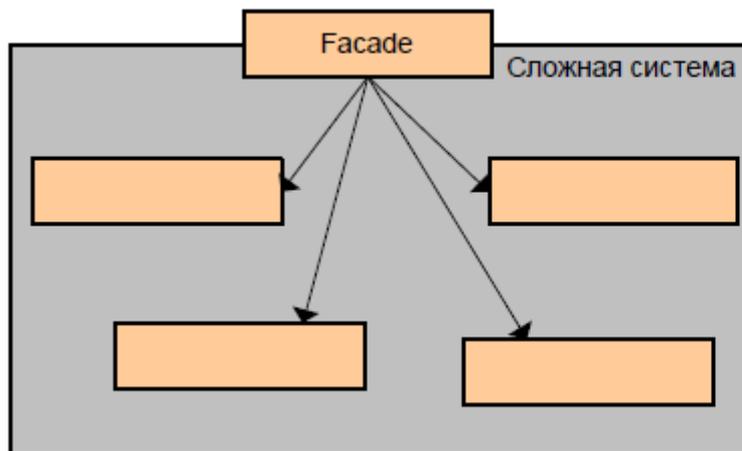
- Factory Method
- Abstract Factory
- Singleton
- Builder
- Prototype



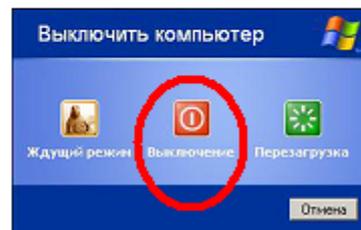
Структурные паттерны проектирования

1. Адаптер
2. Декоратор
3. Заместитель
4. Компоновщик
5. Мост
6. Приспособленец
7. Фасад

Adapter
Decorator
Proxy
Composite
Bridge
Flyweight
Facade



Единый интерфейс
вместо
набора интерфейсов

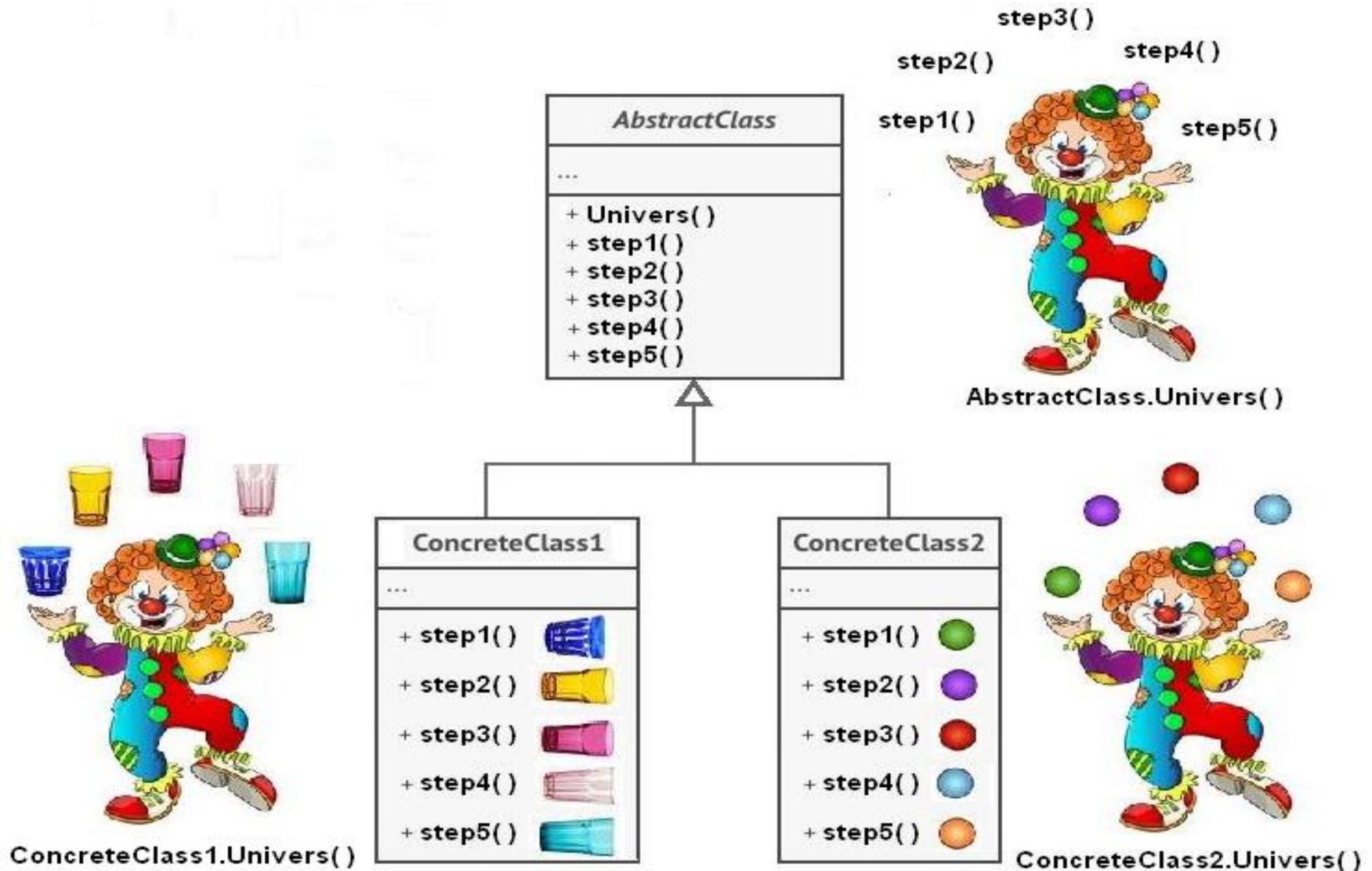


Поведенческие паттерны проектирования

- | | |
|--------------------------|-------------------------|
| 1. Интерпретатор | Interpreter |
| 2. Шаблонный метод ✓ | Template Method |
| 3. Итератор | Iterator |
| 4. Команда | Command |
| 5. Наблюдатель | Observer |
| 6. Посетитель | Visitor |
| 7. Посредник | Mediator |
| 8. Состояние | State |
| 9. Стратегия | Strategy |
| 10. Хранитель | Memento |
| 11. Цепочка обязанностей | Chain of Responsibility |

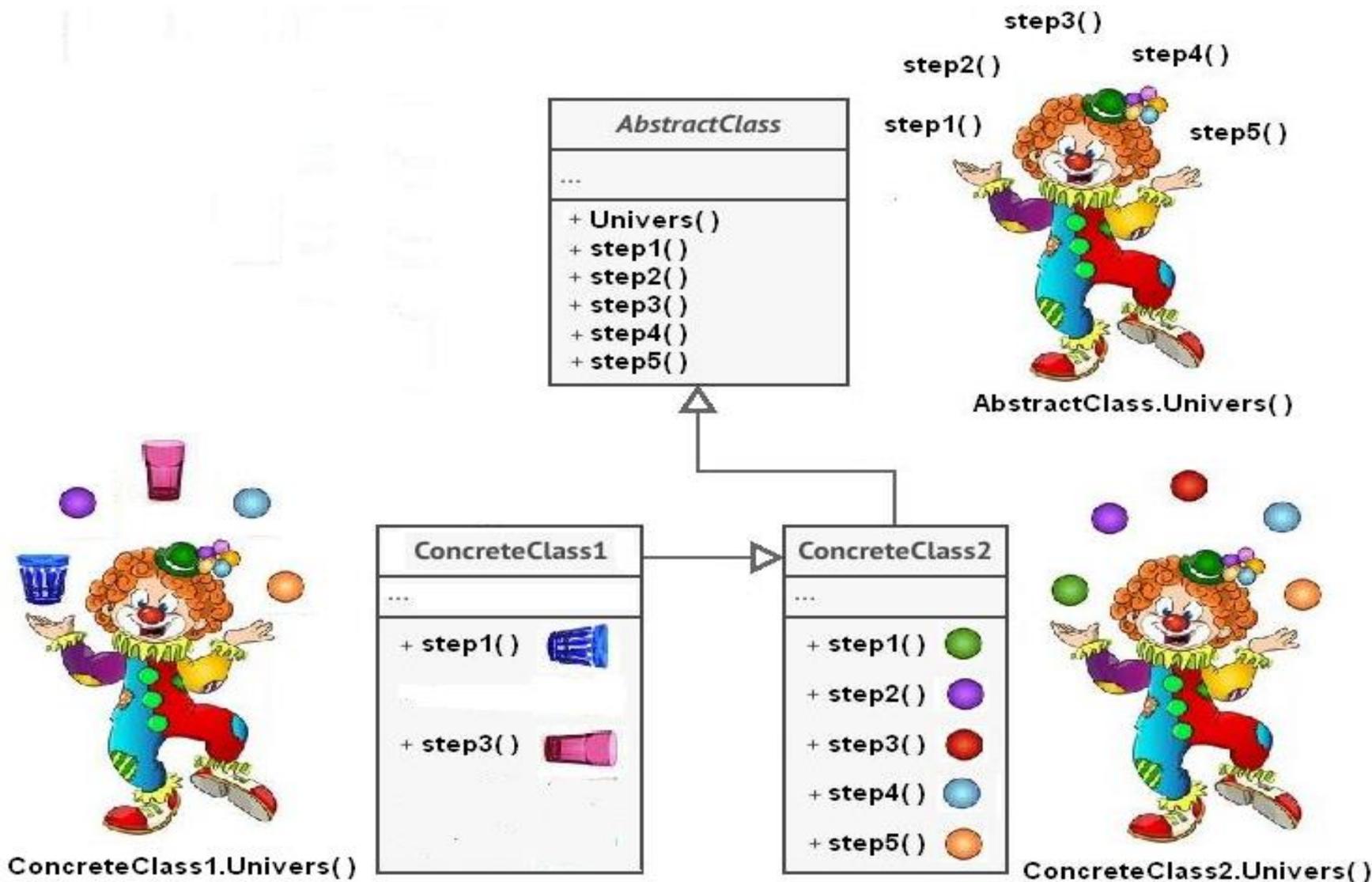
Поведенческий паттерн проектирования

“Шаблонный метод”



Поведенческий паттерн проектирования

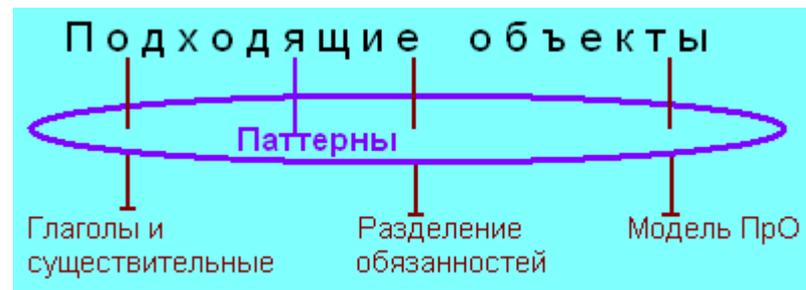
“Шаблонный метод” (продолжение)



как: Проектирование с помощью паттернов



1. Поиск подходящих объектов
разложить ПО на систему объектов.



2. Определение уровня детализации объектов.
3. Специфицирование интерфейсов объектов.
4. Специфицирование реализации объектов:
абстрактные, конкретные и т.п.
5. Механизмы повторного использования:
наследование, композиция, делегирование и т.п.
6. Проектирование с учетом будущих изменений,
освобождение от родительских классов, от конкретных операций и т.п.

[Гамма 2001]

рез-т: Инварианты классов

Пред- и постусловия методов

Инвариант класса – утверждение, истинное для любого объекта этого класса.

Методы класса должны сохранять инвариант.

Предусловие метода – утверждение, которое описывает состояние объекта и аргументов, подходящее для выполнения метода.

Постусловие метода – утверждение, которое описывает состояние объекта после выполнения метода.

В о п р о с ы?

soloviev@glossary.ru

Соловьев С.Ю.

Основы программной инженерии

www.park.glossary.ru