

ПРОЕКТИРОВАНИЕ ПРИ ОБЪЕКТНОМ ПОДХОДЕ

В основе - *объектная декомпозиция.*

Цель - *построить объектную модель предметной области задачи.*

Концептуальная основа (принципы):

 *Абстрагирование*

 *Ограничение доступа*

 *Модульность*

 *Иерархия*

 *Типизация*

 *Параллелизм*

 *Устойчивость*

(Первые четыре принципа являются обязательными)

1. *Абстрагирование*

Это процесс выделения **абстракций** в предметной области задачи.

Абстракция - это множество существенных характеристик некоторого объекта.

Множество должно:

- ✓ отличать данный объект от всех других видов объектов;
- ✓ четко определять особенности данного объекта.

Абстракции могут обладать:

- статическими свойствами;
- динамическими свойствами.

(Динамические свойства определяют поведение объекта в моделируемом пространстве).

2. Ограничение доступа

Это процесс защиты отдельных элементов объекта.

Процесс защиты не должен затрагивать существенных характеристик объекта как целого.

Ограничение доступа касается **внутренней структуры**.

3. Модульность

Обеспечивает возможностью **декомпозиции системы** на ряд отдельно реализуемых модулей.

(см. модульный подход)

4. Иерархия

Это ранжированная (упорядоченная) система абстракций.

Наследование свойств в иерархии - один из важнейших механизмов ООП.

Наследование - отношение между классами, которое дает возможность одному классу использовать структурную или функциональную часть другого или нескольких других классов.

(Другими словами, простое и множественное наследование).

5. Типизация

Это **ограничение**, накладываемое на класс объектов и **препятствующее** взаимозаменяемости объектов различных классов.

6. Параллелизм

Это свойство нескольких объектов одновременно находиться в **активном** или **пассивном** состоянии (реальный параллелизм, режим разделения времени).

7. Устойчивость

Это свойство объекта существовать во **времени*** и/или в **пространстве****.

*Вне зависимости от процесса, породившего объект.

**Перемещение объекта из одного адресного пространства в другое.

Унифицированный язык моделирования

UML (Unified Modeling Language)

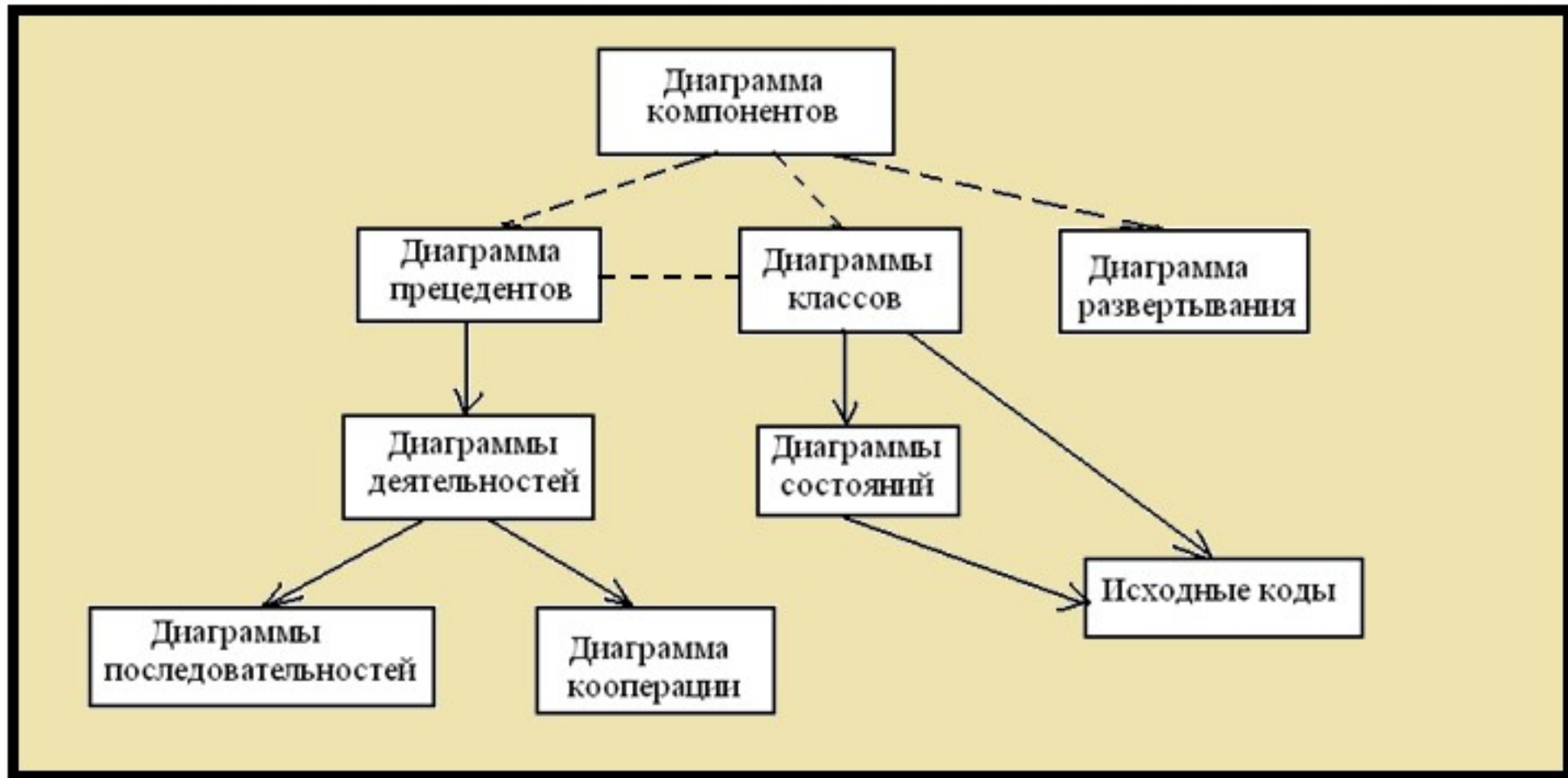
- Первая версия **создана в 1995 г.**
- **Это стандартное средство описания проектов.**
- **Используется для проектов на основе объектно-ориентированного подхода.**
- **Создатели (ведущие специалисты) Гради Буч, Ивар Якобсон и Джеймс Рамбо.**

При описании проекта используют:

- Описание функций;
- Описание вариантов использования;
- Диаграммы вариантов использования;
- Диаграммы деятельности;
- Диаграммы классов;
- Диаграммы последовательностей;
- Описание системных операций;
- Словарь терминов и др.

 Диаграммы вариантов использования и диаграммы классов являются **основными**.

 Диаграммы последовательностей и диаграммы деятельности используются **для уточнения** отдельных моментов поведения.



Диаграммы объектного подхода

Определение вариантов использования

Вариант использования представляет собой последовательность операций для достижения некоторого конечного результата конкретным пользователем.

Типы вариантов использования:

- ▽ *основные;*
- ▽ *вспомогательные;*
- ▽ *дополнительные.*

(Дополнительные являются необязательными, реализуются, **если** не требуют серьезных ресурсов).

Краткая форма

Название варианта	Результаты теста
Цель	<i>Просмотр результатов тестирования.</i>
Действующие лица	<i>Студент, преподаватель.</i>
Краткое описание	<i>Предполагается на экран в текстовой форме выдавать: ФИО, время тестирования, задаваемые вопросы, варианты ответов с указанием весов и суммарный балл.</i>
Тип	<i>Основной.</i>

Подробная форма

Дополнительно включает:

- описание типичного хода событий;
(Представляют в виде диалога между пользователями и системой. Шаги (события) нумеруют последовательно.)
- описание возможных альтернатив.
(альтернативы описывают в отдельных таблицах)

Оформление:

Таблица № X - Типичный ход событий

Действия исполнителя	Отклик системы
1. Событие 1	2. Событие 2
3. Событие 3	4. Событие 4

Таблица 1 - Типичный ход событий

Действия исполнителя	Отклик системы
1. Студент завершил тестирование	2. Вывод на экран первой страницы результатов
3. Выдать следующую страницу	4. Вывод следующей страницы
5. Закончить просмотр	6. Возврат в главное меню

Альтернативы:

- 1. Преподаватель выбирает соответствующий пункт в меню для просмотра результатов тестирования.*
- 4. Если была последняя страница, то предложить пользователю закончить просмотр.*

Диаграммы вариантов использования

- ❖ Представляет исходную концептуальную модель системы.
- ❖ Отображает функциональные требования.

Важное:

- ✓ В дальнейшем необходимо разработать ряд моделей.
(Цель - показать процесс проектирования и получить детализацию с разных сторон)
- ✓ Диаграмму можно строить во время изучения ТЗ.

Основные понятия:

- *действующее лицо (группа лиц);*
- *вариант использования (прецедент);*
- *связи (отношения);*
- *комментарии.*

Порядок построения диаграммы

Необходимо сделать следующее:

1. Выделить группы действующих лиц
(например, в зависимости от прав доступа);
2. Выделить как можно больше вариантов использования;
(**При условии, что** каждый вариант позволяет пользователю получить *значимые результаты* работы программы)
3. Дополнить прецеденты словесным описанием
(сценарием).
(Для каждого варианта использования необходимо выделить главную и альтернативную **последовательность**.
Необходимо **согласовать и уточнить** с заказчиком.)

Основные обозначения (UML)

Вариант использования

- Обозначается эллипсом.
- Внутри эллипса название (должно звучать как функция).

Должен соответствовать:

- ✓ **отдельному сервису** для пользователя;
- ✓ представлять собой **неделимую последовательность действий**.

Примечания

Обозначаются **прямоугольником с загнутым верхним правым углом**, внутри которого текст.



Виды отношений

- **Отношение ассоциации** – обозначается **сплошной линией** между сущностью и вариантом использования.
(Линия может иметь условное обозначение с указанием кратности - количества экземпляров).
- **Отношение расширения** – обозначается **пунктирной линией со стрелкой**.
 - Показывает связь между отдельным вариантом использования и более общим (соответственно и стрелка идет от отдельного к общему).
 - Линия помечается словом «extend» (расширяет).
 - Показывает факт, что один из вариантов использования может **присоединять к себе последовательность действий** другого варианта.

В частности:

- Один вариант использования может быть расширением для **нескольких базовых вариантов**.
- Базовый вариант использования в целом функционально **самодостаточен** (не зависит от своих расширений).
- **Отношение обобщения – обозначается сплошной линией со стрелкой (не закрашенный треугольник).**
 - Указывает, что один вариант использования (потомок) **наследует** все свойства и поведение своего родителя.
 - При этом потомок может быть дополнен новыми свойствами и **иметь несколько родителей**.

- **Отношение включения** - обозначается пунктирной линией со стрелкой, помеченной словом «include» (включает).

Показывает, что

один вариант использования включает **в качестве составного компонента** другой вариант.

Рассмотрим пример **диаграммы вариантов использования системы контроля знаний**.

Система контроля знаний

