UML

**UML** (англ. *Unified Modeling Language* — унифицированный язык моделирования),  язык графического описания для объектного моделирования в области разработки программного обеспечения.

**UML** — открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой *UML-моделью*.

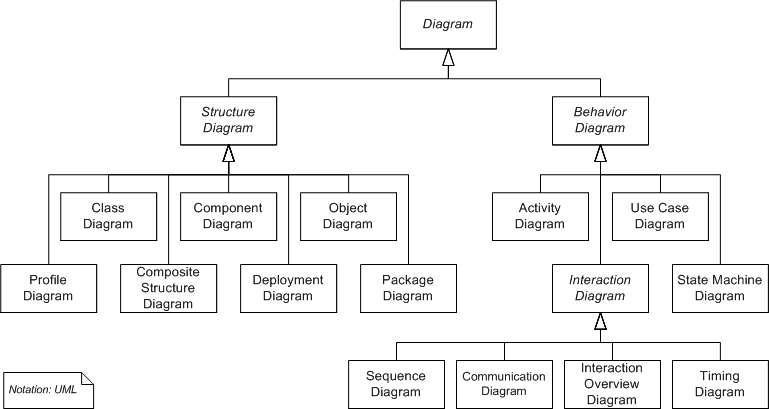
**UML** был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML-моделей возможна генерация кода.

 Его также используют для моделирования бизнес-процессов, системного проектирования и отображения организационных структур.

**UML** позволяет разработчикам программного обеспечения достигнуть соглашения в графических обозначениях для представления общих понятий (таких как класс, компонент, обобщение (англ. *generalization*), агрегация (англ. *aggregation*) и поведение) и больше сконцентрироваться на проектировании и архитектуре.

**UML 2.4.1** принят в качестве международного стандарта ISO/IEC 19505-1, 19505-2

Диаграммы можно разделить на такие основные типы: Структурные, Диаграммы поведения, диаграммы взаимодействия



Структурные диаграммы:

* Диаграмма классов
* Диаграмма компонентов
* [Диаграмма композитной/составной структуры](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%B7%D0%B8%D1%82%D0%BD%D0%BE%D0%B9_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B)
  + Диаграмма кооперации (UML2.0)
* Диаграмма развёртывания
* Диаграмма объектов
* Диаграмма пакетов
* Диаграмма профилей (UML2.2)

***Диаграмма классов*** (Class diagram) — статическая структурная диаграмма, описывающая структуру системы, демонстрирующая классы системы, их атрибуты, методы и зависимости между классами.

***Диаграмма компонентов*** (Component diagram) — статическая структурная диаграмма, показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами. В качестве физических компонентов могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты

***Диаграмма композитной/составной структуры***

Шаблон проектирования Декоратор на диаграмме кооперации

Диаграмма композитной/составной структуры (Composite structure diagram) — статическая структурная диаграмма, демонстрирует внутреннюю структуру классов и, по возможности, взаимодействие элементов (частей) внутренней структуры класса.

***Диаграмма объектов***

*Диаграмма объектов* (Object diagram) — демонстрирует полный или частичный снимок моделируемой системы в заданный момент времени. На диаграмме объектов отображаются экземпляры классов (объекты) системы с указанием текущих значений их атрибутов и связей между объектами

***Диаграммы поведения:***

*-* [Диаграмма деятельности](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_%D0%B4%D0%B5%D1%8F%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D0%B8)

- [Диаграмма состояний](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D0%B9_(UML))

- [Диаграмма вариантов использования](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D0%BE%D0%B2)

***Диаграмма деятельности*** (Activity diagram) — диаграмма, на которой показано разложение некоторой[*деятельности*](https://ru.wikipedia.org/w/index.php?title=%D0%94%D0%B5%D1%8F%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%8C_(UML)&action=edit&redlink=1) на её составные части.

Диаграммы деятельности используются при моделировании бизнес-процессов, технологических процессов, последовательных и параллельных вычислений.

**Диаграммы взаимодействия:**

Диаграмма коммуникации (UML2.0) / Диаграмма кооперации (UML1.x)

Диаграмма обзора взаимодействия (UML2.0)

Диаграмма последовательности

Диаграмма синхронизации (UML2.0)\

# **Диаграмма классов**

1 Взаимосвязи

* 1.1 Ассоциации
  + 1.1.1 Агрегация
  + 1.1.2 Композиция
  + 1.1.3 Различия между композицией и агрегацией
* 1.2 Обобщение (наследование)
* 1.3 Реализация
* 1.4 Зависимость
* 1.5 Уточнения отношений

***Ассоциация***показывает, что объекты одной сущности (класса) связаны с объектами другой сущности

***Агрегация*.** Простая ассоциация между двумя классами отражает структурное отношение между равноправными сущностями,

***Композиция***— более строгий вариант агрегации. Известна также как агрегация по значению.

**Обобщение (Generalization)** показывает, что один из двух связанных классов (*подтип*) является частной формой другого (*надтипа*), который называется **обобщением** первого. На практике это означает, что любой экземпляр подтипа является также экземпляром надтипа.

**Реализация** — отношение между двумя элементами модели, в котором один элемент (*клиент*) реализует поведение, заданное другим (*поставщиком*). Реализация — отношение целое-часть. Графически реализация представляется так же, как и наследование, но с пунктирной линией.