Роли объектов

* Актер - не разрешать влиять на данные
* Сервер – объект представляющий ресурсы, которые подлежать воздействию
* Агент – Способен как подвергаться воздействию, так и подвергаться воздействием на другие объекты. Является посредником между актером и сервером

Для определения спецификации в объектном подходе используют UML. Является стандартным средством описания проекта, и описателем для ООП. Любая сложная система может быть описана через совокупность моделей. Выделяют концептуальную и физическую модель системы. Концептуальная модель определяет логическое представление системы, т.е. определение правил работы системы конечного пользователя с описанием внешних и внутренних структурных отношений и представление системы во время ее функционирования с учетом масштабируемости отдельных компонентов и производительности отдельных узлов.

Физическая модель определяется через реализацию и размещение компонентов на физическом устройстве. За концептуальную отвечает конечный пользователь. За реализацию – программист, за размещение – системный администратор.

Для описания моделей в UML введен набор предметов. Предмет – абстракция, являющаяся основным элементом модели, которая взаимодействует через отношение и объединяется через диаграмму. В UML существует 4 разновидности предмета:

* Структурные. Являются существительными в UML моделях, представляют статическую часть их существует 8:
* Класс. Представляет описание множества объектов разделяющий одинаковые свойства, выполняющие одинаковые операции и общую семантику. Имя, набор свойств и операций. Способен реализовать различные интерфейсы. Интерфейс – набор операций, который определяет услуги класса или компонента. Он описывает поведение компонента, видимого из вне. Интерфейс описывает полные или частичный набор услуг
* Кооперация. Определяет взаимодействие совокупности ролей одного или нескольких объектов. Обычно реализуется через шаблоны
* Актер. Набор согласованных ролей, которые может играть пользователь при взаимодействии с системой. Каждая роль требует от системы соответствующего поведения.
* Прецедент. Определяет описание последовательности действий, выполняемых системой в интересах отдельного автора. Прецедент создает видимый для автора результат.
* Компонент. Физическая часть системы, соотвествующая набору интерфейсов, чаще всего реализовано в виде отдельного файла. Компоненты существуют различного рода: библиотечные, исполняемые и тд
* Узел. Физический элемент, который существует в процессе работы системы, представляющий собой ресурс для выполнения ПО. Бывают узлы с памятью и без.
* Поведения.
* Взаимодействия. Набор сообщений, которыми обмениваются объекты в конкретном контексте для достижения определенной цели. Взаимодействие определяется самим сообщением, последовательностью действий и связи между объектами.
* Конечный автомат. Определяет последовательные состояния объекта, в которые он переходит при возникновения тех или иных событий, элементы являются: переходы, состояния и события.
* Группирующие. Организационная часть UML, по сути пакет документов, описывающий интерфейс системы
* Поясняющие. Разъясняющая часть UML, проявляющаяся в виде описания, комментария и тд

В UML есть 4 типа отношений:

* Отношение зависимость. -----🡪 . Семантическое отношение между 2 предметами, при котором изменение в одном предмете влияет на семантику в другом предмете. Тот который влияет – независимы, а наоборот зависимый
* Ассоциация. Структурные отношение, которое описывает набор связей, соединяющих 2 объекта. Они именуются и ассоциируются.
* Обобщение. Отношение специализации или обобщения, в котором объекты специализированного элемента могут объединять объектами общего.
* Отношение реализации. Семантическое отношение, применяемое для описания реализации отдельных интерфейсов между предметами.

Используя предметы и решения разрабатываются диаграммы. Все диаграммы делят:

* Диаграммы поведения:
* Диаграмма состояния объектов. Моделирует поведение объекта при переходе из одного состояния в другое. Состояния ассоциируются с набором выполняемых действий:
* Простые
* Составные. Включают 3 вида действия: при в ходе, в процессе нахождения и действие перед выходом
* Диаграмма деятельности. Моделирует поведение системы в рамках различных вариантов использования. Для каждого актера приводится плавательная дорожка. Они имеют линии синхронизации, которые определяют начало или окончания действий отдельных актеров.
* Последовательности. Характеризует набор действий выполняемыми объектами системы, развернутых во времени. Элементами диаграммы последовательности являются объекты с их временем жизни. Запуск каждого объекта осуществляется либо внешним воздействием, либо сообщением другого объекта.
* Кооперации. Определяет процесс обмена сообщениями между объектами.
* Статические
* Диаграмма классов. Для моделирования структуры для определения связи между элементами
* Вариантов использования. Диаграмма вариантов использования применяется для моделирования бизнес процессов и выявления требований к системе. Элементами являются:
* Авторы
* Прецеденты

Авторы и прецеденты связанны различными видами отношений.

* Диаграммы реализации и развертывания.
* Развертывания. Определяет топологию системы. Показывает набор узлов с размещаемыми на них компонентами.
* Компонентов. Определяется набор интерфейсов и модулей из которых она состоит