**ДИАГРАММА ОБЪЕКТОВ**

Представляет собой диаграмму состояний объектов.

На диаграмме объектов отношения между объектами унаследуются от диграммы классов.

Примеры:

* Online compiler
  + классы: LexicalAnalyzer, SyntaxAnalyzer, Lexeme
  + LexicalAnalyzer принимает исходную строку, разбивает на лексемы
  + SyntaxAnalyzer принимает набор из лексем
* Facebook стена
  + классы: User, Post, Comments
  + пусть post имеет владельца user, контент content, оставленный комментарий пользователем user2
* Git Remote Repo
  + классы: Repository, User, Branch
  + пусть пользователь admin создаёт ветку dev из ветки master. Владельцем ветки dev теперь является admin.

**ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТЕЙ**

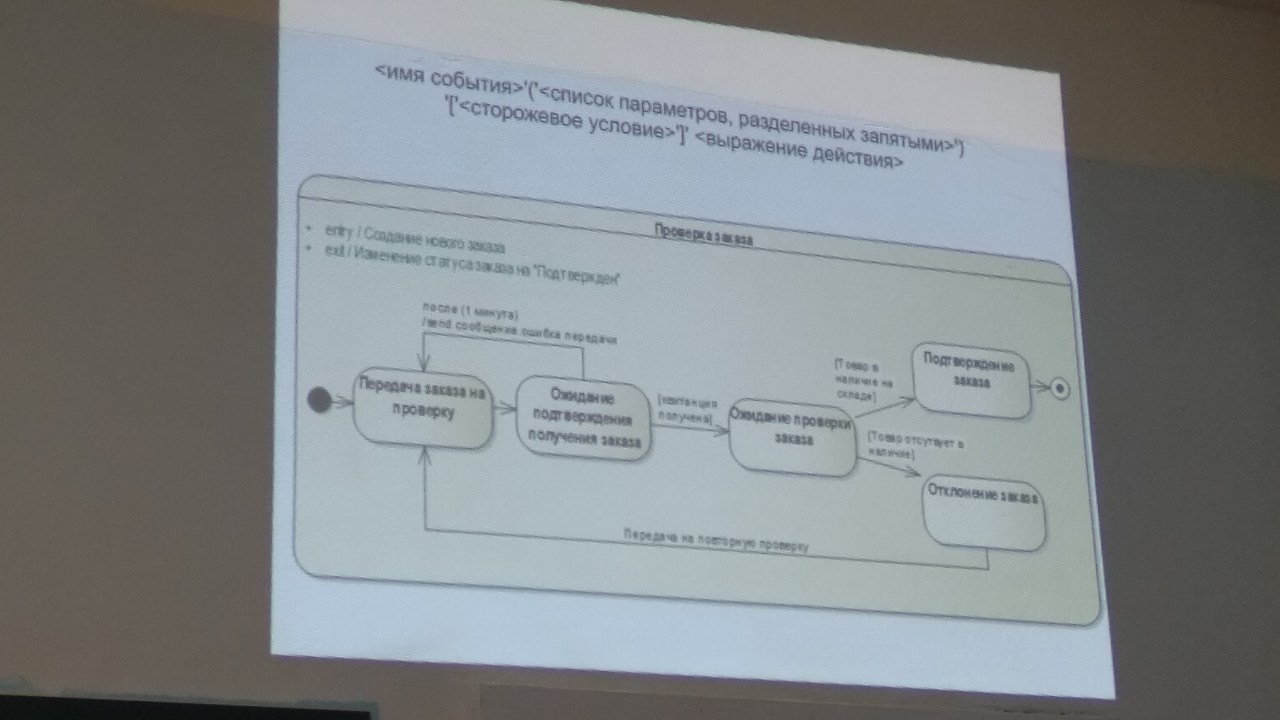
Линия жизни - отображает объект взаимодействия класса анализа, но не указывает на конкретный экзепляр класса.

Между линиями жизни пересылаются сообщения, которые являются типами коммуникации; могут совмещать в себе:

* вызов метода
* создание/уничтожение экземпляра
* подача сигнала

**ДИАГРАММА СОСТОЯНИЙ**

Моделирует ситуацию (часто неявно), когда выполняется некоторые постоянные условия, тем самым представляет статическую ситуацию, например, когда объект ждёт некоторого события. Состояние также может моделировать динамические условия, например, выполнение некоторого процесса. В таком случае элемент попадает в состояние начала процесса и выходит из состояния, когда процесс завершнен.



События:

* call event
* signal event
* time event
* change event

**ПСЕВДОСОСТОЯНИЯ**