

Проектирование ПО

Лекция 6: Моделирование поведения

Тимофей Брыксин
timofey.bryksin@gmail.com

Модели поведения

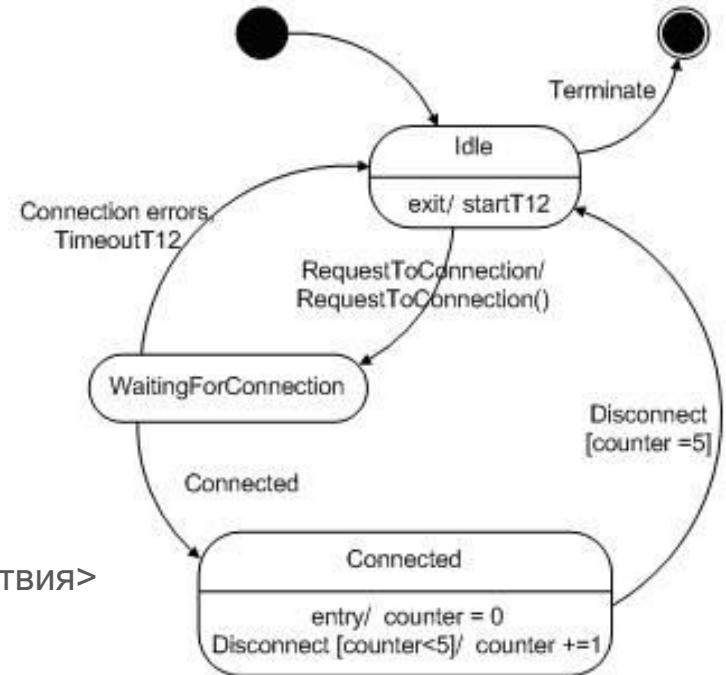
- детальность
- компактность
- независимость от оборудования и технологий
- популярные нотации
- непротиворечивость здравому смыслу

Виды моделирования поведения

- описание поведения с явным выделением состояний
- описание поведения с явным выделением потоков данных и управления
- описание поведения как последовательности сообщений во времени
- описание параллельного поведения

Диаграммы конечных автоматов

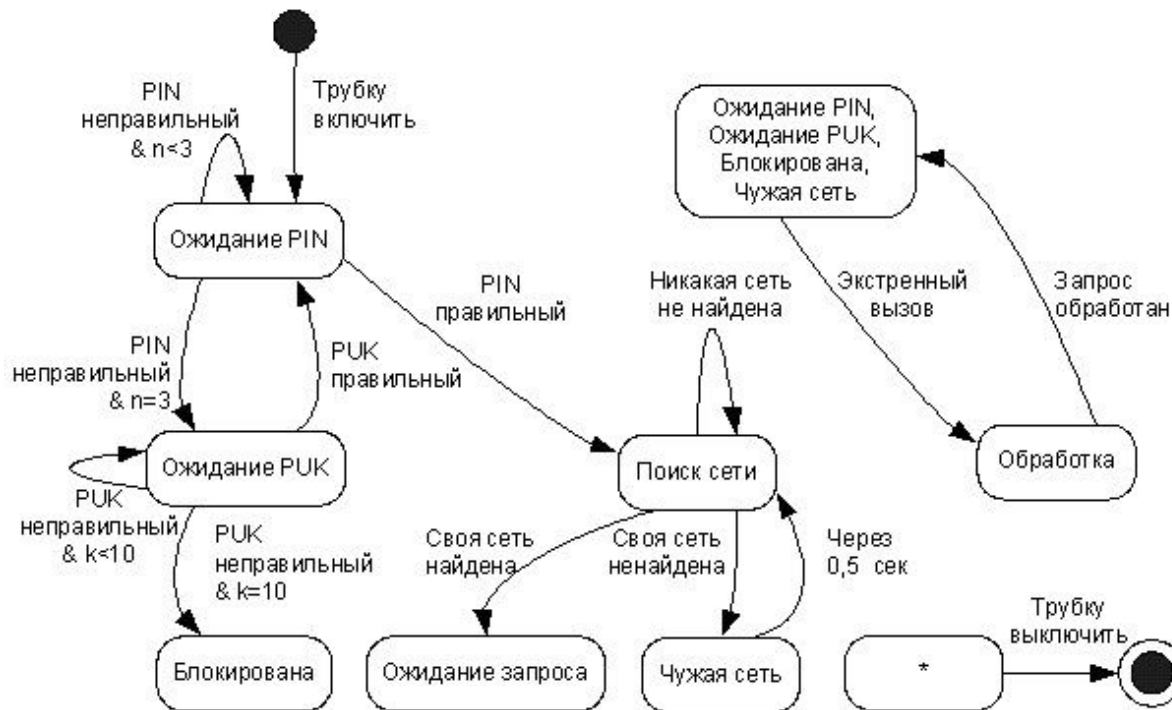
- состояния объекта как часть жизненного цикла
- моделирование реактивных объектов
- основные элементы
 - состояние
 - entry activity
 - exit activity
 - do activity
 - внутренний переход
 - событие
 - переход
 - <имя события> (<список параметров>)
 - [<сторожевое условие>] <выражение действия>



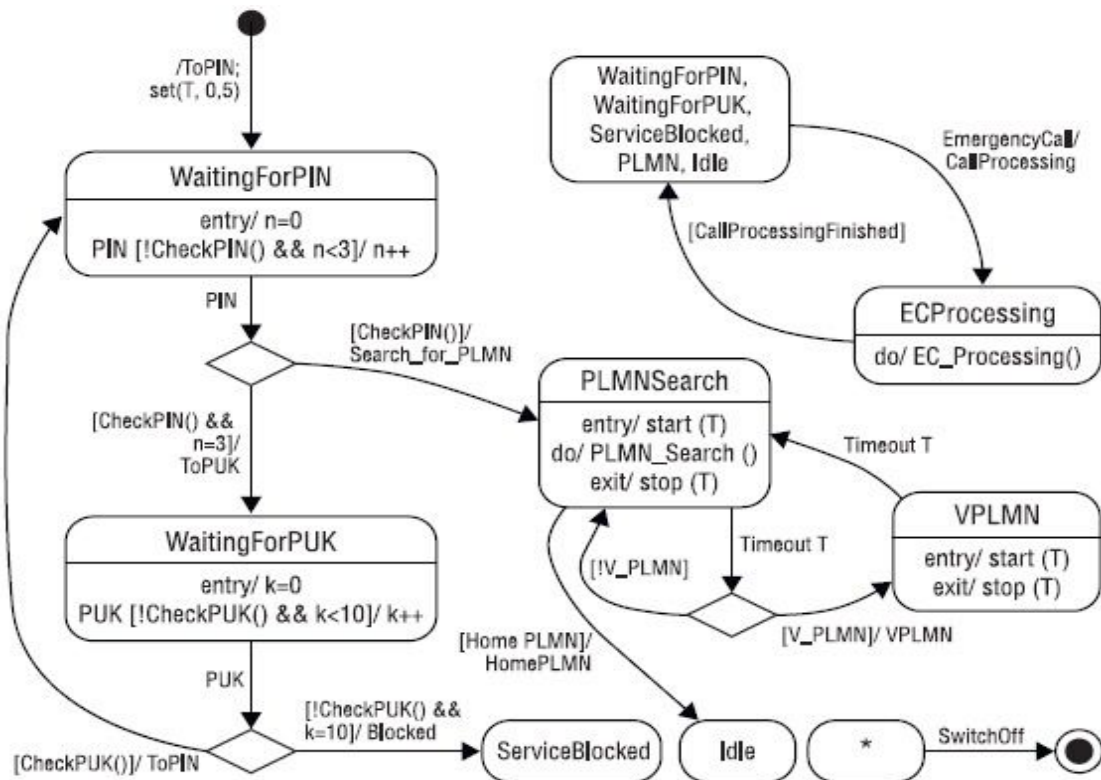
Псевдосостояния



Пример



Пример: уточнение



Генерация кода

```
void processStateMachine()
{
    bool finish = false;
    SendMessage(ToPin);
    set (T, 0.5);
    state = WatingForPIN;
    while (!finish)
        if (getEvent())
            finish = processEvents();
}
```

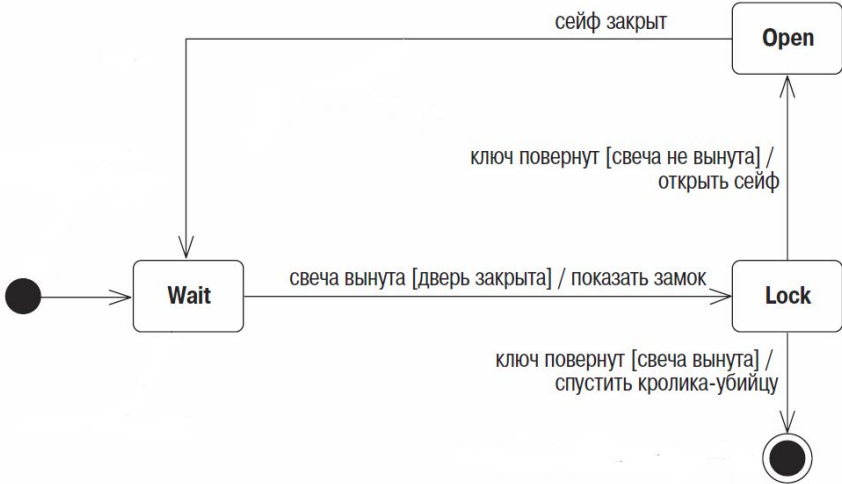
```
bool processEvents(){
    bool f = false;
    switch (state){
    case WatingForPIN:
        if (nextstate) n = 0;    nextstate = true;
        if (input_message == PIN) {
            if (CheckPIN()) {
                SendMessage(Search_for_PLMN);
                state = PLMNSearch;
            } else if (n == 3) {
                SendMessage(ToPUK);
                state = WatingForPUK;
            } else { n++; nextstate = false; }
        }
    }
}
```

...

```
} else if (input_message == EmergencyCall) {
    SendMessage(CallProcessing);
    prevstate = state;
    state = ECProcessing;
} else if (input_message == SwitchOff) {
    f = true;
}
break;
case WatingForPUK:
    if (nextstate) k = 0;
    nextstate = true;
    ...
}

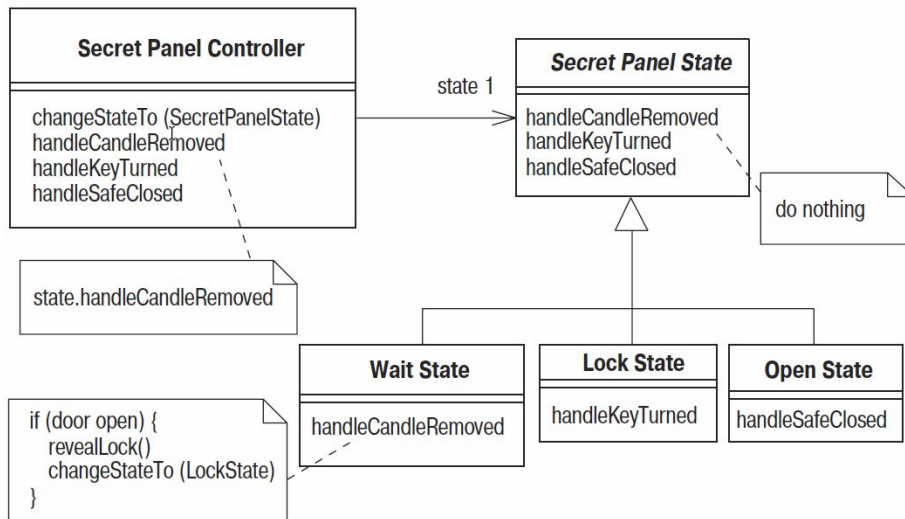
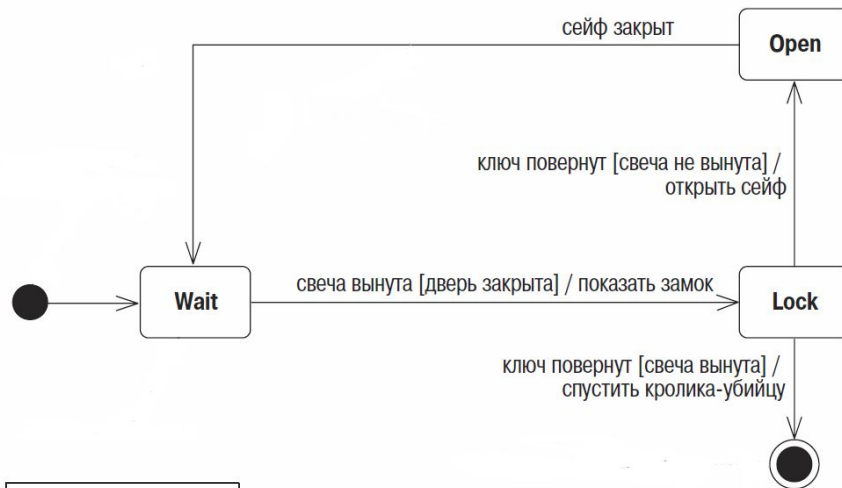
return f;
}
```


Таблица переходов



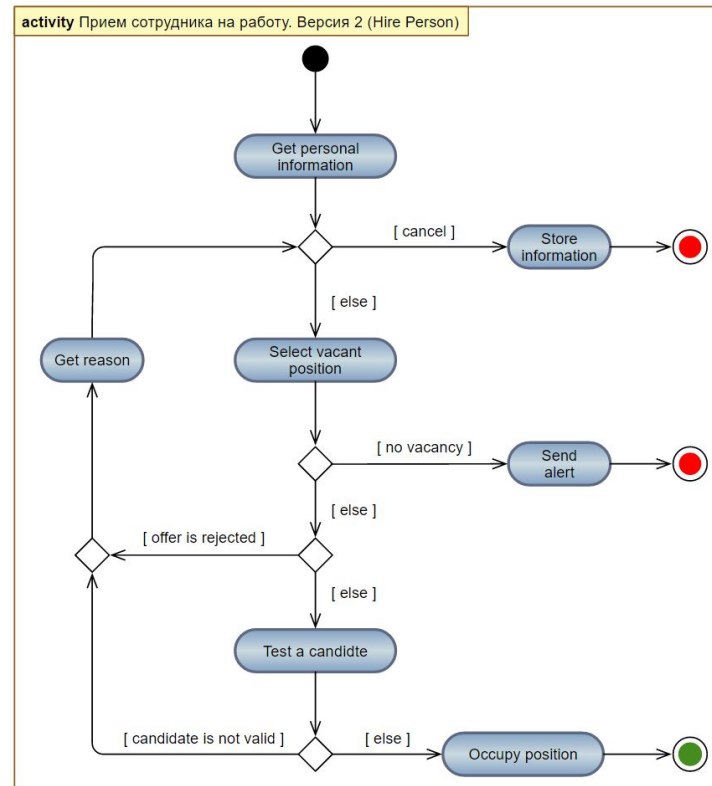
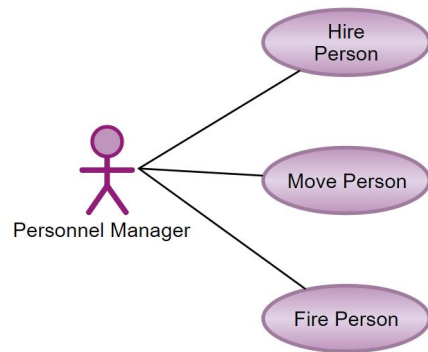
Исходное состояние	Целевое состояние	Событие	Защита	Процедура
Wait	Lock	Candle removed (свеча удалена)	Door open (дверца открыта)	Reveal lock (показать замок)
Lock	Open	Key turned (ключ повернут)	Candle in (свеча на месте)	Open safe (открыть сейф)
Lock	Final	Key turned (ключ повернут)	Candle out (свеча удалена)	Release killer rabbit (освободить убийцу-кролика)
Open	Wait	Safe closed (сейф закрыт)		

Паттерн “Состояние”



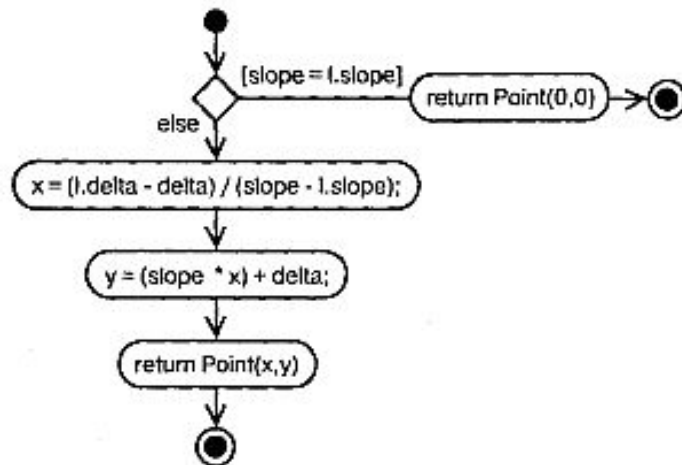
Моделирование потока управления и данных

- Диаграммы активности



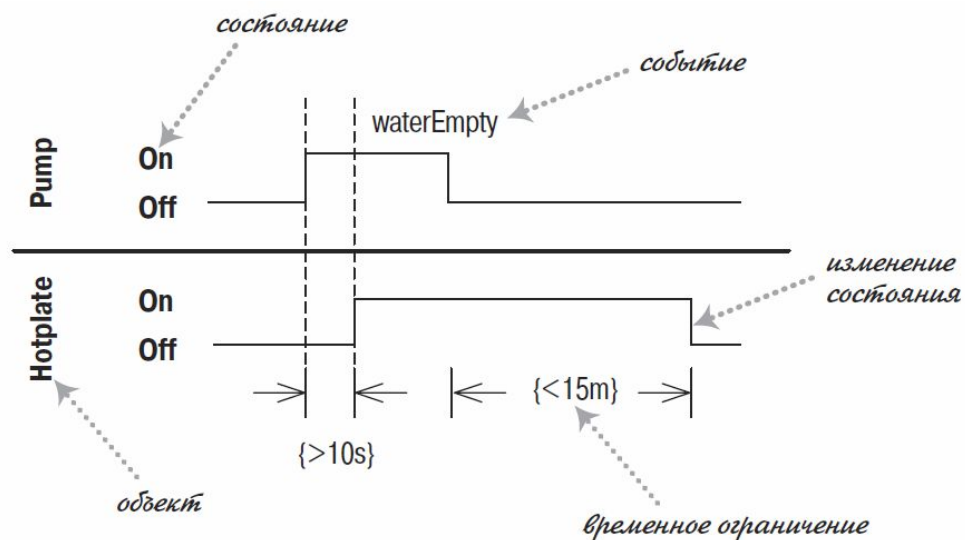
Диаграммы активности

- визуализация, специфицирование, конструирование и документирование поведения
- одна диаграмма -- один аспект поведения
- не стоит перегружать деталями
 - особенно кодом
- формализация -- сети Петри



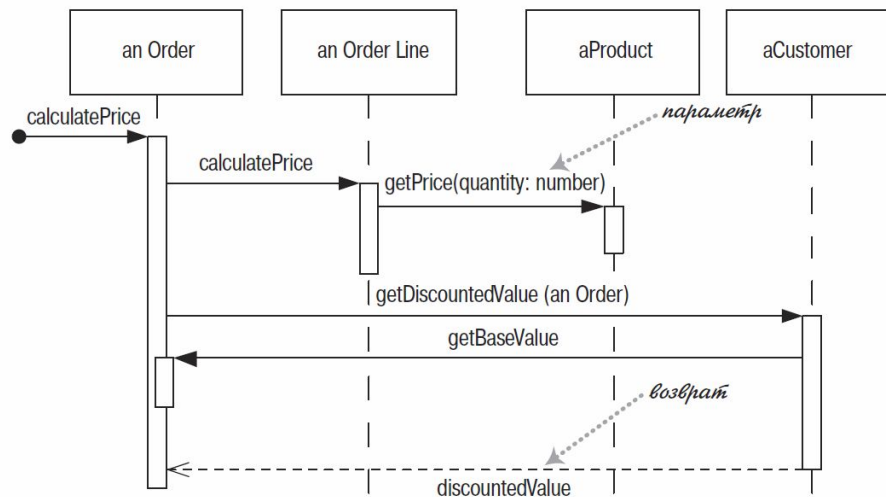
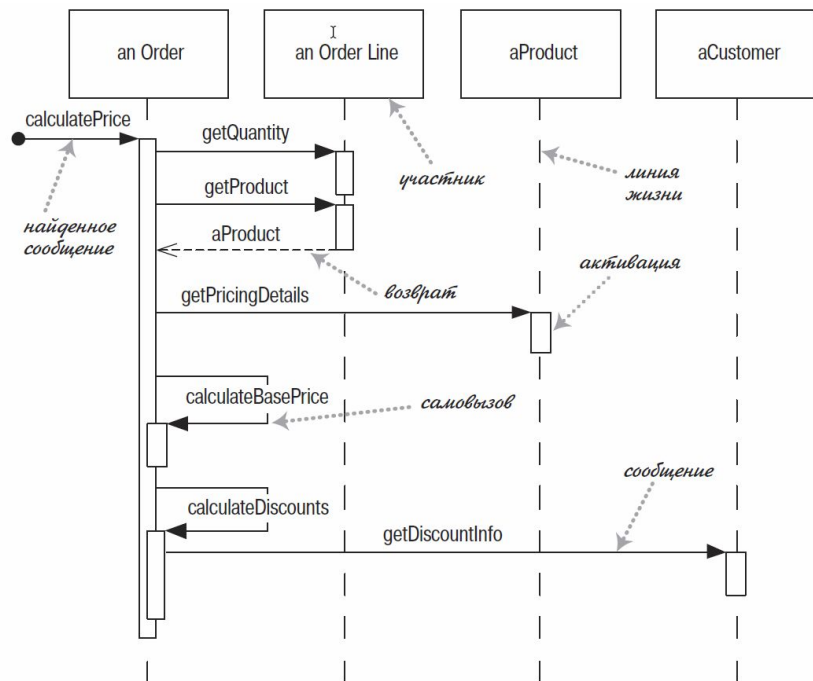
Временные диаграммы

- изменение состояний объектов во времени

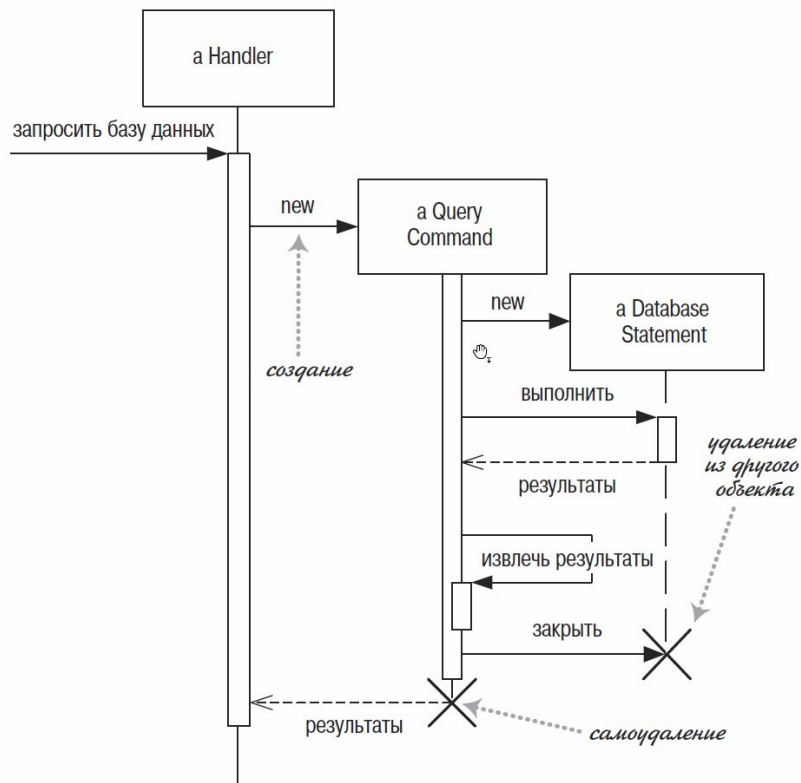


Моделирование последовательности сообщений

- Диаграммы последовательностей

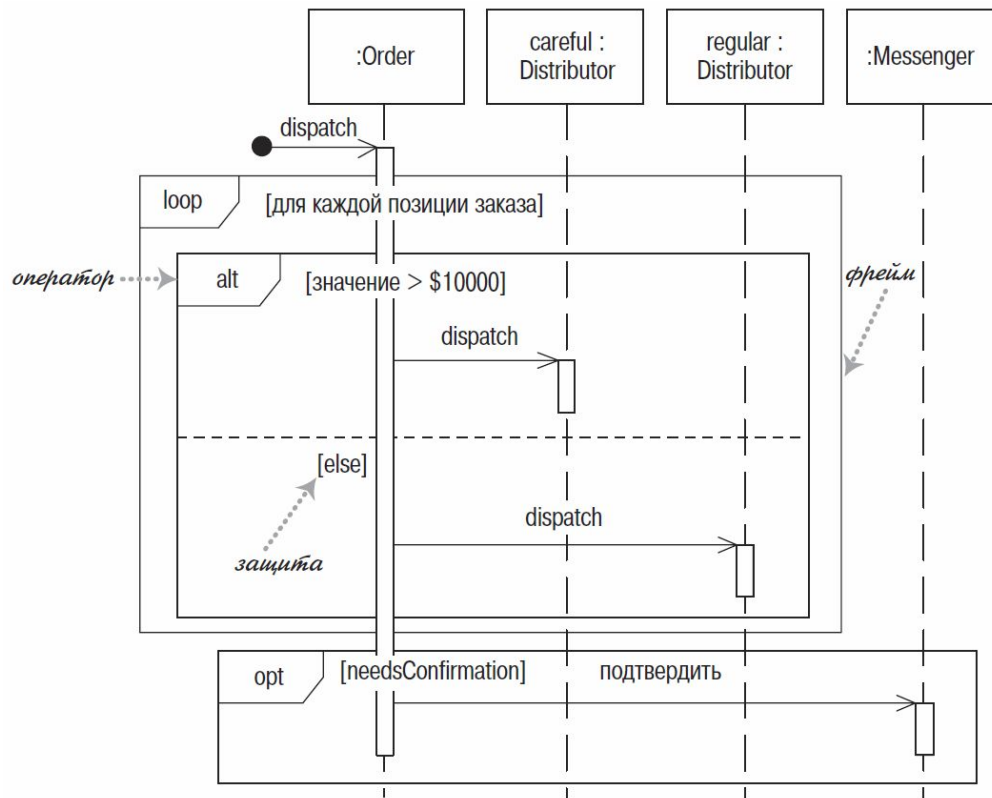


Создание и удаление объектов

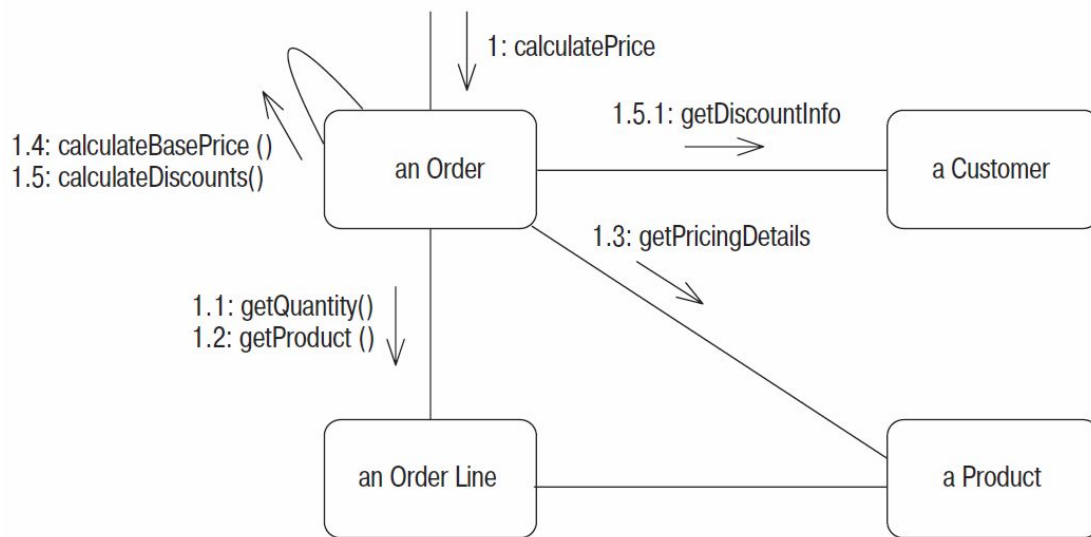


Циклы и условия

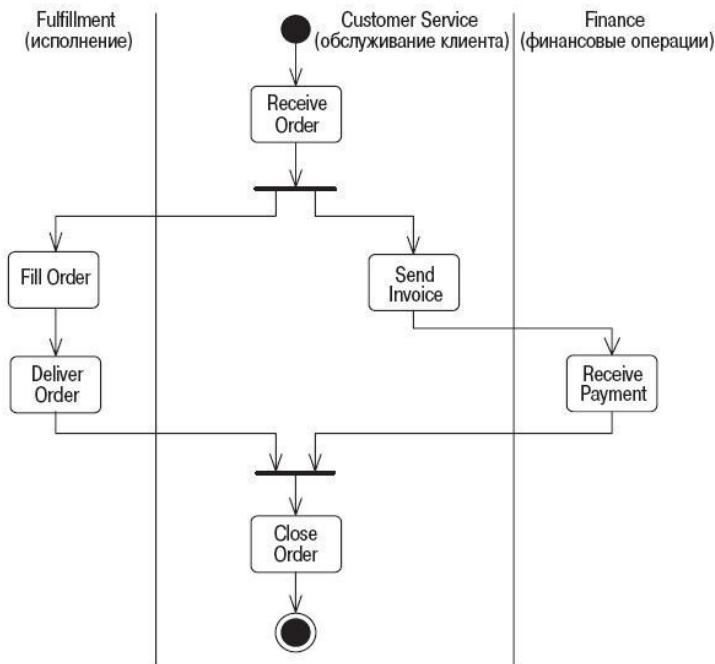
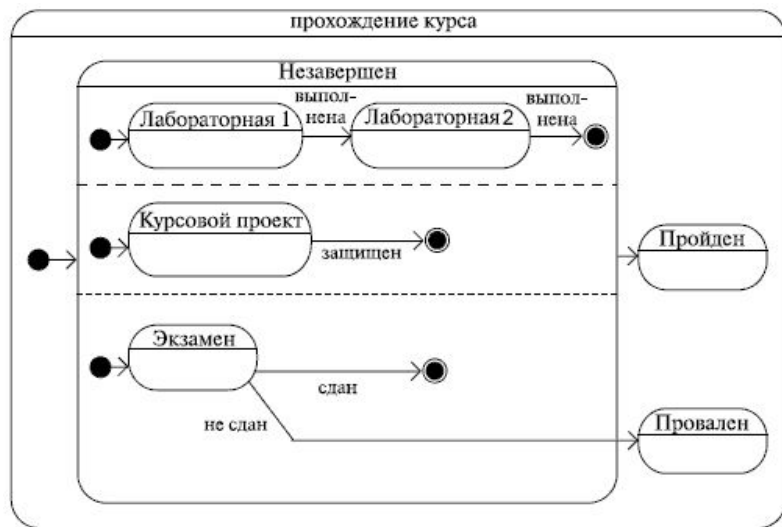
```
foreach (lineitem)
  if (product.value > $10K)
    careful.dispatch
  else
    regular.dispatch
  end if
end for
if (needsConfirmation)
  messenger.confirm
```



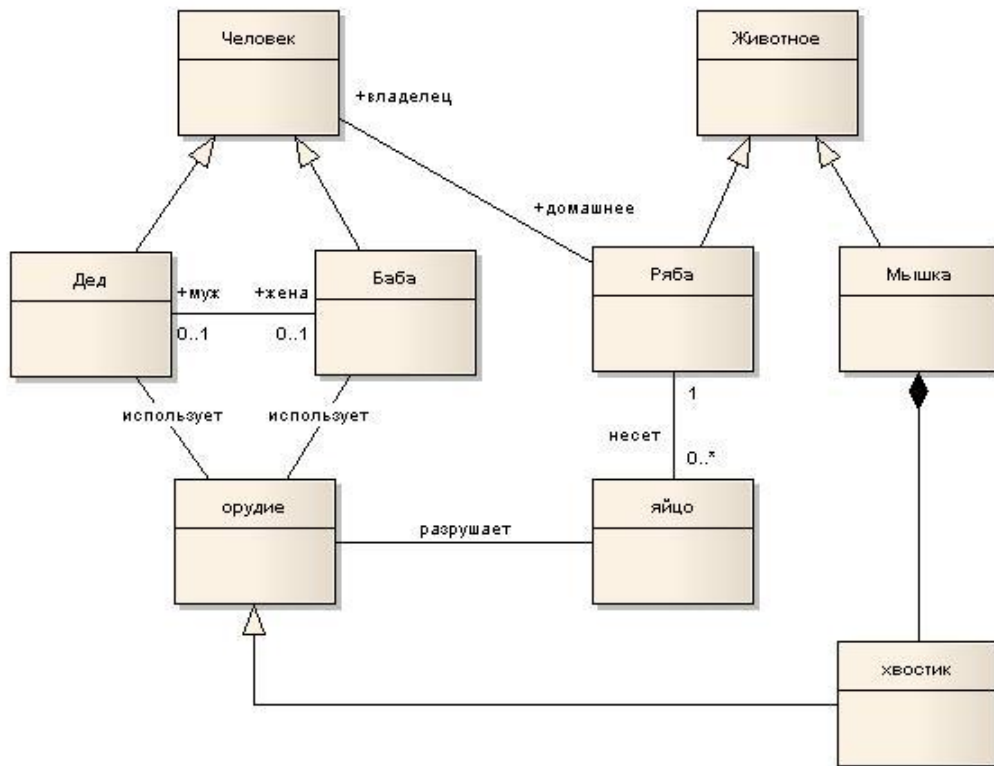
Коммуникационные диаграммы



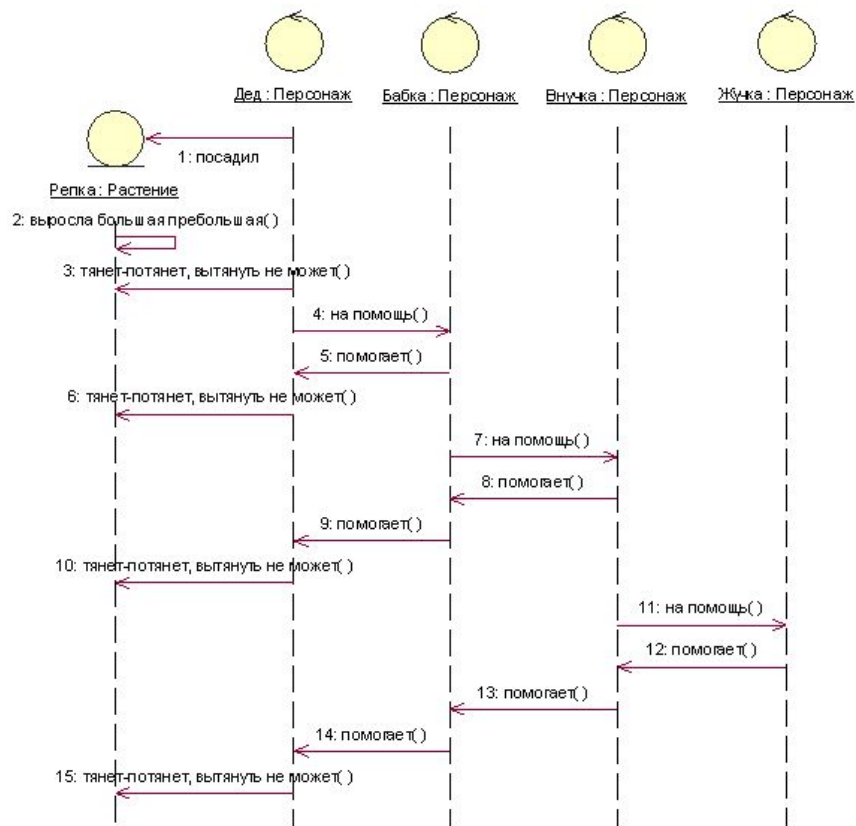
Моделирование параллельного выполнения



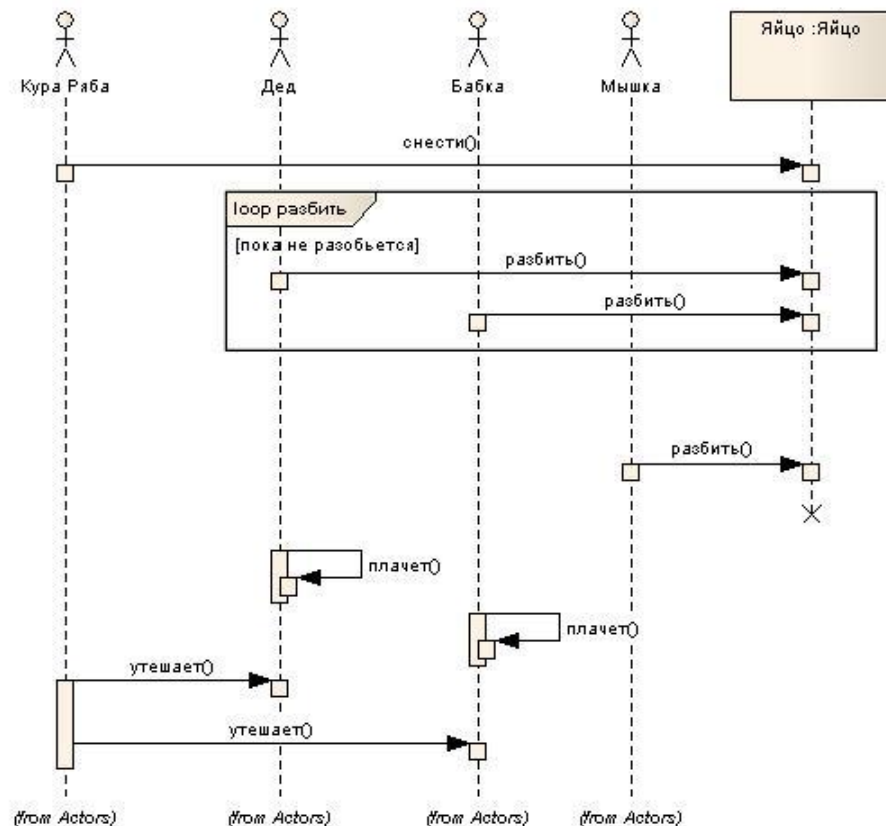
Примеры диаграмм UML: 1



Примеры диаграмм UML: 2



Примеры диаграмм UML: 3



Примеры диаграмм UML: 4

