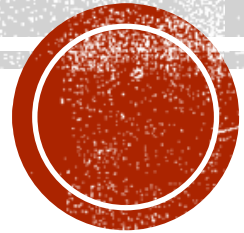
UML

Behavior Diagrams



БГТУ – 2020

лектор: Парамонов А.И.

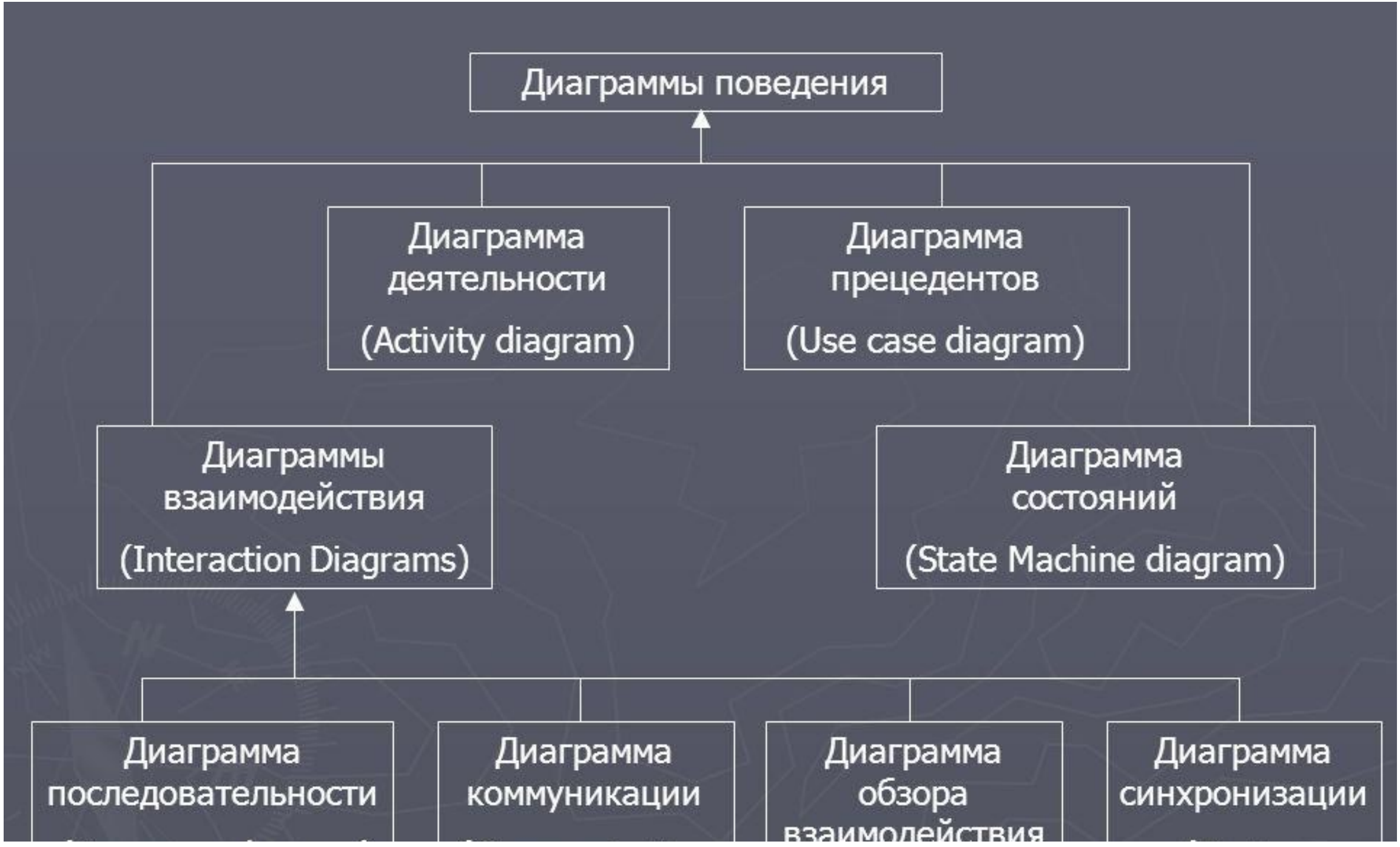
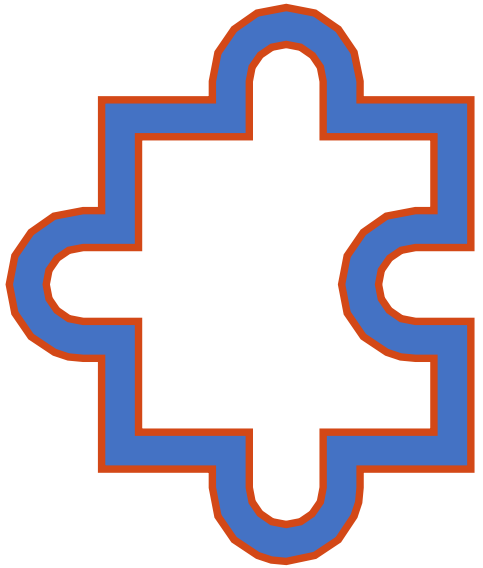

Диаграммы поведения
Диаграмма деятельности
(Activity diagram)
Диаграмма прецедентов
(Use case diagram)
Диаграммы взаимодействия
(Interaction Diagrams)
Диаграмма состояний
(State Machine diagram)
Диаграмма последовательности
Диаграмма коммуникации
Диаграмма обзора взаимодействия
Диаграмма синхронизации

ДИАГРАММА ДЕЯТЕЛЬНОСТИ

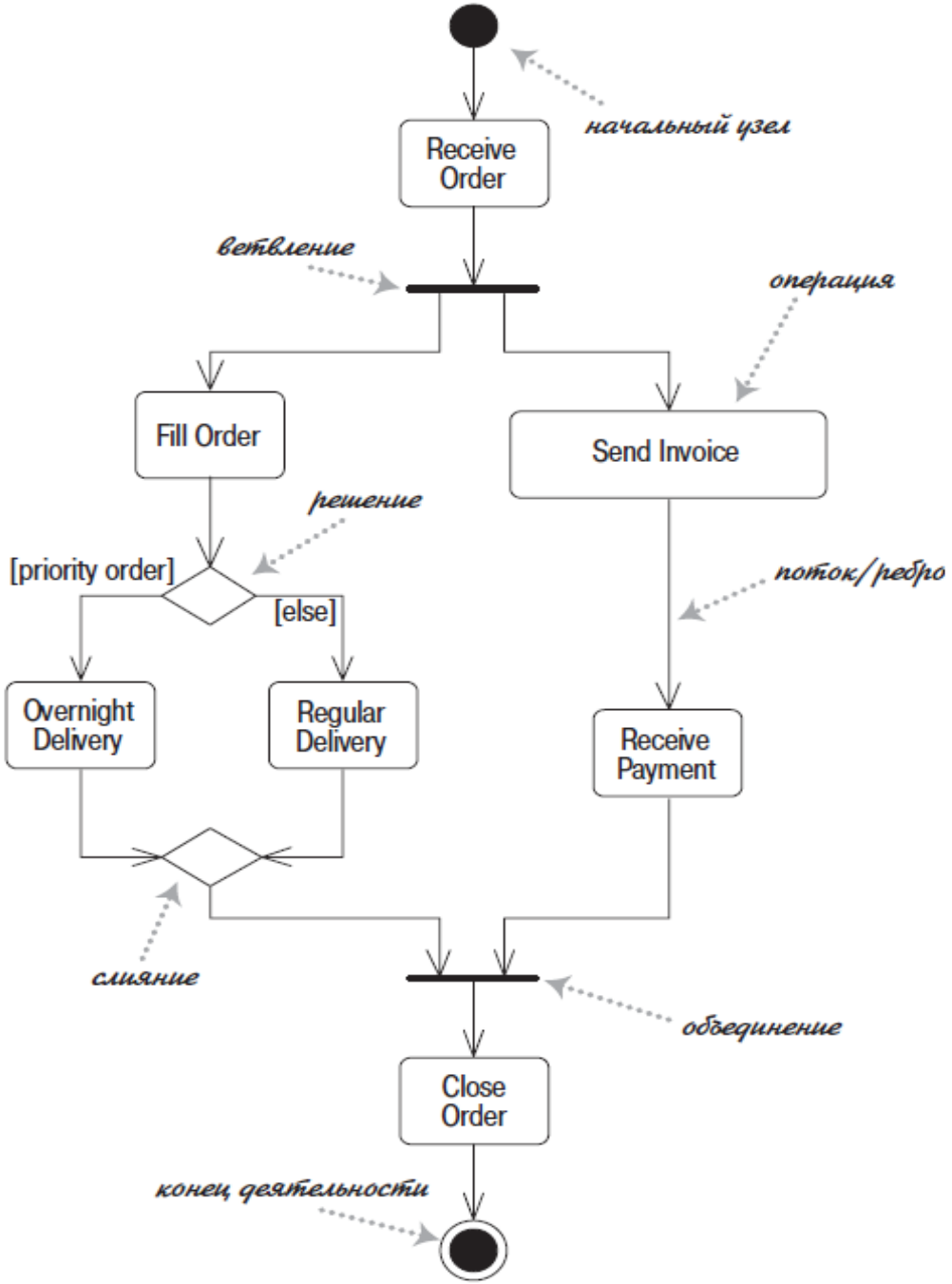


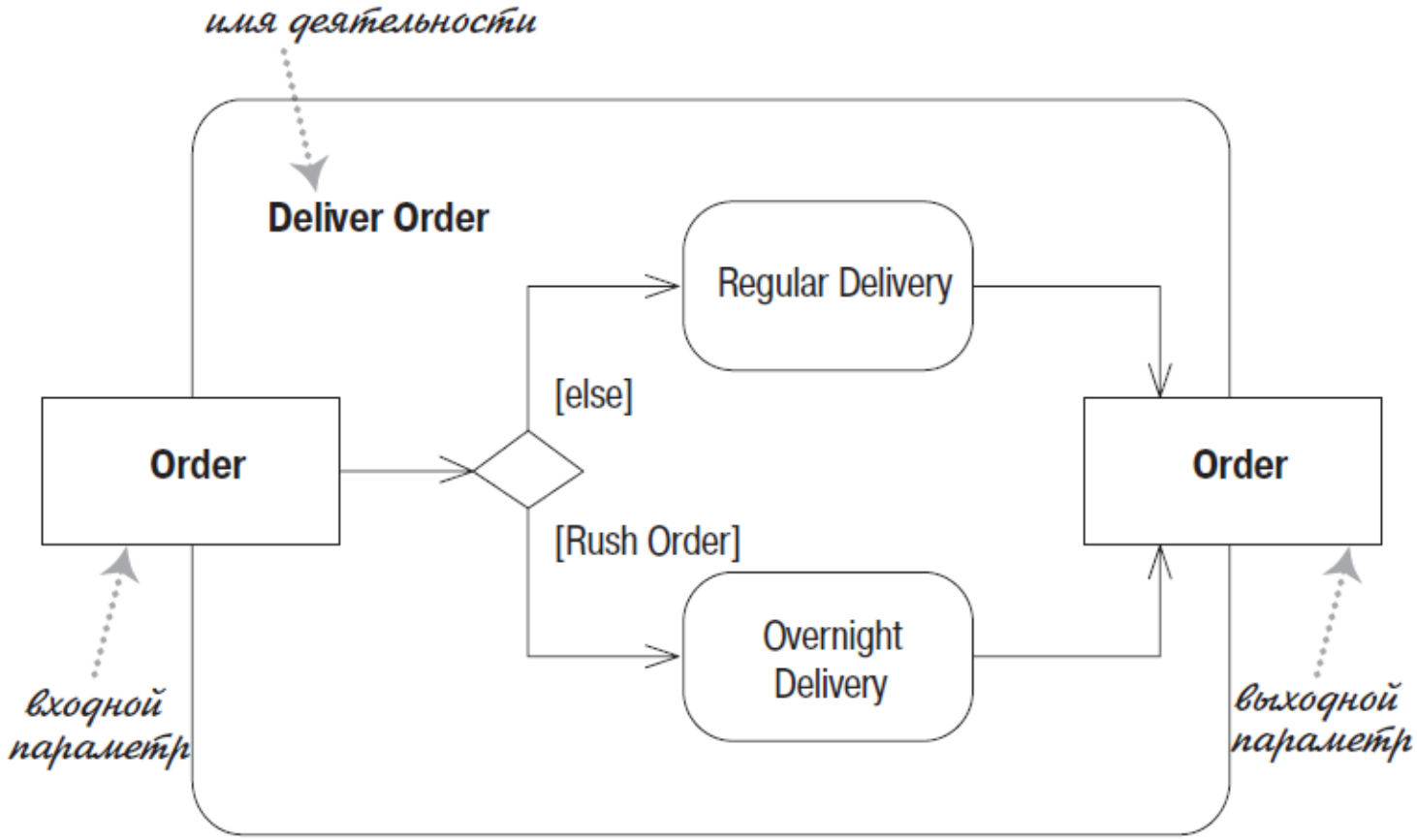
- **Диаграммы деятельности** – это технология, позволяющая описывать логику процедур, бизнес-процессы и потоки работ.

ОСНОВНЫЕ НОТАЦИИ ДИАГРАММЫ:

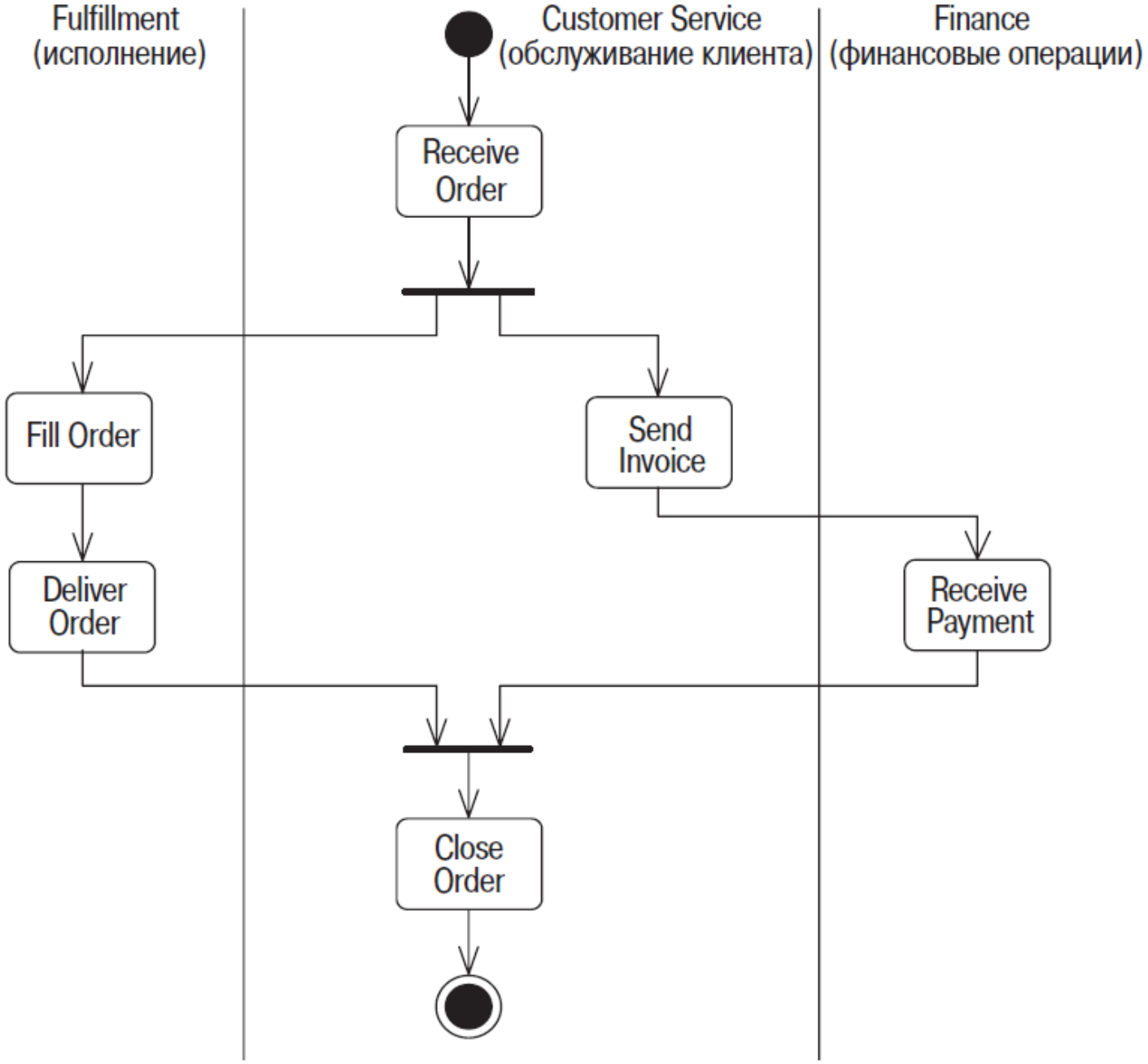
- Начальный узел (**initial node**)
- Операции (**actions**)
- Ветвление (**fork**) / Объединение (**join**)
- Решение (**decisions**) / Слияние (**merges**)
- Конечный узел (**final node**)

ПРИМЕР ПРОСТОЙ ДИАГРАММЫ ДЕЯТЕЛЬНОСТИ





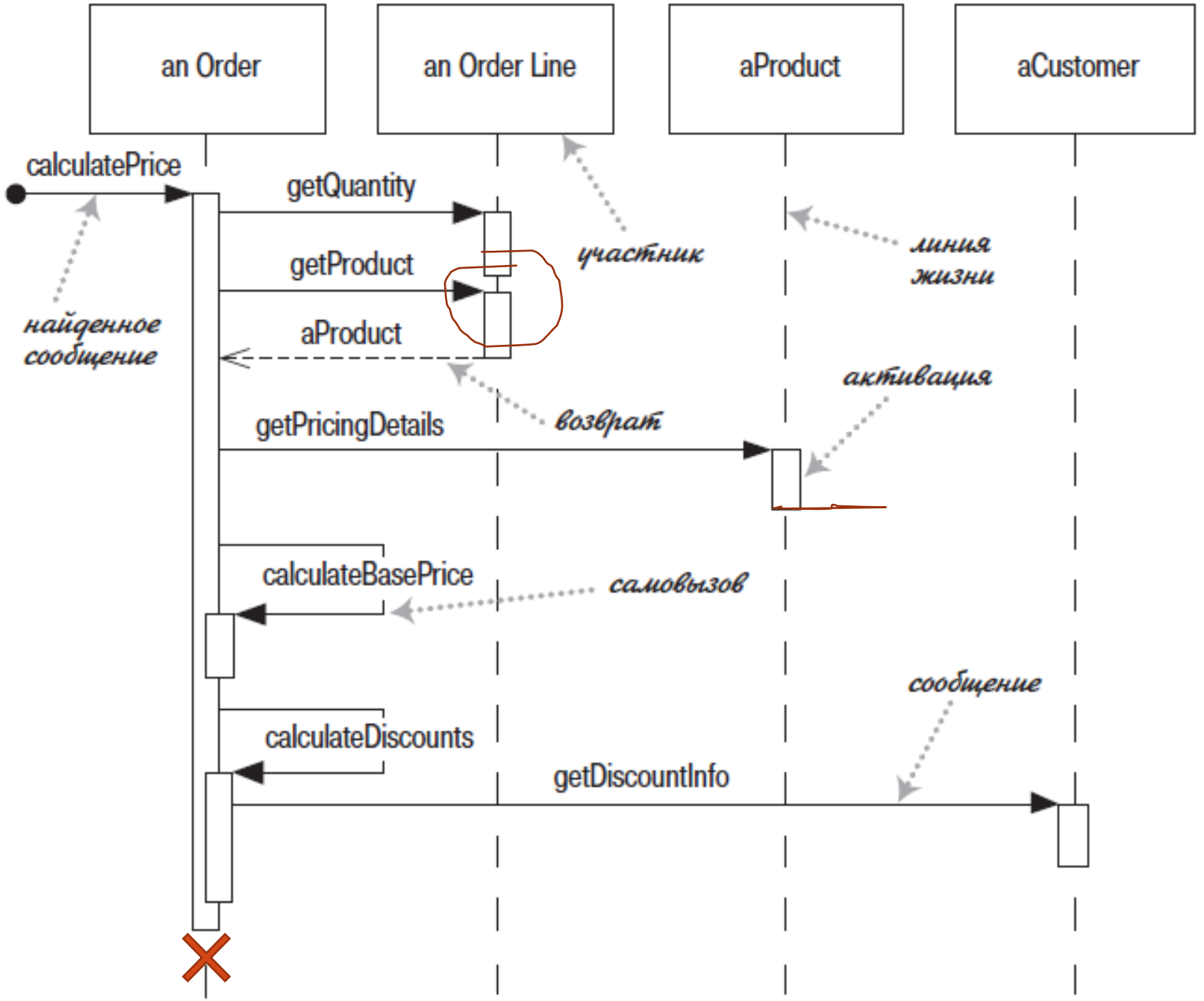
ДЕКОМПОЗИЦИЯ ОПЕРАЦИИ



ОПИСАНИЕ ПАРАЛЛЕЛЬНЫХ ПРОЦЕССОВ

ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТИ

ДИАГРАММЫ ВЗАИМОДЕЙСТВИЯ (INTERACTION DIAGRAMS) ОПИСЫВАЮТ ВЗАИМОДЕЙСТВИЕ ГРУПП ОБЪЕКТОВ В РАЗЛИЧНЫХ УСЛОВИЯХ ИХ ПОВЕДЕНИЯ.


an Order
an Order Line
aProduct
aCustomer
calculatePrice
getQuantity
getProduct
aProduct
getPricingDetails
calculateBasePrice
calculateDiscounts
getDiscountInfo
участник
линия жизни
найденное сообщение
возврат
активация
самовызов
сообщение

ПРИМЕР

СИНХРОННЫЕ И АСИНХРОННЫЕ ВЫЗОВЫ

- Если вызывающий объект посылает синхронное сообщение (**synchronous message**), то он должен ждать, пока обработка сообщения не будет закончена, например при вызове подпрограммы.
- Если вызывающий объект посылает асинхронное сообщение (**asynchronous message**), то он может продолжать работу и не должен ждать ответа.

ДИАГРАММА СОСТОЯНИЙ

ДИАГРАММЫ СОСТОЯНИЙ (STATE MACHINE DIAGRAMS) –

ЭТО ТЕХНОЛОГИЯ ОПИСАНИЯ ПОВЕДЕНИЯ СИСТЕМЫ.

ОСНОВНЫЕ НОТАЦИИ ДИАГРАММЫ:

- Начальное псевдосостояние (**initial pseudostate**)
- Состояния (**states**)
- Переход (**transition**)
- Конечное состояние (**final state**)

КАЖДЫЙ **ПЕРЕХОД** ИМЕЕТ МЕТКУ – ТРИ ПАРАМЕТРА **(!)**

Триггер-идентификатор [Защита] / Активность

(trigger-signature [guard] / activity).

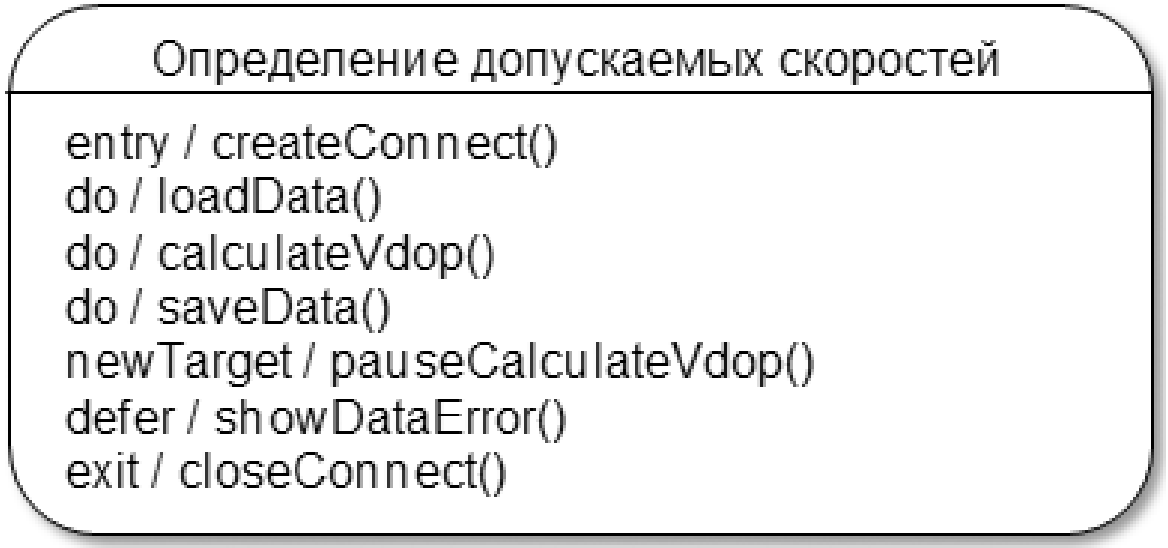
«событие [сторожевое условие] / действие»

ВНУТРЕННИЕ АКТИВНОСТИ

Состояния могут реагировать на события без совершения перехода, используя **внутренние активности** (internal activities),
и в этом случае событие, защита и активность размещаются внутри прямоугольника состояния.

Входная активность (entry/) выполняется всякий раз, когда входит в состояние;
Выходная активность (exit/) – всякий раз, когда покидает состояние.

ПРИМЕР СОСТОЯНИЯ С ВНУТРЕННЕЙ АКТИВНОСТЬЮ



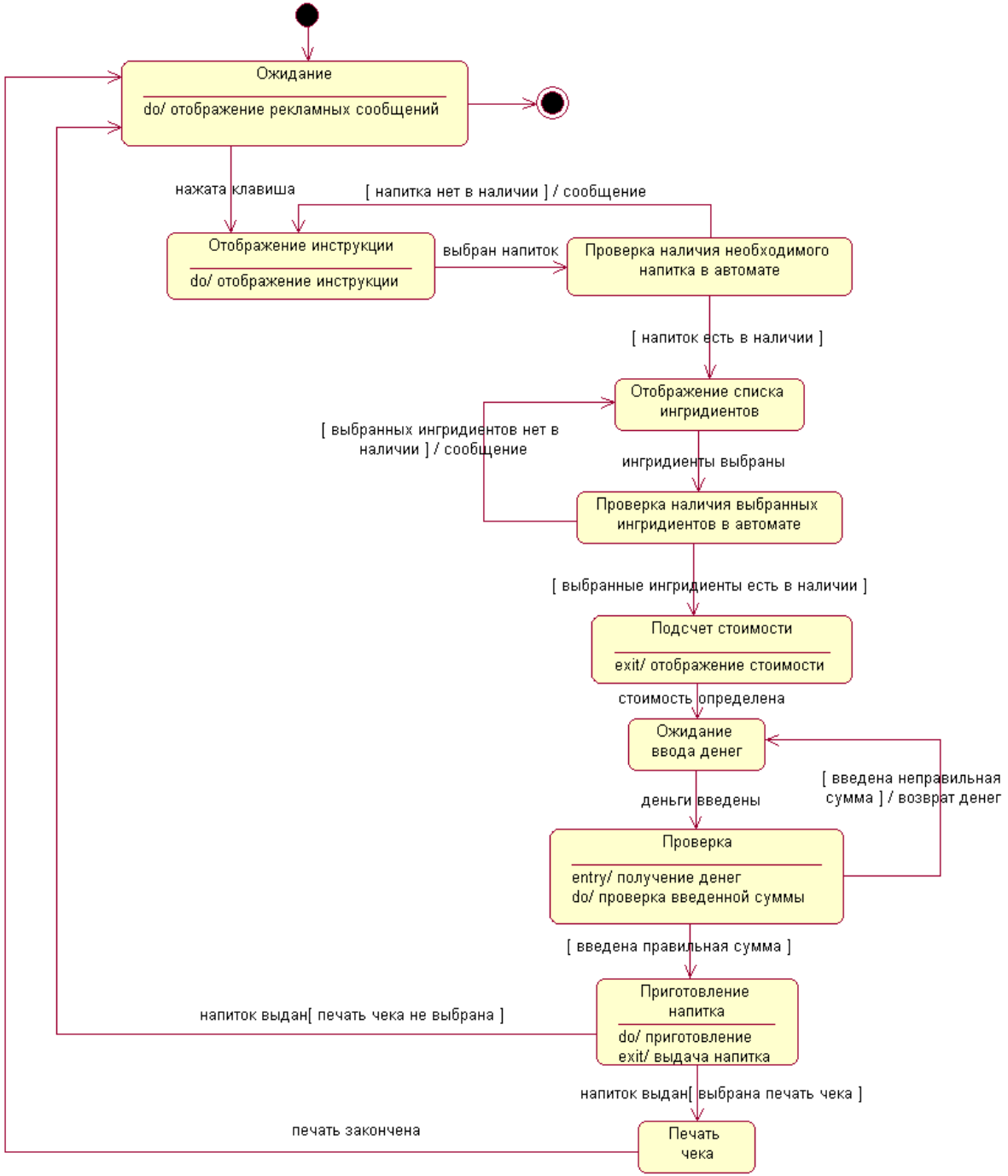
РАЗЛИЧАЮТ ДВА ВИДА ПЕРЕХОДОВ: ***НЕТРИГГЕРНЫЙ*** И ***ТРИГГЕРНЫЙ***.

- Переход первого вида, называемый также переходом по завершении, срабатывает неявно, когда все основные операции (с метками entry, do и exit) в исходном состоянии успешно завершают свою работу.
  Данный вид перехода обозначается стрелкой без надписи.

- Для наступления триггерного перехода необходимо наступление некоторого события, которое записывается над стрелкой.

- Переход также может быть ***рефлексивным,***
  т.е. направлен в то же состояние, из которого он выходит.

ПРИМЕР


Ожидание
do/ отображение рекламных сообщений
нажата клавиша
[напитка нет в наличии] / сообщение
Отображение инструкции
do/ отображение инструкции
выбран напиток
Проверка наличия необходимого напитка в автомате
[напиток есть в наличии]
Отображение списка ингридиентов
[выбранных ингридиентов нет в наличии] / сообщение
ингридиенты выбраны
Проверка наличия выбранных ингридиентов в автомате
[выбранные ингридиенты есть в наличии]
Подсчет стоимости
exit/ отображение стоимости
стоимость определена
Ожидание ввода денег
[введена неправильная сумма] / возврат денег
деньги введены
Проверка
entry/ получение денег
do/ проверка введенной суммы
[введена правильная сумма]
Приготовление напитка
do/ приготовление
exit/ выдача напитка
напиток выдан[печать чека не выбрана]
напиток выдан[выбрана печать чека]
печать закончена
Печать чека