

Использование UML для проектирования параллельных приложений

Параллельные вычисления


- **Параллельные вычисления** — способ организации компьютерных вычислений, при котором программы разрабатываются как набор взаимодействующих вычислительных процессов, работающих параллельно (одновременно) (Wikipedia)
- При проектировании параллельных программ необходимо:
 - обеспечить правильную последовательность взаимодействий между параллельными процессами;
 - обеспечить совместное использование разделяемых ресурсов, к которым имеют доступ несколько параллельных процессов.

Диаграмма деятельности


- Диаграмма деятельности используется на различных фазах жизненного цикла программного обеспечения для описания:
 - бизнес-процессов;
 - потока событий некоторого варианта использования;
 - алгоритма выполнения программы.
- Диаграмма деятельности позволяет:
 - моделировать динамические аспекты поведения системы
 - описывать как последовательные, так и параллельные действия.
- Диаграммы деятельности основаны на технологии сетей Петри. Они моделируют поведение с помощью маркеров, которые двигаются по узлам и ребрам сети согласно определенным правилам.

Действия


- Действие -элементарная единица деятельности. Описывает поведение или операцию.

 Opaque Action

- Вызов деятельности - деятельность, которая описывается отдельной диаграммой.

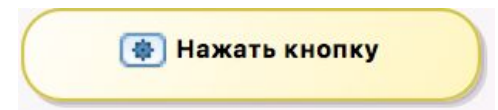
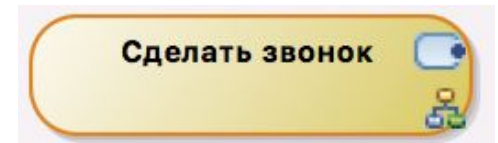
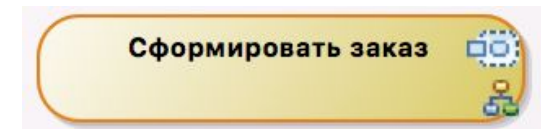
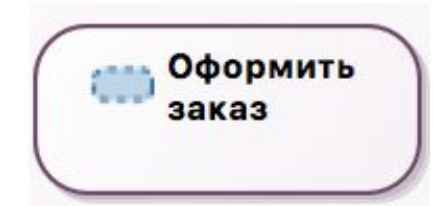
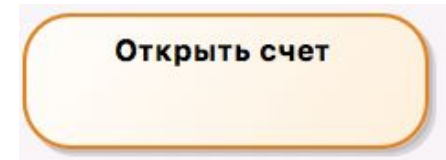
 StructuredActivityNode

- Поведение, показанное на другой диаграмме деятельности или состояний


 Call Behavior Action

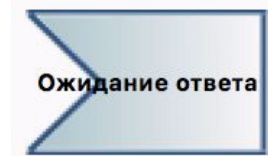
- Вызов операции класса

 Call Operation Action



Обработка событий

- Принятие события - ожидание некоторого события. После принятия события запускается следующее действие  **Accept Event Action**



- Принятие события времени - наступление определенного момента времени



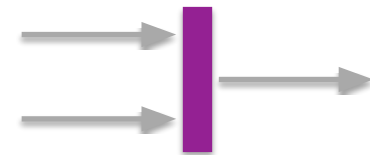
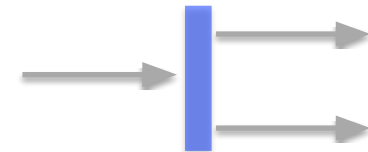
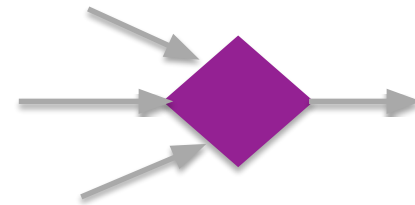
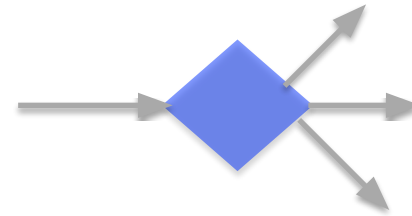
или ожидание некоторого промежутка времени

 **Wait Time Action**

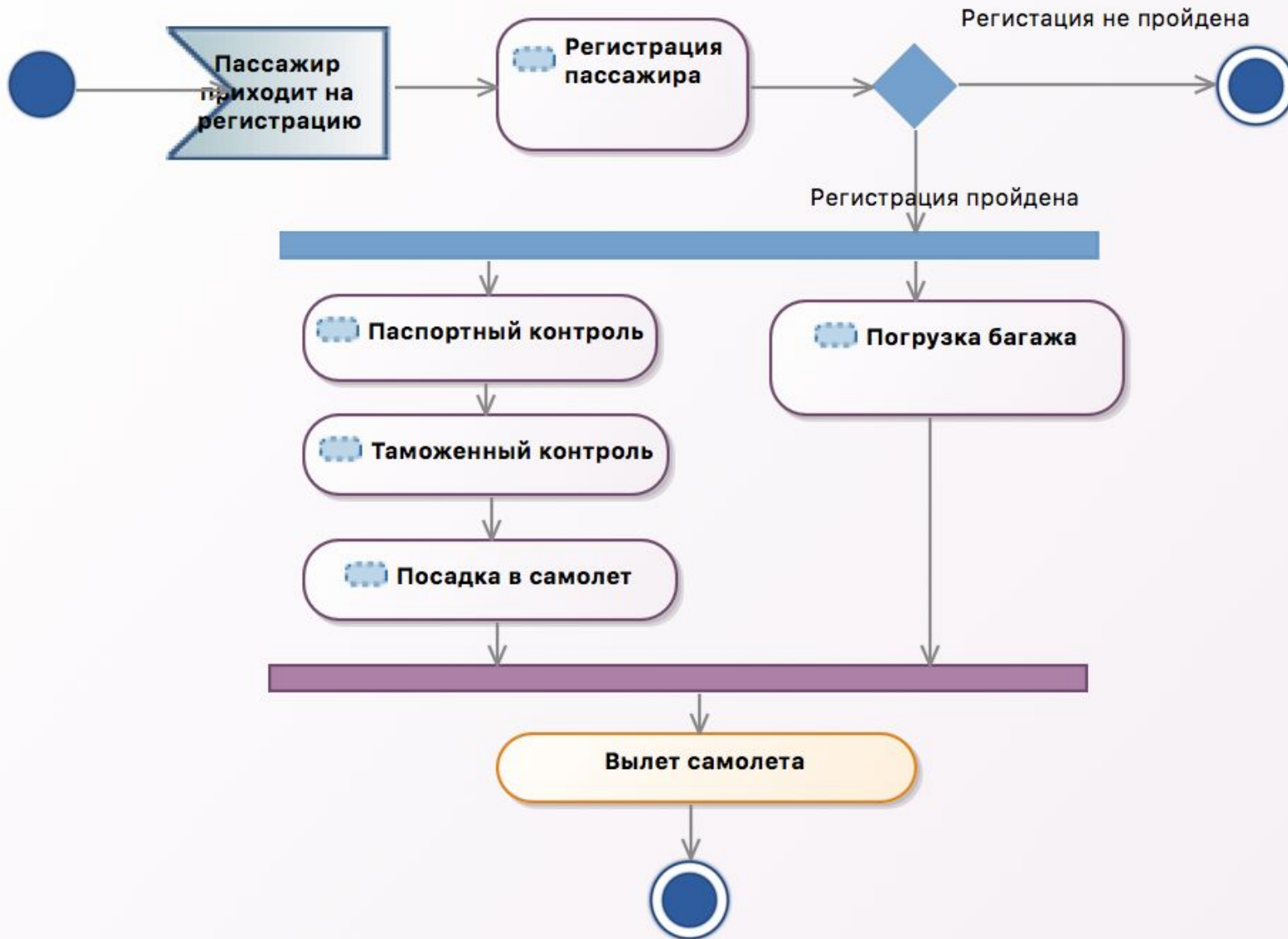


Элементы диаграммы деятельности

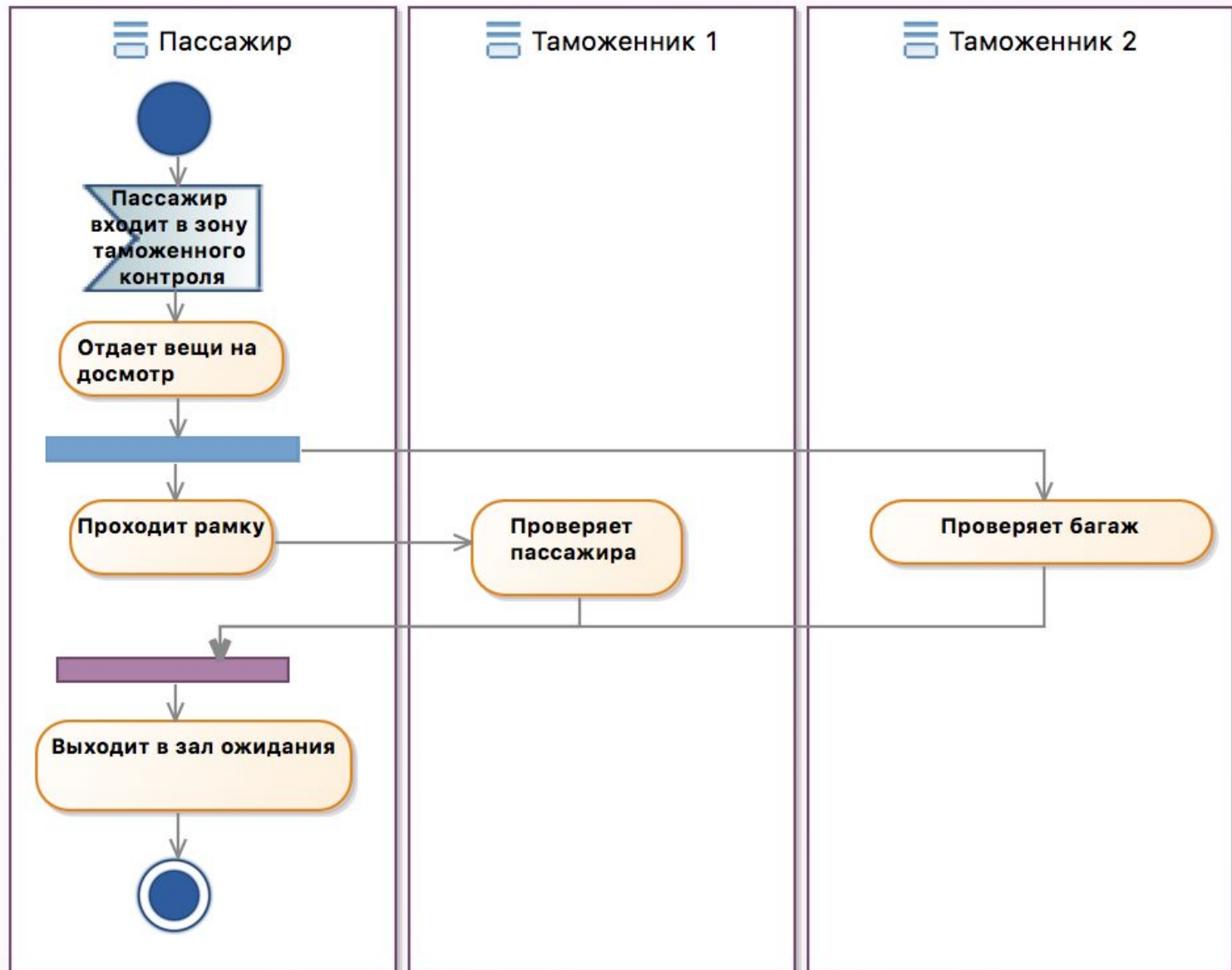
- **Узел ветвления** - определяет переход по одному из путей, условие которого принимает значение истина.
- **Узел слияния** - перенаправляет поток, пришедший из разных узлов в одно исходящее ребро.
- **Узел разделения** - распараллеливает входной поток на несколько параллельных потоков.
- **Узел соединения** - ожидает выполнения (синхронизирует) несколько параллельных потоков.




В аэропорту



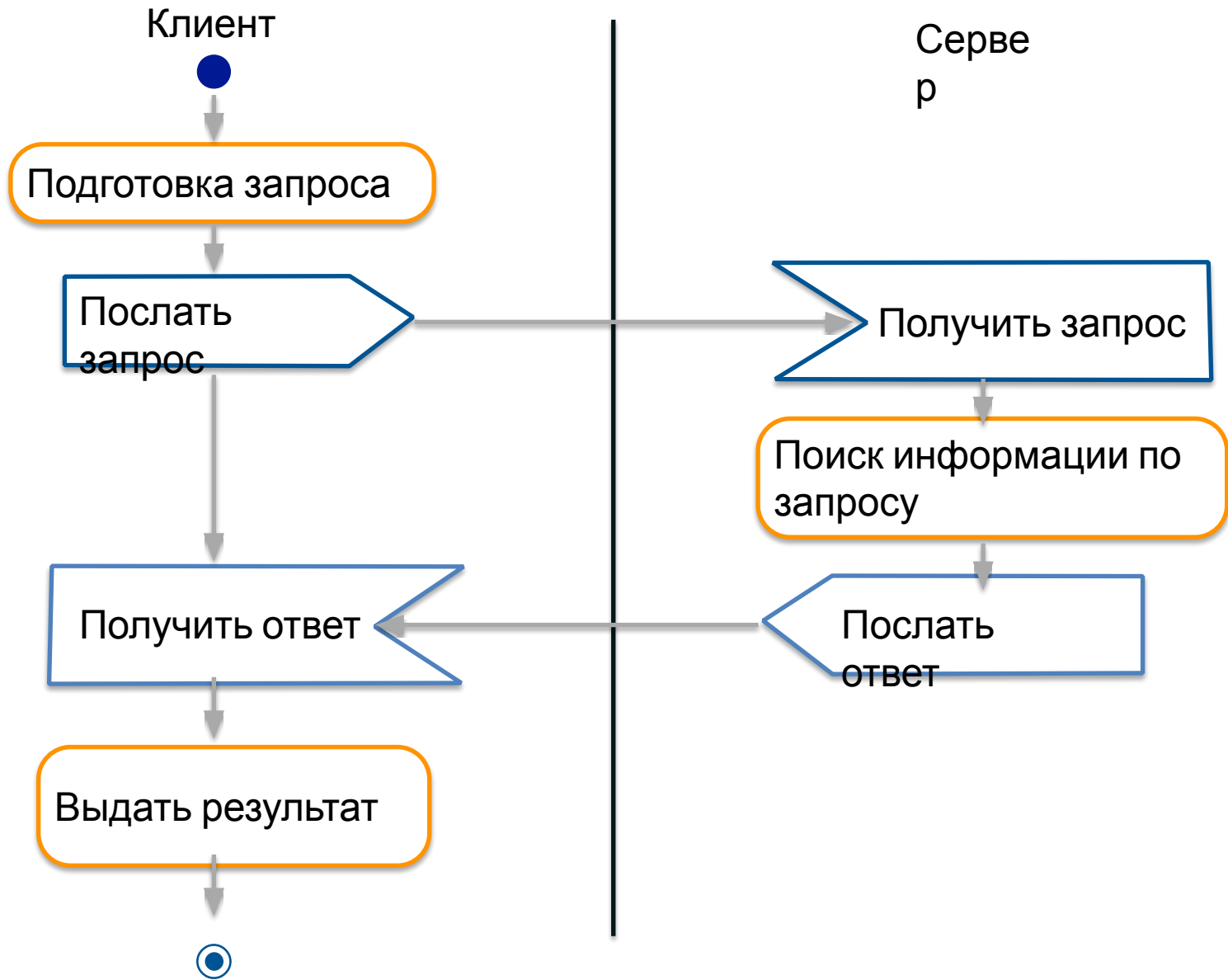
Таможенный контроль



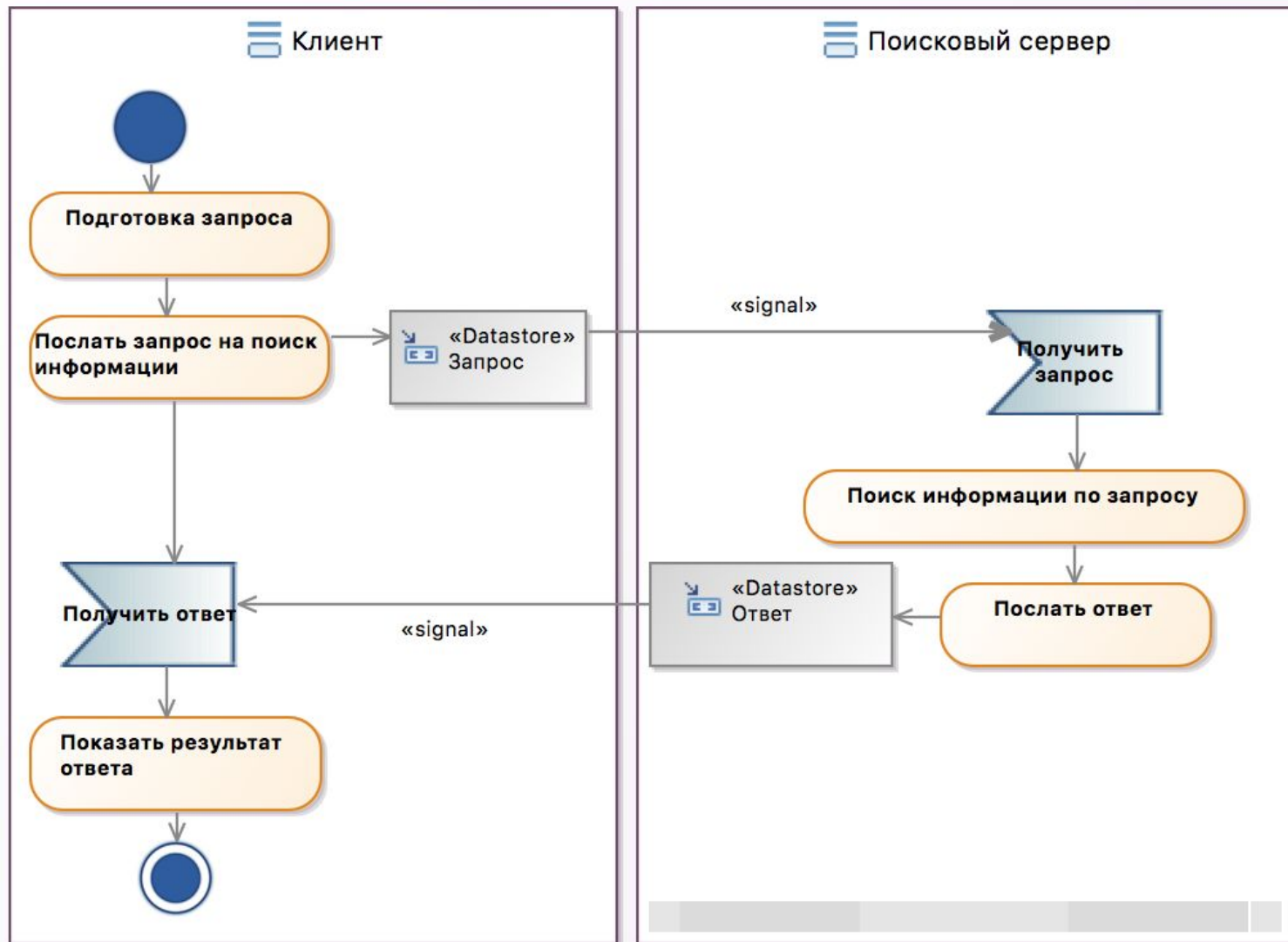
Отправка сигналов и прием событий

- Сигнал - это асинхронно передаваемая информация.
- Сигнал моделируется как класс со стереотипом «signal». Информация хранится в атрибутах сигнала.
- При выполнении действия «сигнал» создается и посылается объект сигнала.
- Сигнал не ожидает ответа и происходит переход к следующему действию.
- Символ отправки сигнала  отсутствует в UML Designer.
- Действие «событие» имеет один вход, который запускает событие, или не имеет его, тогда действие запускается при вызове деятельности владельца.
- Действие «событие» ожидает получение события определенного вида.

Деятельность «клиент-сервер»



Клиент-сервер

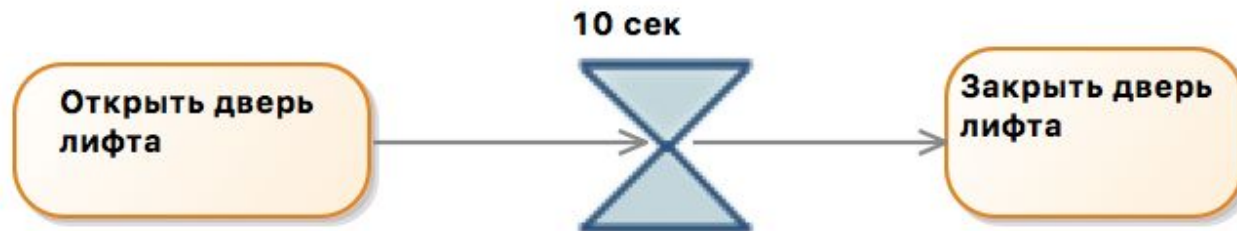


Обработка событий времени

- События, которые происходят в определенный момент времени

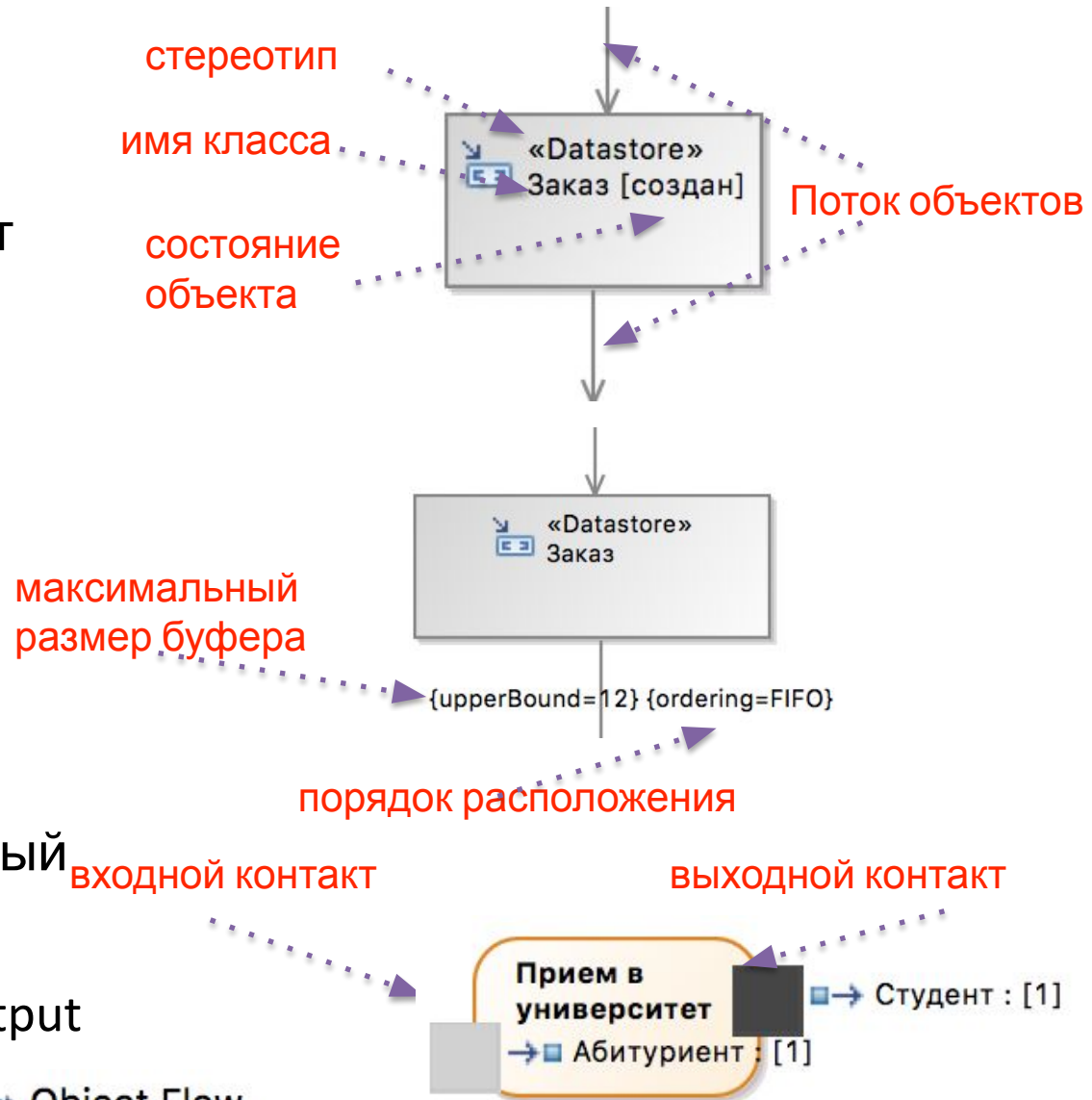


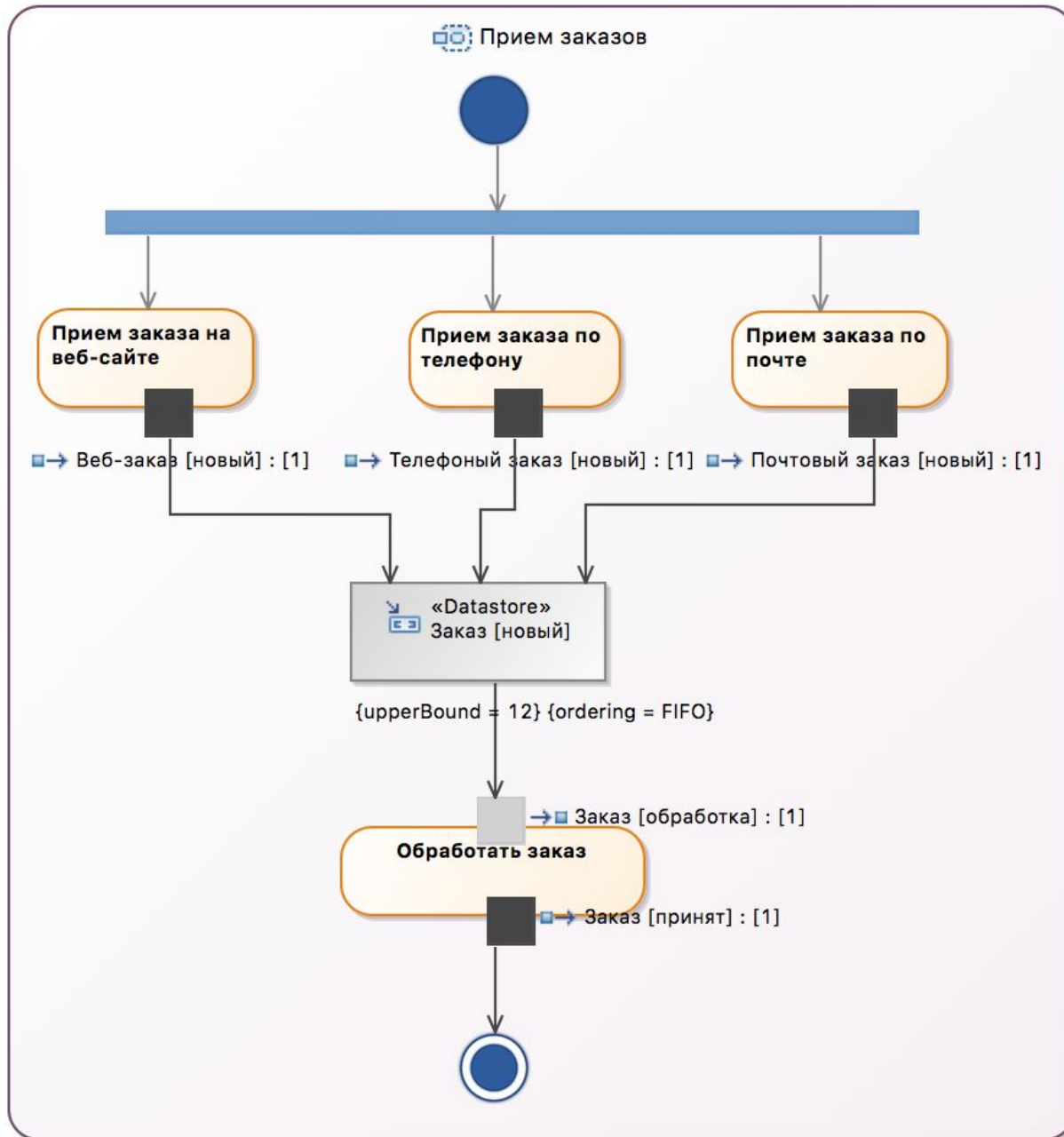
- События, которые происходят через определенный интервал времени




Объекты на диаграмме деятельности

- **Объектный узел** служит для показа того, что объект некоторого класса доступен в некоторой точке деятельности.
- Объект может иметь **состояние**.
- Объектный узел используется для буферизации потока объектов.
- **Контакт** - объектный узел, являющийся входом (input pin) или выходом действия (output pin).
- Для связи объектных узлов используется **Object Flow**




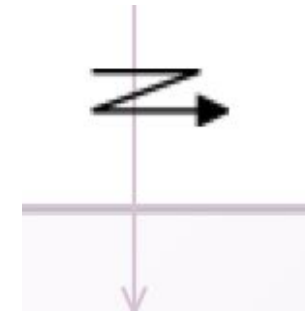


Область с прерыванием выполнения действий

- **Область с прерыванием** - это область, включающая в себя ряд действий, выполнение которых может быть прервано. Визуализируется с помощью  InterruptibleActivityRegion

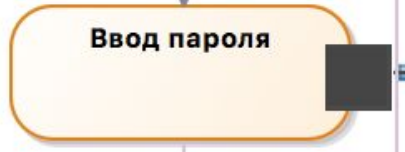


- **Прерывающее ребро** показывает выход из области по прерыванию и используется для показа асинхронных биз. Визуализируется с помощью  Interrupting Edge



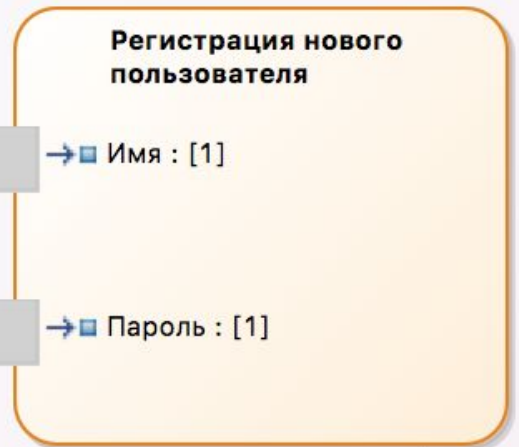
Регистрация пользователя

Ввод данных



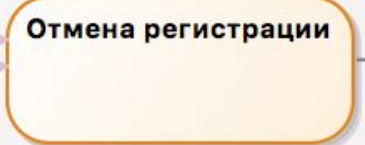
Имя [допустимое] : [1]

Пароль [допустимый] : [1]

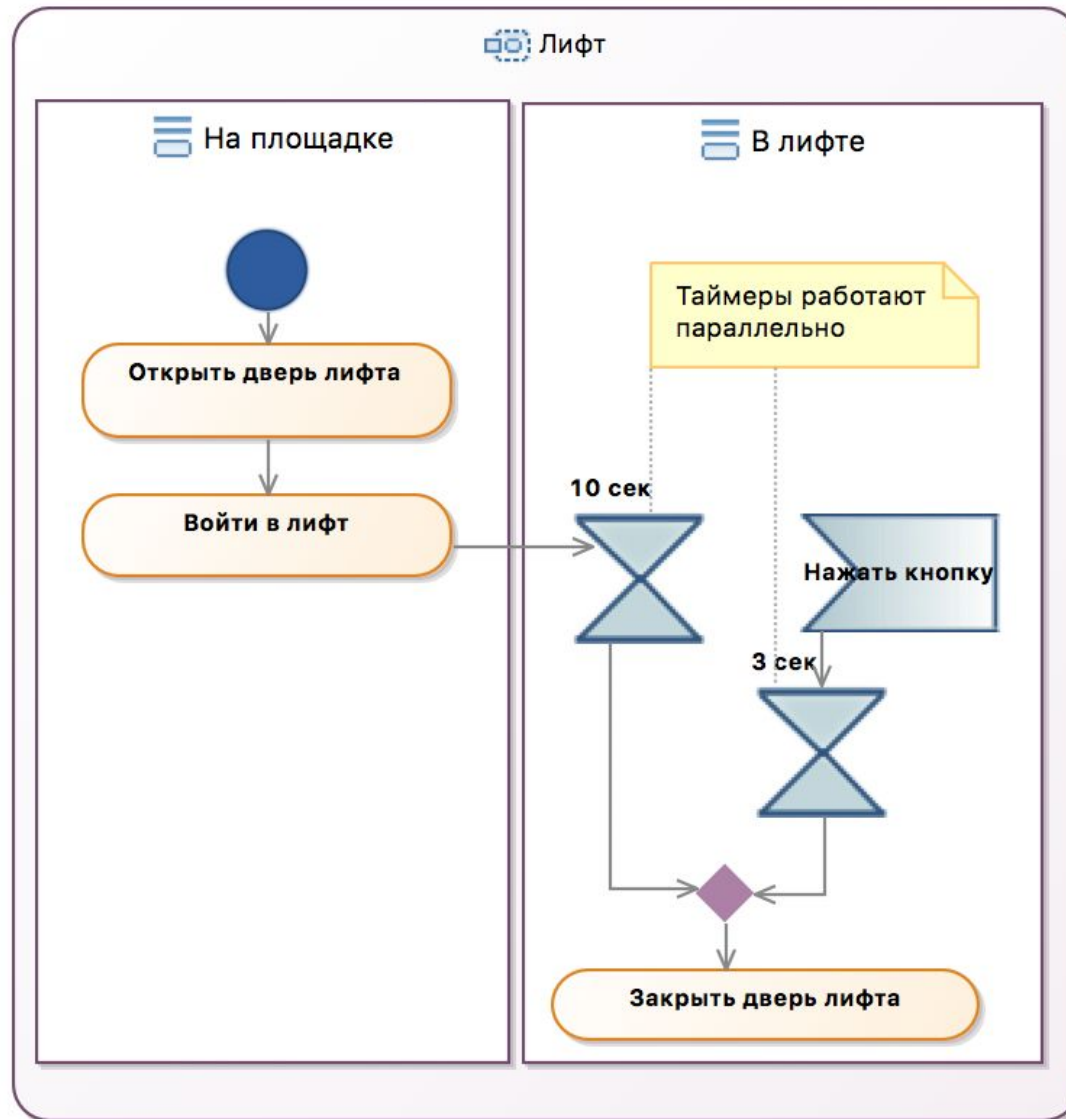


Имя : [1]

Пароль : [1]



Параллельные события



Групповая рассылка

При групповой рассылке один и тот же объект передается нескольким получателям

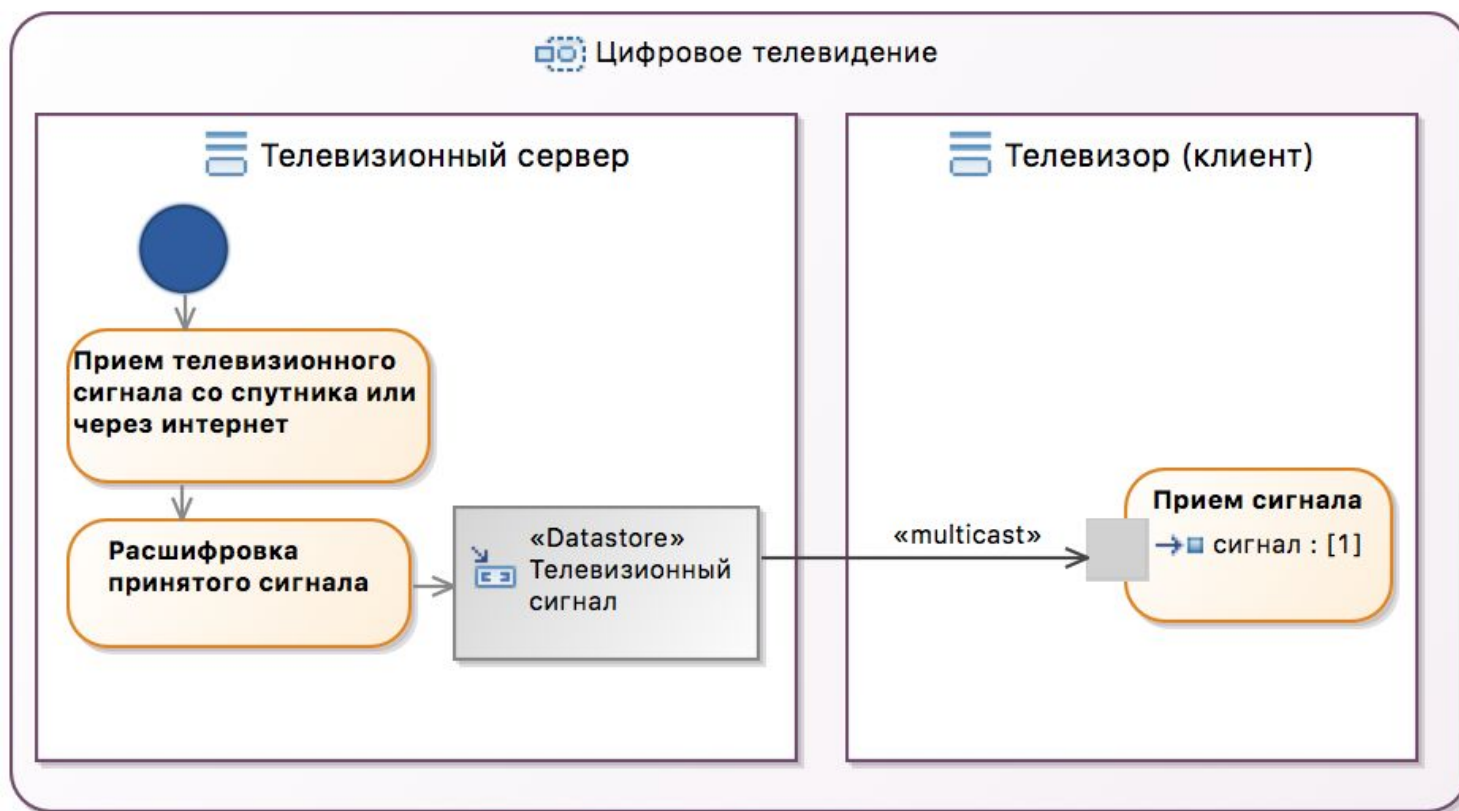
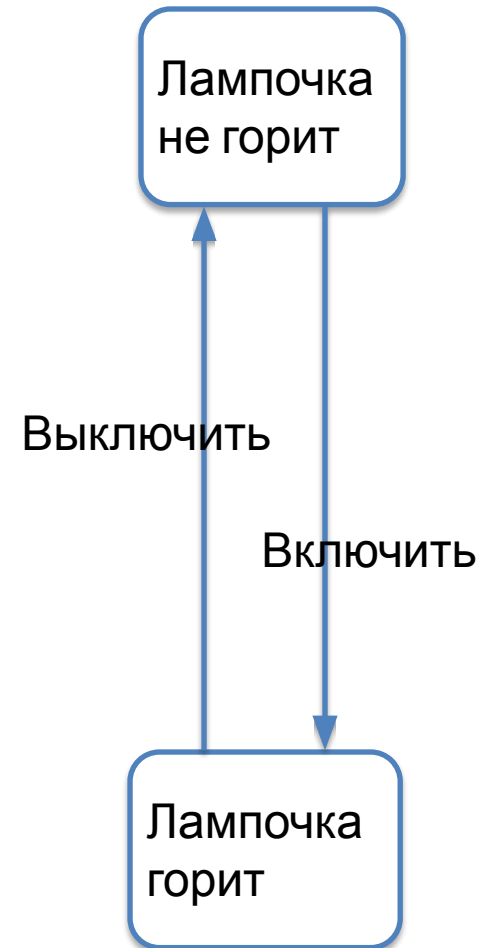


Диаграмма состояний

- Моделирует динамические аспекты поведения системы.
- Представляет собой конечный автомат
- Основные элементы: состояния, события, переходы.
- Может использоваться для моделирования поведения: объектов классов, вариантов использования, подсистем, систем.

Элементы диаграммы состояний

- **Состояние.** В каждый момент времени объект может находиться в некотором состоянии, которое определяется: значениями атрибутов объекта, отношениями с другими объектами, выполняемой деятельностью.
- **Переход** - показывает переход объекта из одного состояния в другое.
- **Событие** - некоторое воздействие, вызывающее переход объекта из одного



Виды событий

- **Событие вызова** - вызов некоторой операции объекта.
- **Сигнал** - информационный объект, асинхронно передаваемый между классами.
- **Событие изменения** - смена значения логического выражения.
- **Событие времени** - наступление некоторого момента времени или прохождение некоторого интервала времени.

Снять деньги со счета
[положительный баланс]

Послать письмо

Получить письмо

(Поступил звонок AND
Абонент свободен)

when(Конец года)

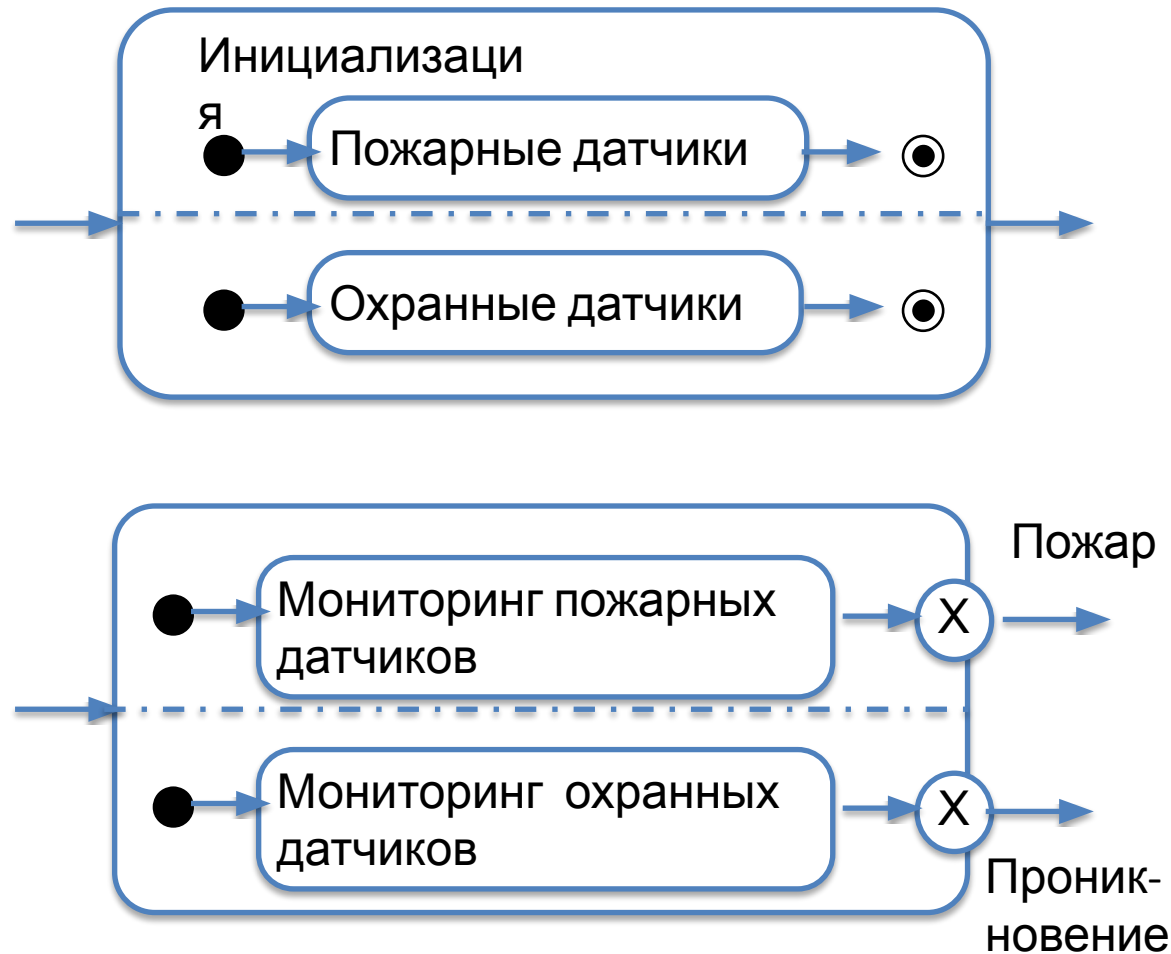
after(5 мин)

Составное (композиционное) состояние

- Это состояние, которое представляет из себя диаграмму состояний.
- Содержит вложенные состояния, которые реагируют на все события составного состояния.
- Любой переход в/из составного состояния также выполняется для вложенного состояния.
- Вложенные состояния объединяются в один или несколько конечных автоматов (подавтоматов).
- Простое составное состояние содержит один конечный автомат.
- Ортогональное составное состояние содержит два и более параллельно работающих конечных автоматов.

Вход и выход из ортогонального составного состояния

- При входе в ортогональное составное состояние одновременно запускаются все подавтоматы.
- Выход осуществляется только по завершении всех подавтоматов.
- Некоторый подавтомат может осуществить выход. Выход обычно обозначается псевдосостоянием. Остальные подавтоматы прерывают свое



Ортогональное вложенное состояние «Обучение» для объекта класса «Студент»

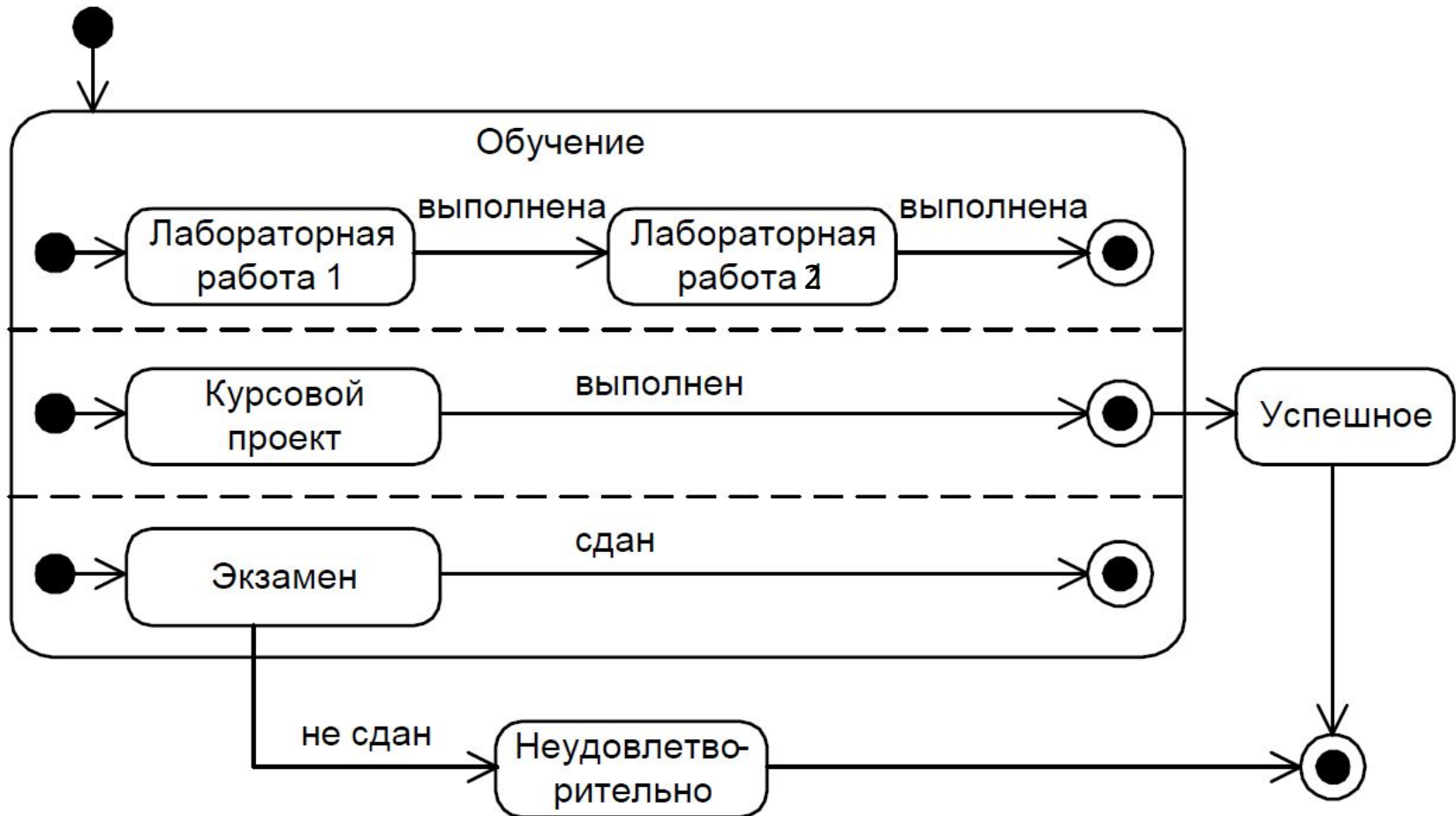
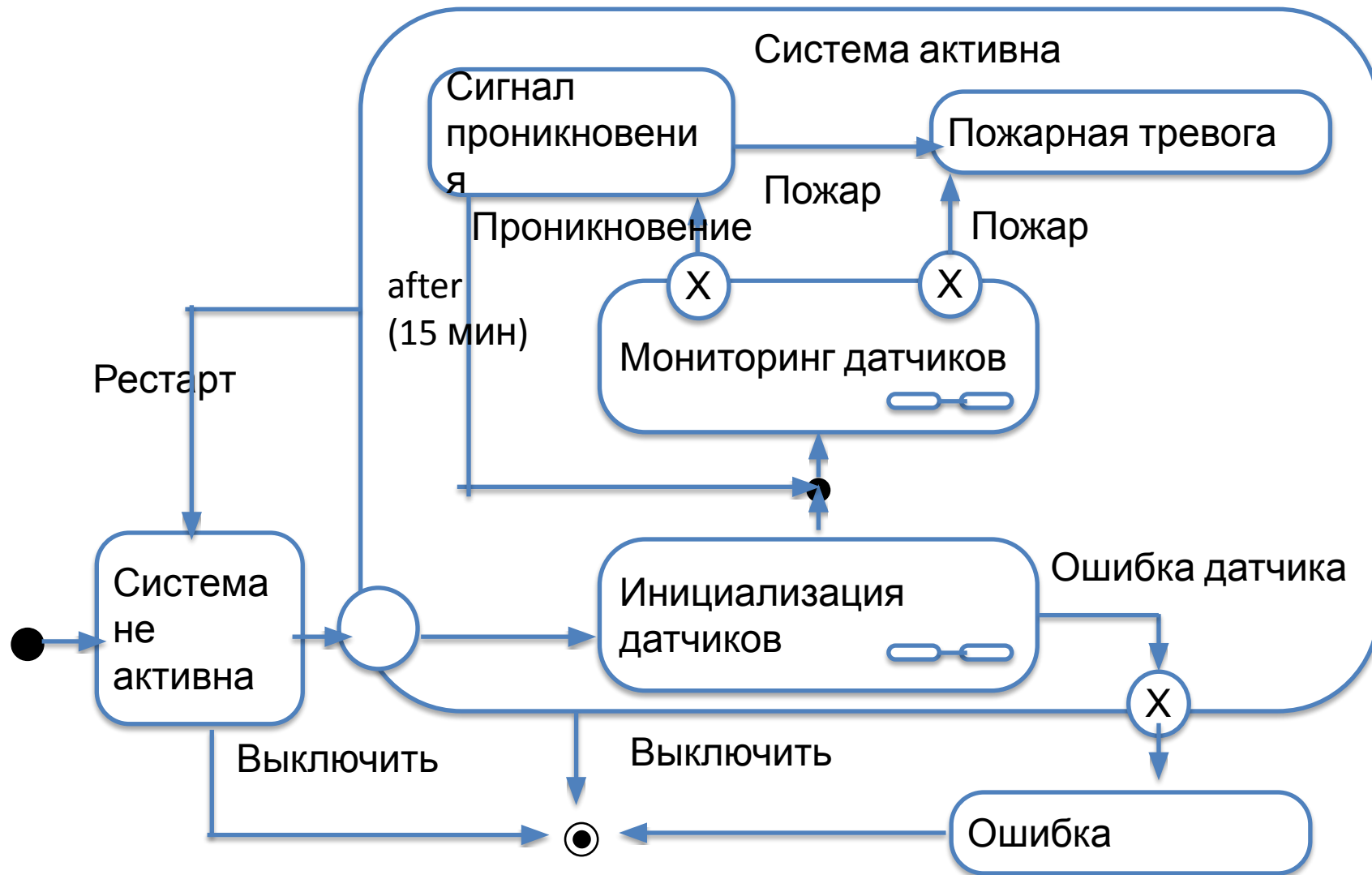
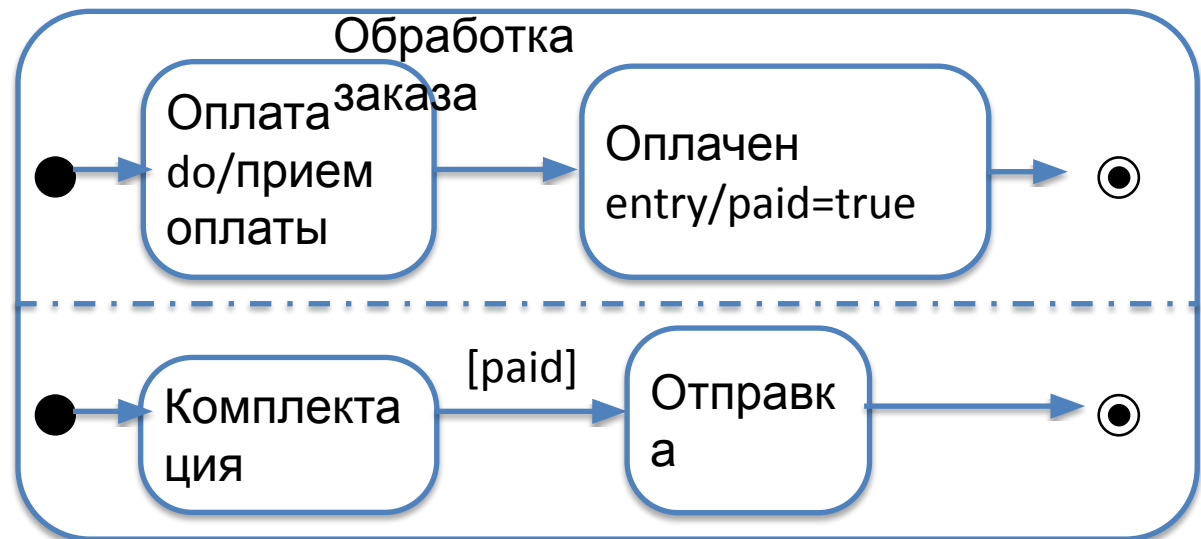
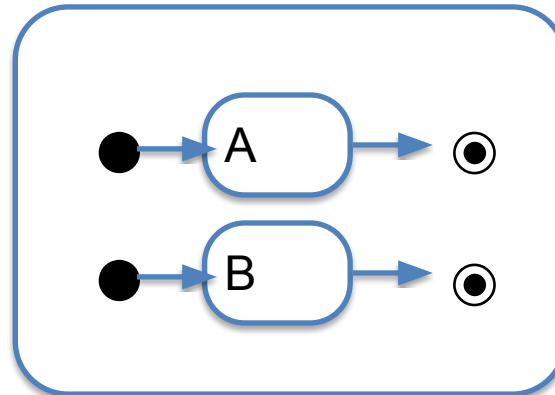


Диаграмма состояний системы сигнализации



Взаимодействие подавтоматов

- Автоматы могут ждать окончания выполнения друг друга (синхронное взаимодействие)
- Асинхронное взаимодействие может быть организовано с помощью установки значений атрибутов



Предистория

- Выход из составного состояния S по некоторому событию возможен из различных подсостояний (например, подсостояния A).
- После выхода из составного состояния S автомат может побывать в нескольких внешних состояниях, а затем вернуться в состояние S .
- Как продолжить работу в подсостоянии A с момента, на котором она была прервана?
- Для этой цели вводится псевдосостояние предистории.

Псевдосостояние предистории

- Псевдосостояние предистории имеет один или несколько входов и один выход.
- Псевдосостояние предистории запоминает подсостояние, в котором находился автомат при выходе из составного состояния. При возврате из внешнего состояния в псевдосостояние произойдет переход в запомненное состояние.
- Например, при переходе компьютера в спящий режим, операционная система запоминает с каким приложением вы работали и восстанавливает работу с ним после выхода из этого режима.

Типы предисторий

- Неглубокая предистория обозначается псевдосостоянием H (Shallow History).
- Глубокая предистория обозначается псевдосостоянием H^* (Deep History).
- Псевдосостояние неглубокой предистории запоминает подсостояние, расположенное на первом уровне вложенности состояний в составное состояние.
- Псевдосостояние глубокой предистории запоминает подсостояние, расположенное на любом уровне вложенности состояний в составное состояние.

Диаграмма состояний «Просмотр каталога товаров»



Разделы диаграммы последовательности

- Отдельные фрагменты диаграммы взаимодействия можно выделить с помощью раздела. Изображается в форме прямоугольника.
- **alt** - раздел показывает нескольких альтернативных (alternative) сценариев выполнения процесса. Выполняется только один фрагмент, условие которого истинно.
- **opt** - раздел показывает не обязательный (optional) фрагмент, который выполняется при указанном условии.
- **par** - параллельный (parallel) раздел содержит несколько параллельно выполняемых фрагментов. Объекты, работающие параллельно, изображаются с двойной рамкой справа и слева.
- **loop** - цикл (loop) - раздел содержит действия, выполняемые в цикле.
- **critical** - критическая область (critical region) содержит последовательность действий, которые нельзя прервать.
- **neg** - отрицательный (negative) раздел обозначает неверное взаимодействие.
- **ref** - ссылка (reference) - раздел, который ссылается на взаимодействие, определенное на другой диаграмме. Раздел рисуется так чтобы охватить линии жизни объектов вовлеченных во

Диаграмма последовательности «Активация системы сигнализации»

