

# Диаграммы взаимодействия

# Диаграммы взаимодействия

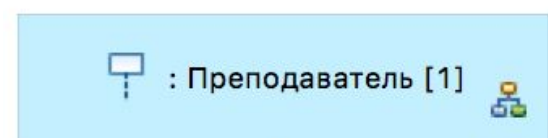
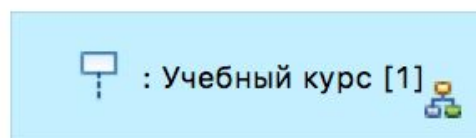
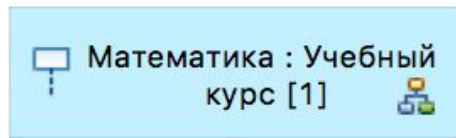
- Диаграмма взаимодействия служит для показа взаимодействия между объектами системы с помощью передачи сообщений.
- Диаграмма взаимодействия используется для описания потока событий (сценария) некоторого варианта использования.
- Существует два вида диаграмм взаимодействия:
  - диаграмма последовательности (sequence diagram),
  - диаграмма коммуникации (communication diagram).
- Оба вида диаграмм используются для показа динамических аспектов системы.

# Диаграмма последовательности

- Диаграмма последовательности (sequence diagram) показывает взаимодействие объектов во времени.
- На диаграмме показаны объекты и классы, используемые в сценарии, и последовательность сообщений, которыми обмениваются объекты, для выполнения сценария.
- На диаграмме изображаются только объекты, которые непосредственно участвуют во взаимодействии и не показываются возможные статические ассоциации с другими объектами.
- Инициатором взаимодействия в системе может быть актер. В этом случае актер изображается на диаграмме последовательности самым первым объектом слева.
- Для одного варианта использования может быть построено несколько диаграмм последовательностей. Обязательно надо изображать на диаграммах альтернативные потоки событий.

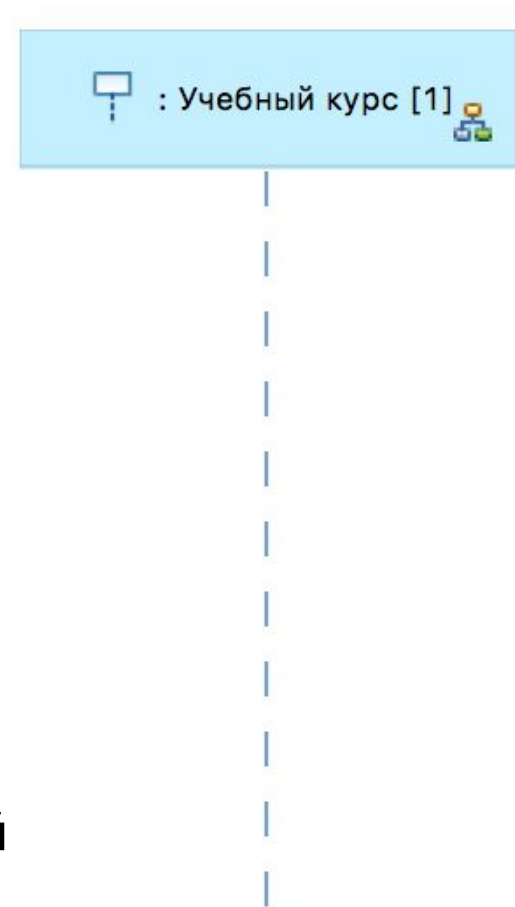
# Объекты

- Объект на диаграмме последовательности изображается прямоугольником.
- Название объекта может состоять из имени объекта и его класса или только имени класса (анонимный объект).
- На диаграмме можно показывать актеров. В UML Designer они изображаются также прямоугольником.
- В UML Designer можно создать новый объект (актера) или выбрать уже созданный ранее объект (актера).



# Линия жизни объекта

- Линия жизни объекта (lifeline) изображается пунктирной вертикальной линией. Она связана с единственным объектом на диаграмме последовательности.
- Линия жизни служит для обозначения периода времени, в течение которого объект существует в системе и может потенциально участвовать во всех взаимодействиях.
- Если объект существует в системе постоянно, то и его линия жизни должна продолжаться по всей плоскости диаграммы последовательности от самой верхней части до нижней части.

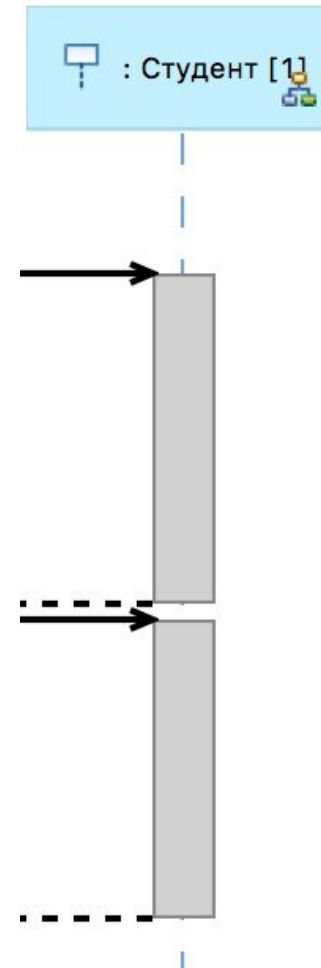


# Создание и уничтожение объектов

- Объекты, создаваемые в начальный момент времени, изображаются в верхней части диаграммы.
- Объекты в системе могут создаваться по мере необходимости, существенно экономя ресурсы и повышая производительность системы. Прямоугольник такого объекта изображается в той части диаграммы, которая соответствует моменту создания объекта.
- Отдельные объекты, выполнив свою роль в системе, могут быть уничтожены, чтобы освободить занимаемые ими ресурсы. Для таких объектов линия жизни обрывается в момент его уничтожения.

# Фокус управления

- В процессе функционирования системы одни объекты могут находиться в активном состоянии, непосредственно выполняя определенные действия, другие - в пассивном, ожидая сообщений от других объектов.
- Фокуса управления (focus of control) используется для показа активного состояния объекта.
- Фокус управления показывает период времени, в течении которого объект выполняет некоторую деятельность.



# Сообщения

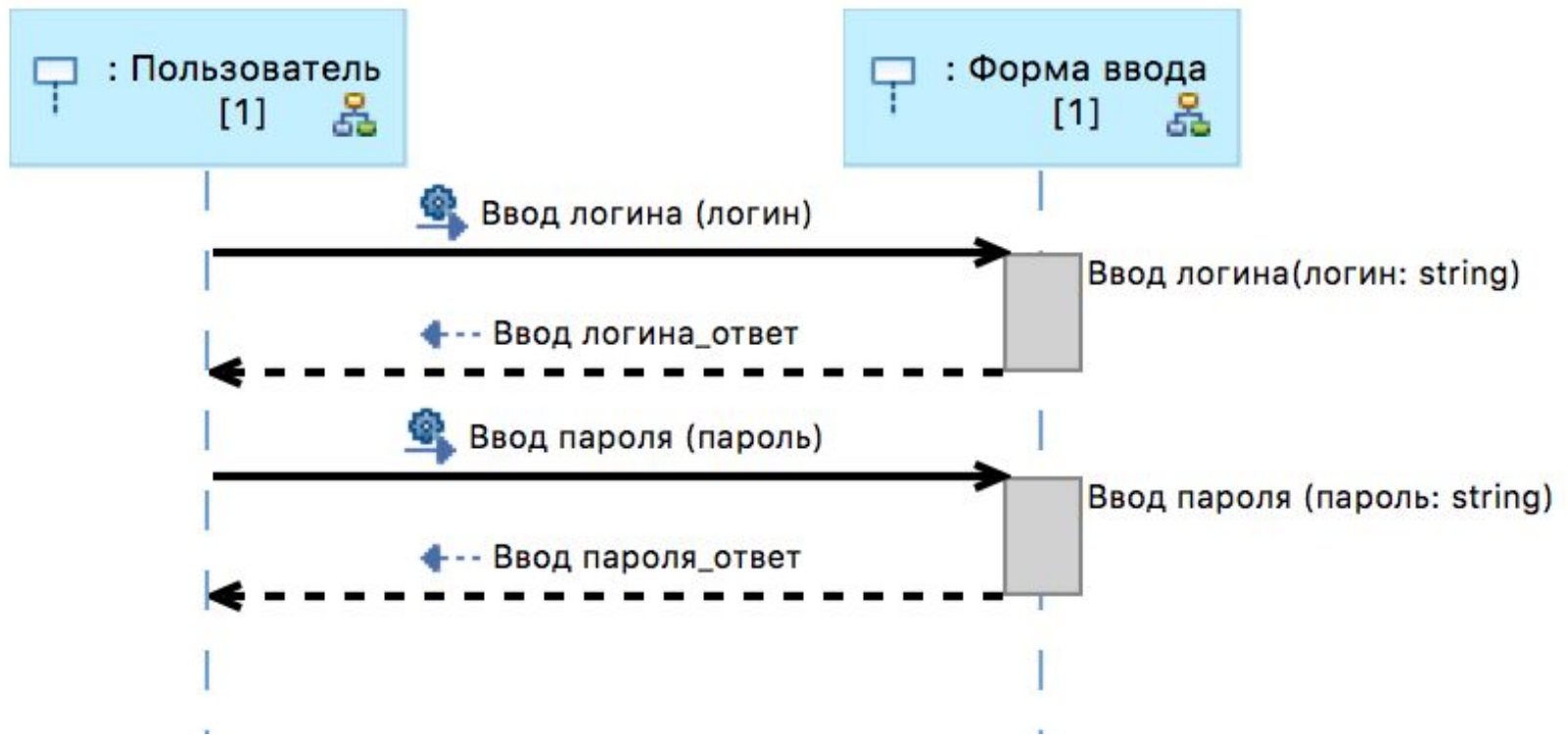
- Сообщение (message) представляет собой законченный фрагмент информации, который один объект отправляет другому объекту.
- При получении сообщения объект выполняет определенные действия.
- Отправителя сообщения иногда называют клиентом, а получателя - сервером.
- Клиент посылает запрос серверу.
- Сервер после получения сообщения может выполнить определенные действия и передать клиенту необходимую информацию тоже в форме сообщения.
- Порядок передачи сообщений соответствует их расположению на диаграмме. Сообщения, расположенные выше передаются раньше нижних. Очередность сообщений, расположенных на одной линии, определяется стрелками.



# Синхронные сообщения

- Сообщения бывают **синхронными и асинхронными**.
- Если клиент посылает синхронное сообщение, то он ждет ответа на это сообщение.
- Сообщение возврата может не указываться, т. к. синхронное сообщение обычно реализуется с помощью вызова операции, которая предполагает возврат одного или нескольких параметров.

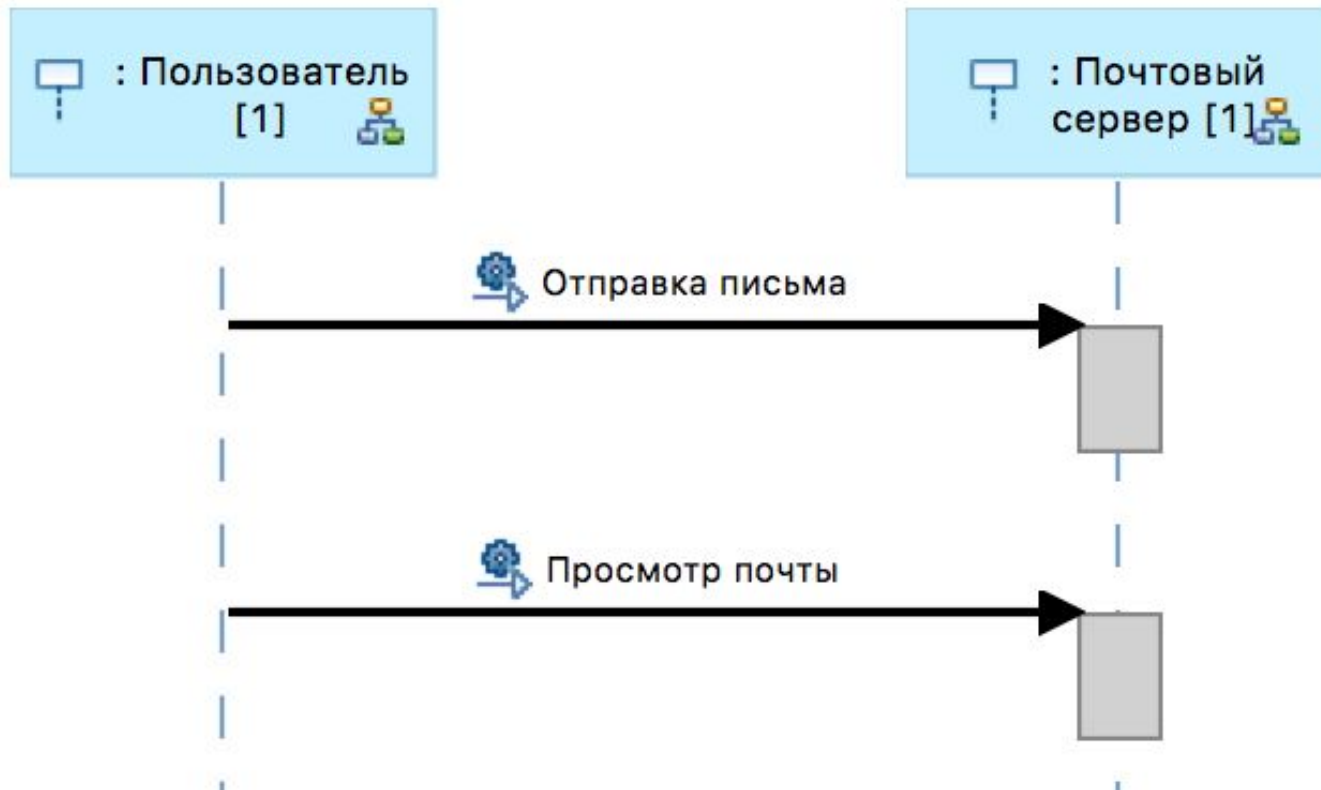
# Пример синхронного сообщения



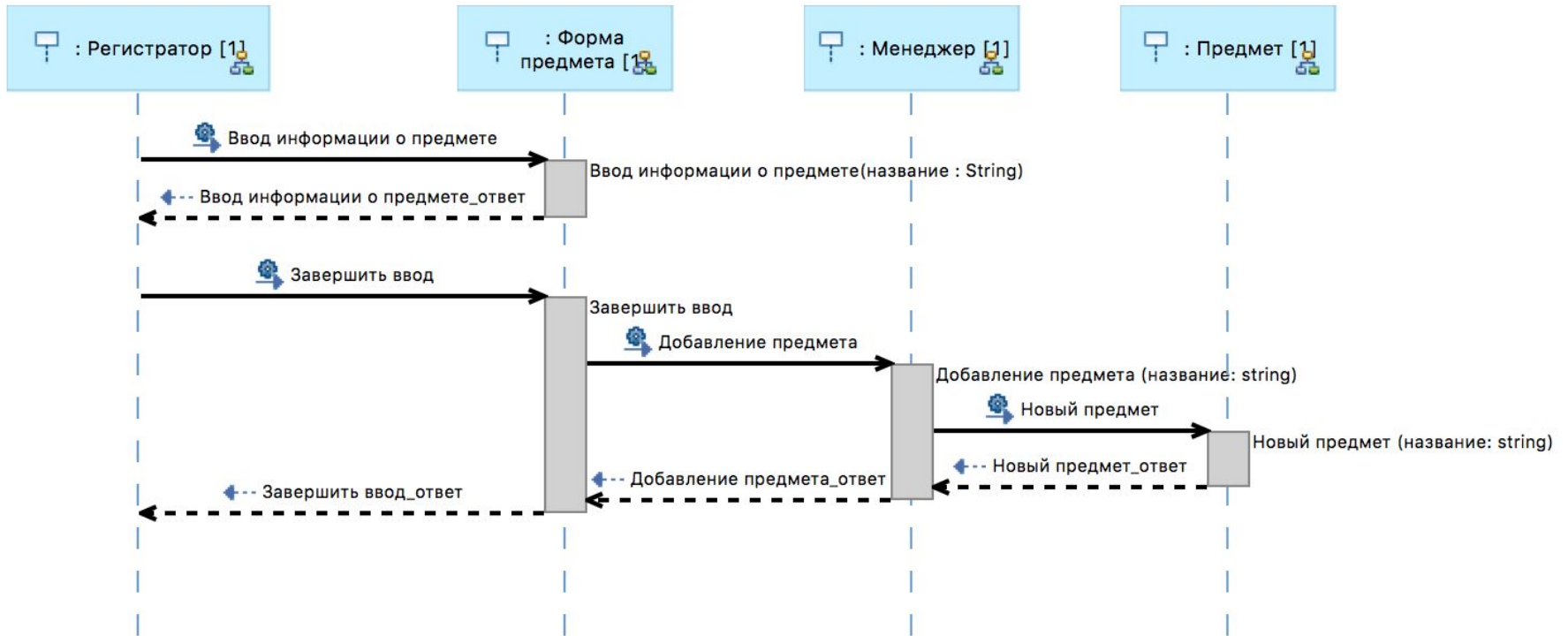
# Асинхронные сообщения

- Если клиент посылает асинхронное сообщение, то после отправки сообщения он продолжает выполнение своей деятельности.
- Обычно на начальном этапе проектирования все сообщения изображаются как синхронные.
- Применение асинхронных сообщений нужно для того, чтобы показать, что различные объекты могут выполнять свою деятельность параллельно друг с другом.

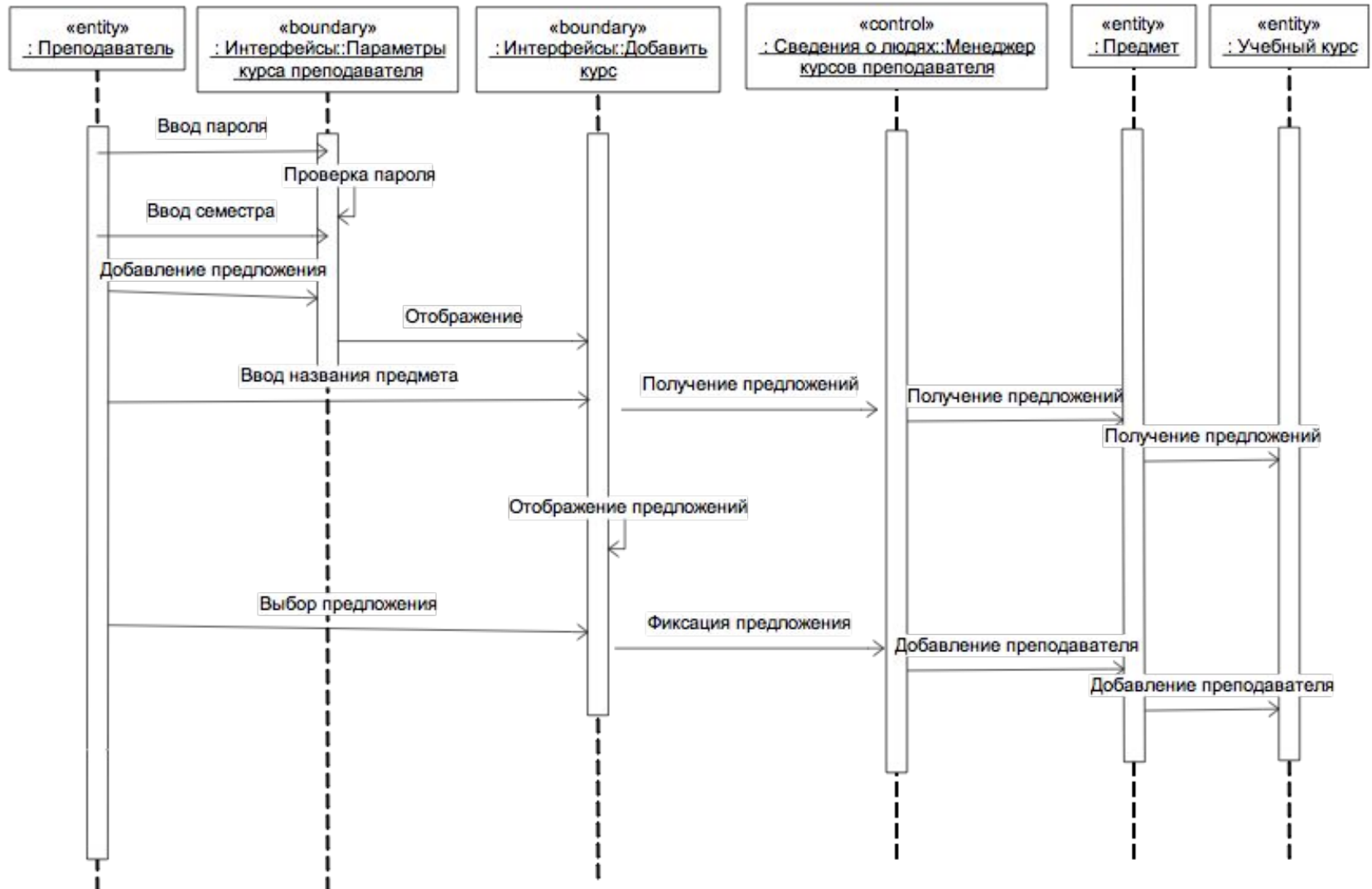
# Пример асинхронного сообщения

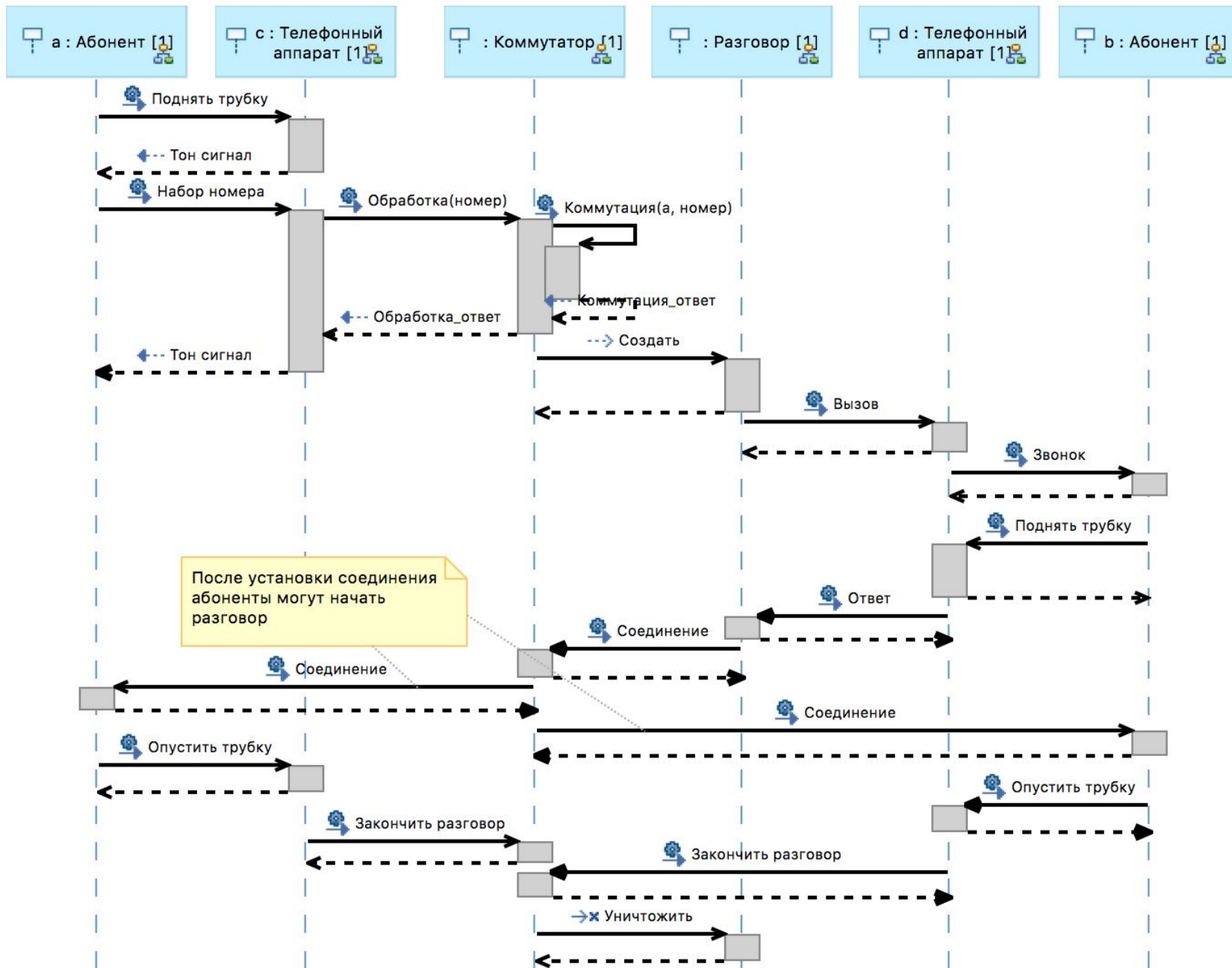


# Создание учебного предмета



# Добавление учебного курса



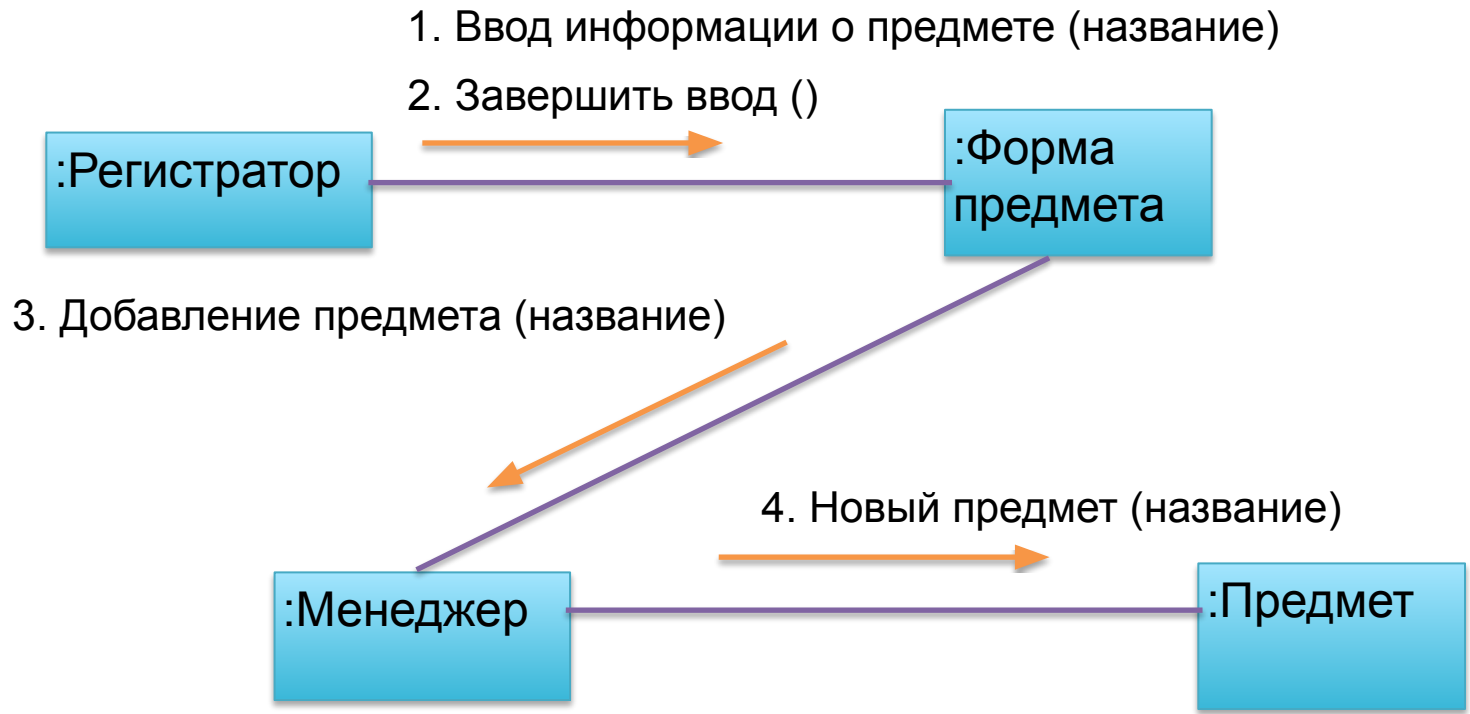


# Диаграмма коммуникации

- Диаграмма коммуникации (communication diagram) используется как альтернативный вариант диаграммы последовательности.
- На диаграмме коммуникации в виде прямоугольников изображаются участвующие во взаимодействии объекты, содержащие имя объекта и его класс или одно имя класса.
- На диаграмме указываются ассоциации между объектами.
- На диаграмме изображаются потоки сообщений.
- Для сообщения указывается его имя и направление (с помощью стрелки).
- Последовательность передачи сообщений определяется порядковыми номерами, стоящими перед именем сообщения.
- UML Designer не поддерживает диаграммы коммуникации.



# Создание учебного предмета



# Соединение абонентов

