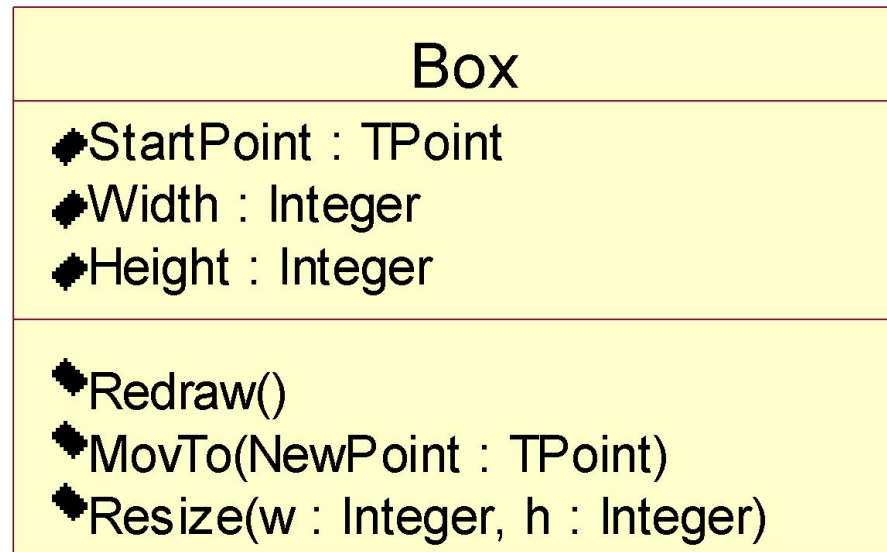


- Работу с языком UML существенно облегчает последовательное использование общих механизмов, перечисленных ниже:
- спецификации (Specifications);
- дополнения (Adornments);
- принятые деления (Common divisions);
- механизмы расширения (Extensibility mechanisms).

Спецификации UML

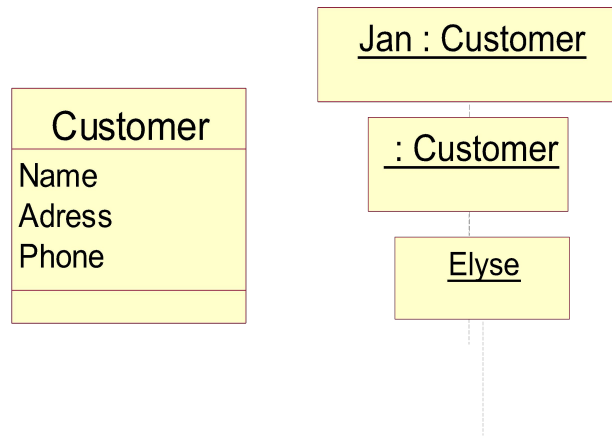
- **Спецификации** UML создают семантический задний план, который полностью включает в себя составные части всех моделей системы, согласованные между собой.



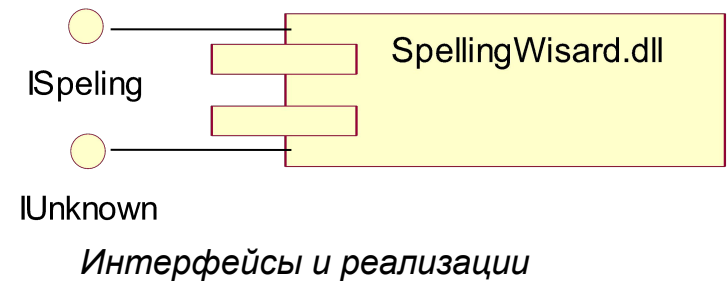
*Подробная
спецификация класса*

Принятые деления

- Прежде всего, существует разделение на классы и объекты

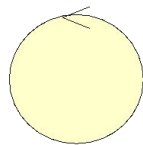


- Еще одним вариантом членения является деление на интерфейс и его реализацию

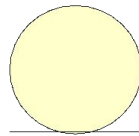


Стереотип

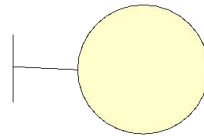
- **Стереотип** (Stereotype) расширяет словарь UML, позволяя на основе существующих блоков языка создавать новые, специфичные для решения конкретной проблемы.



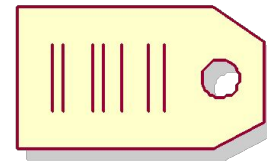
T_Control



T_Entity



T_Boundary



T_Resource

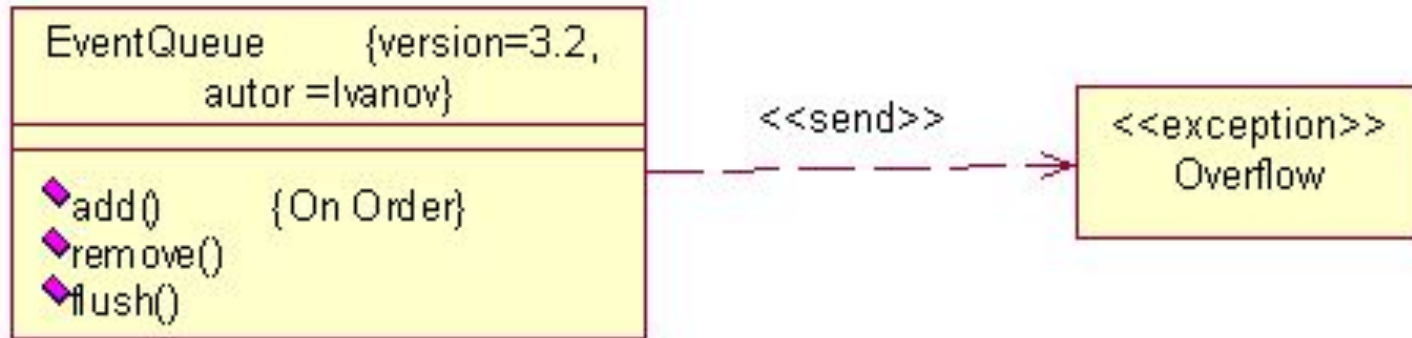
Помеченное значение

- ***Помеченное значение*** (Tagged value) расширяет свойства строительных блоков UML, позволяя включать новую информацию в спецификацию элемента.

Ограничения

- *Ограничения* (Constraints) расширяют семантику строительных блоков UML, позволяя определять новые или изменять существующие правила.

Механизмы расширения. Пример



Диаграммы

UML выделяют девять типов диаграмм:

- диаграммы классов
- диаграммы объектов;
- диаграммы прецедентов;
- диаграммы последовательностей;
- диаграммы кооперации;
- диаграммы состояний;
- диаграммы действий;
- диаграммы компонентов;
- диаграммы развертывания.

Диаграмма классов

- На диаграмме классов показывают классы, интерфейсы, объекты и кооперации, а также их отношения. При моделировании объектно-ориентированных систем этот тип диаграмм используют чаще всего. Диаграммы классов соответствуют статическому виду системы с точки зрения проектирования. Диаграммы классов, которые включают активные классы, соответствуют статическому виду системы с точки зрения процессов.

Диаграмма классов

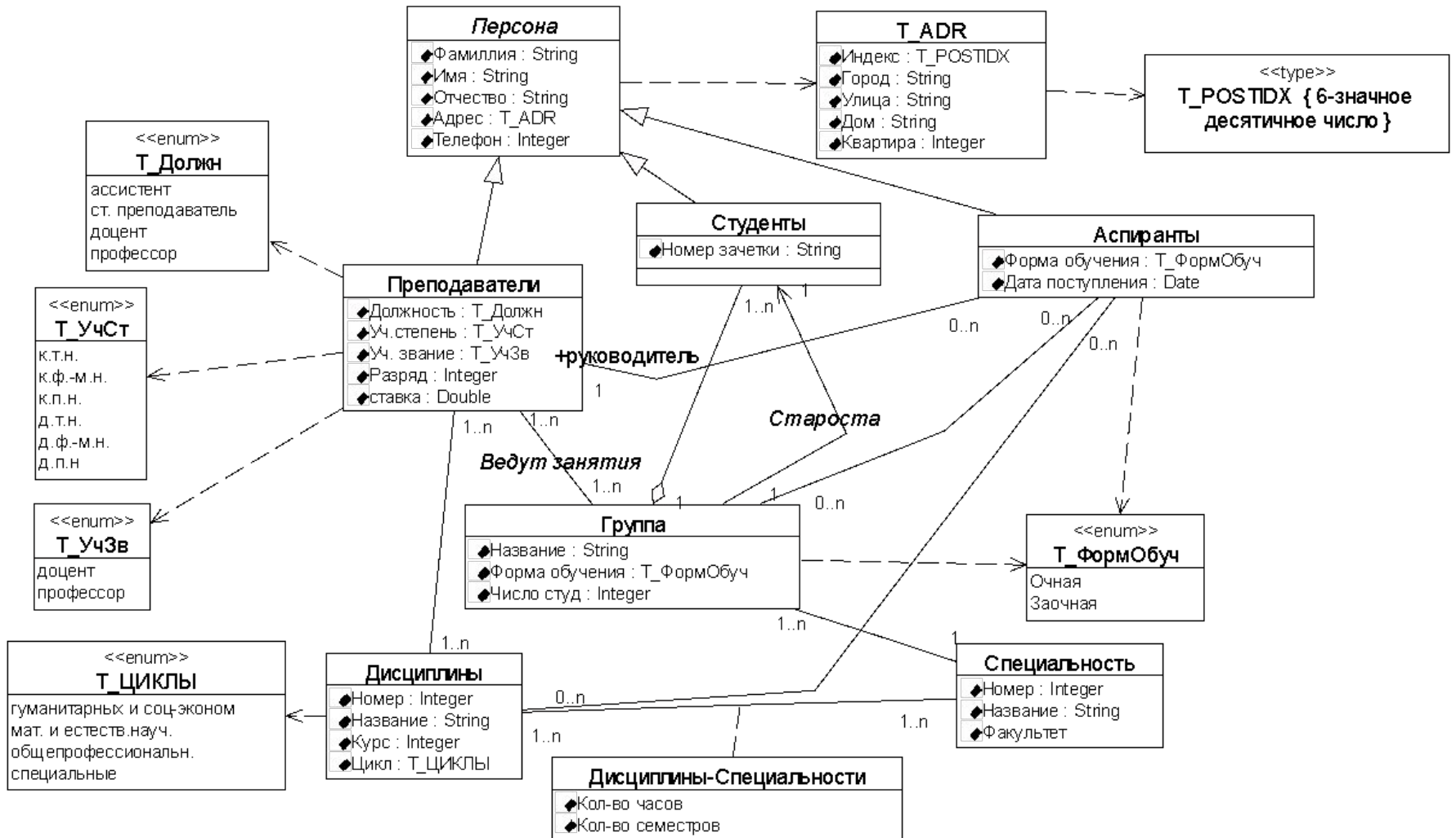


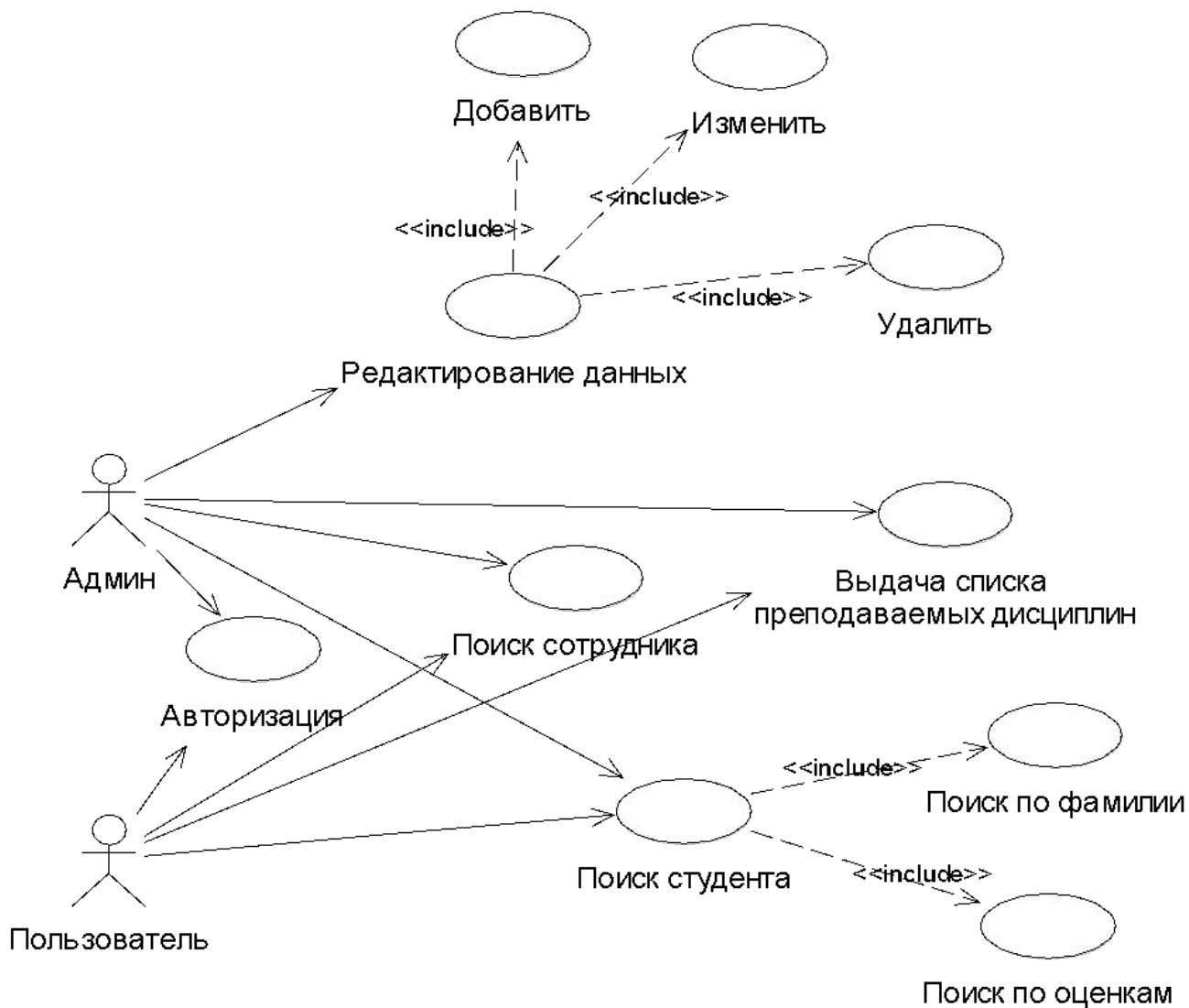
Диаграмма объектов

- На диаграмме объектов представлены объекты и отношения между ними. Они являются статическими «фотографиями» экземпляров сущностей, показанных на диаграммах классов. Диаграммы объектов, как и диаграммы классов, относятся к статическому виду системы с точки зрения проектирования или процессов, но с расчетом на настоящую или макетную реализацию.

Диаграммы прецедентов

- На диаграмме прецедентов представлены прецеденты и актеры (частный случай классов), а также отношения между ними. Диаграммы прецедентов относятся к статическому виду системы с точки зрения прецедентов использования. Они особенно важны при организации и моделировании поведения системы.

Диаграмма прецедентов



Диаграммы взаимодействий

- Диаграммы последовательностей и кооперации являются частными случаями диаграмм взаимодействия. На диаграммах взаимодействия представлены связи между объектами; показаны, в частности, сообщения, которыми объекты могут обмениваться. Диаграммы взаимодействия относятся к динамическому виду системы. При этом диаграммы последовательности отражают временную упорядоченность сообщений, а диаграммы кооперации – структурную организацию обменивающихся сообщениями объектов.

Диаграмма последовательности

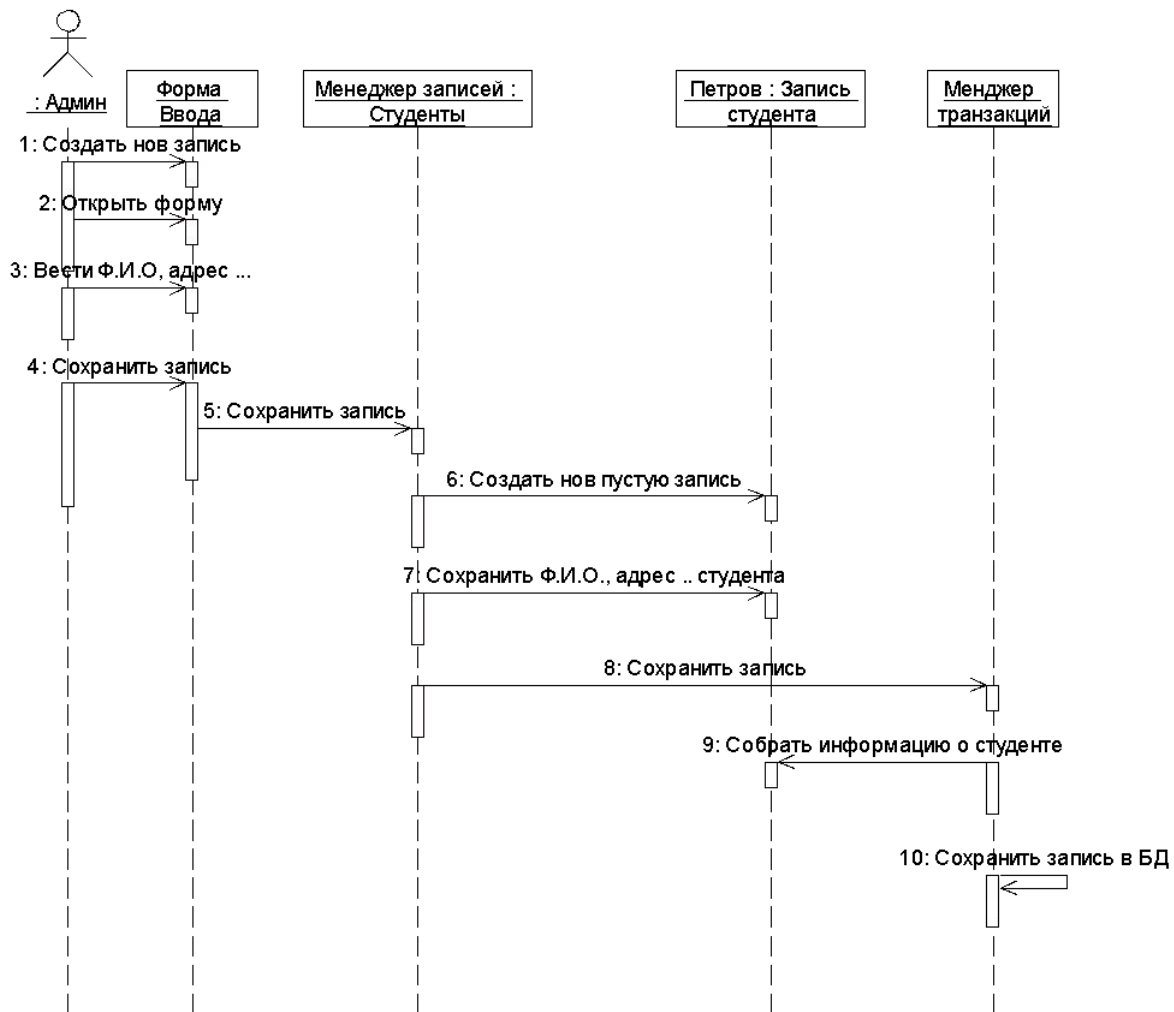
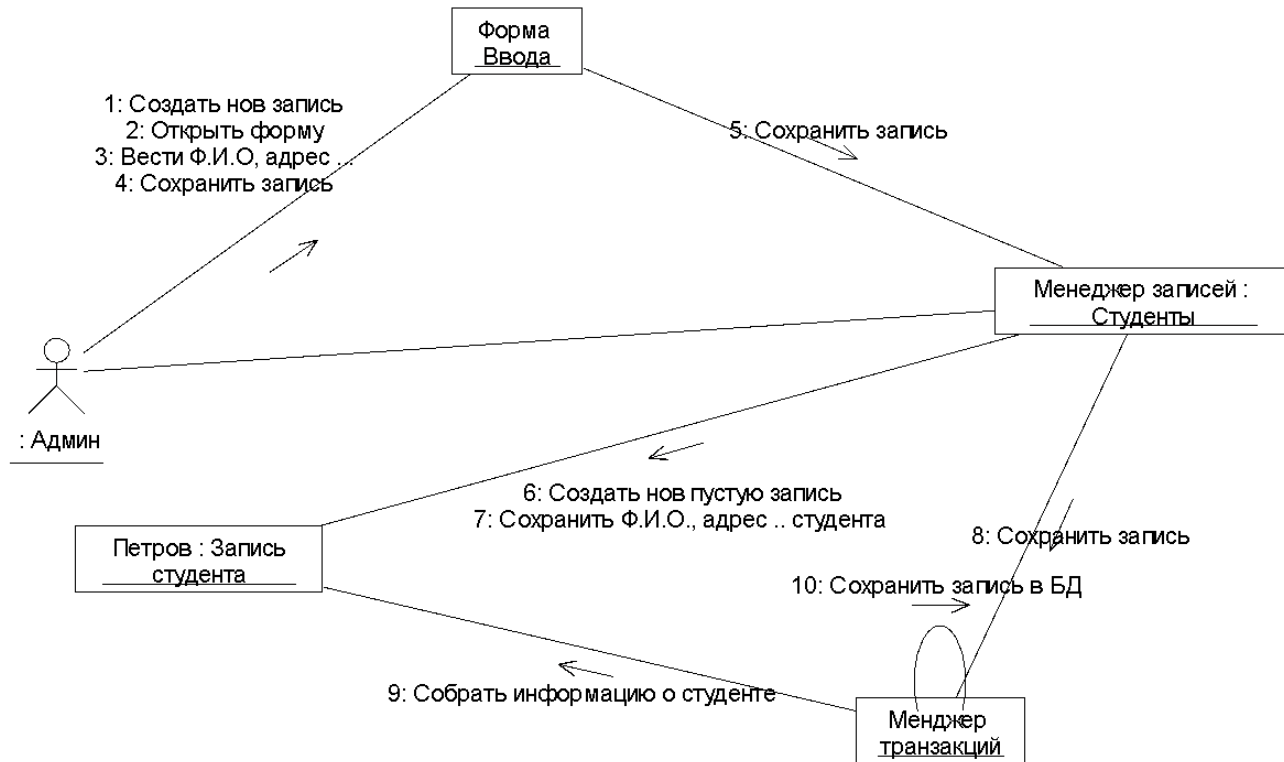


Диаграмма кооперации



Диаграммы состояний

- На диаграммах состояний (Statechart diagrams) представлен автомат, включающий в себя состояния, переходы, события и виды действий. Диаграммы состояний относятся к динамическому виду системы; особенно они важны при моделировании поведения интерфейса, класса или кооперации. Они акцентируют внимание на поведении объекта, зависящем от последовательности событий, что очень полезно для моделирования реактивных систем.

Диаграмма состояний

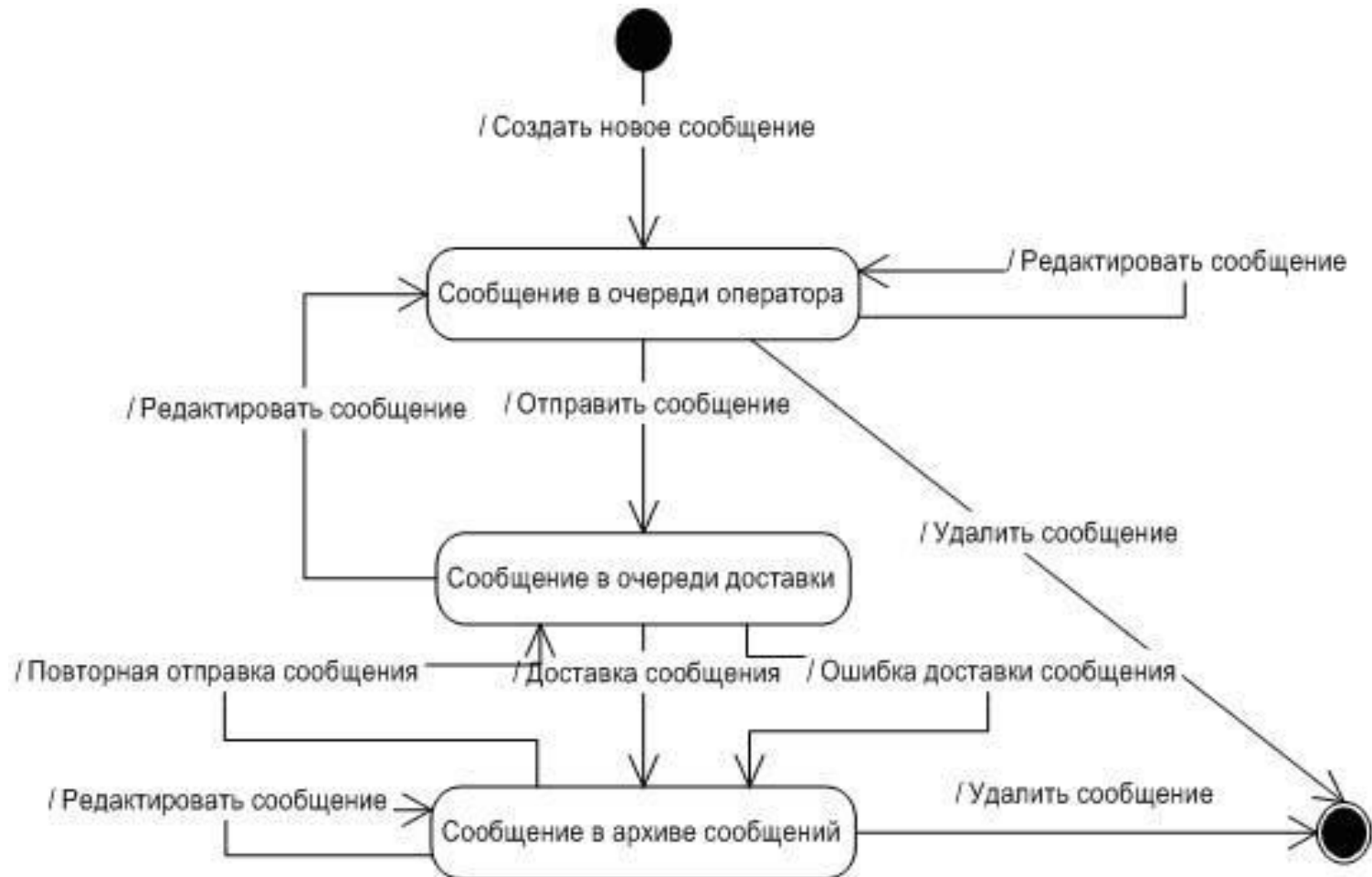


Диаграмма деятельности

- Диаграмма деятельности – это частный случай диаграммы состояний; на ней представлены переходы потока управления от одной деятельности к другой внутри системы. Диаграммы деятельности относятся к динамическому виду системы; они наиболее важны при моделировании ее функционирования и отражают поток управления между объектами.

Диаграмма деятельности

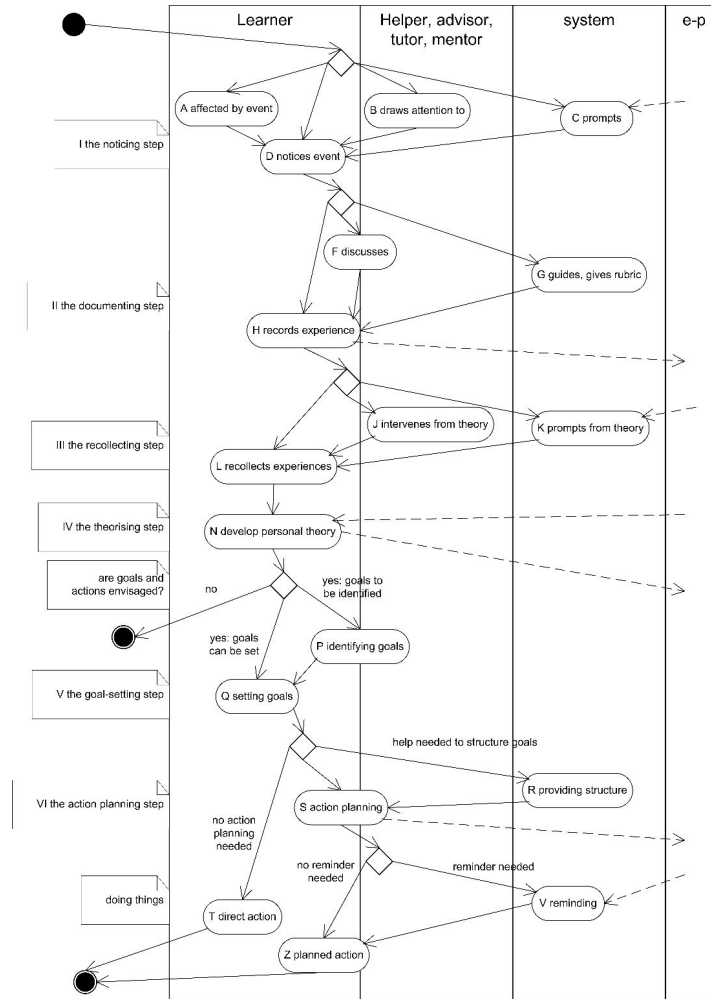


Диаграмма компонентов

- На диаграмме компонентов представлена организация совокупности компонентов и существующие между ними зависимости. Диаграммы компонентов относятся к статическому виду системы с точки зрения реализации. Они могут быть соотнесены с диаграммами классов, так как компонент обычно отображается на один или несколько классов, интерфейсов или коопераций.

Диаграмма КОМПОНЕНТОВ

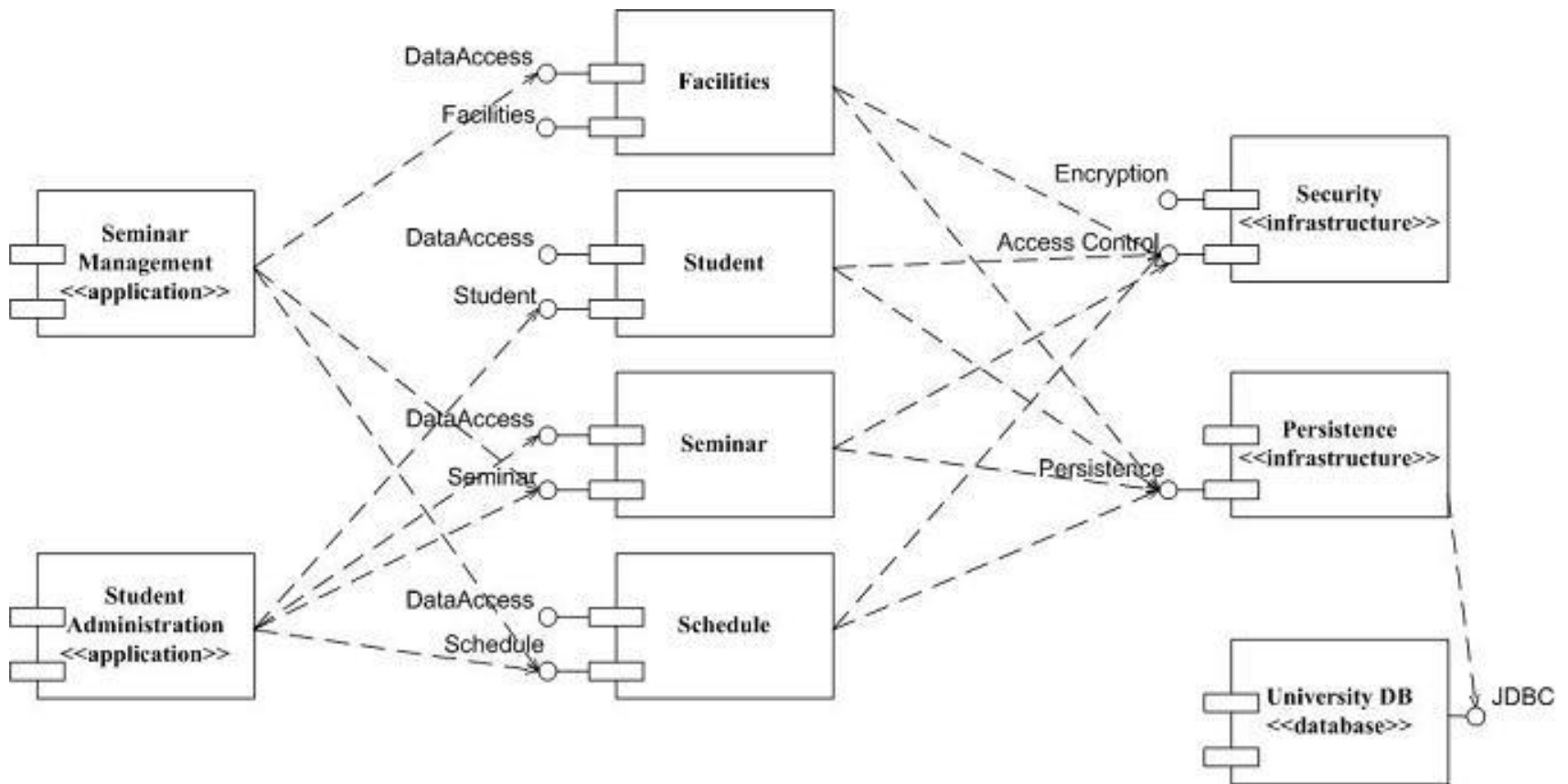
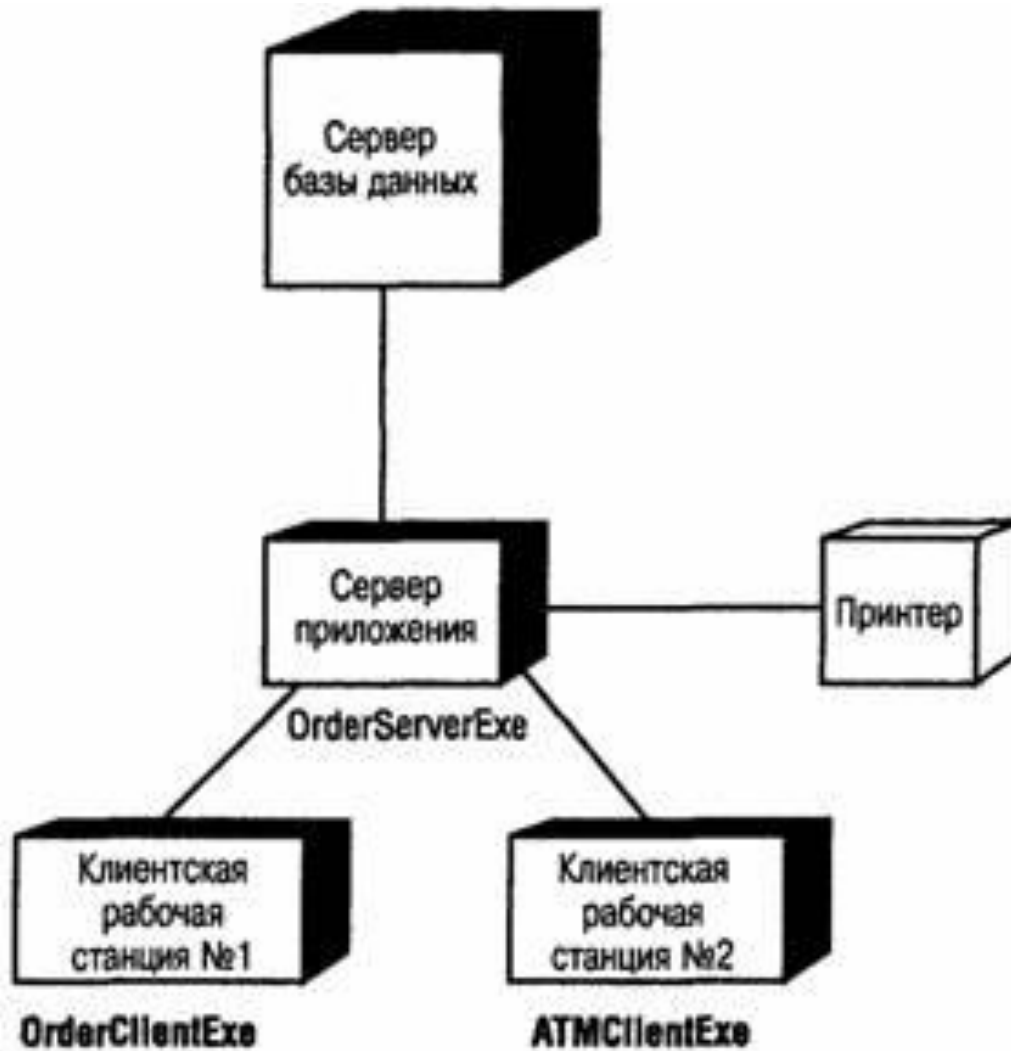


Диаграмма развертывания

- На диаграмме развертывания представлена конфигурация обрабатывающих узлов системы и размещенных в них компонентов. Диаграммы развертывания относятся к статическому виду архитектуры системы с точки зрения развертывания. Они связаны с диаграммами компонентов, поскольку в узле обычно размещаются один или несколько компонентов.

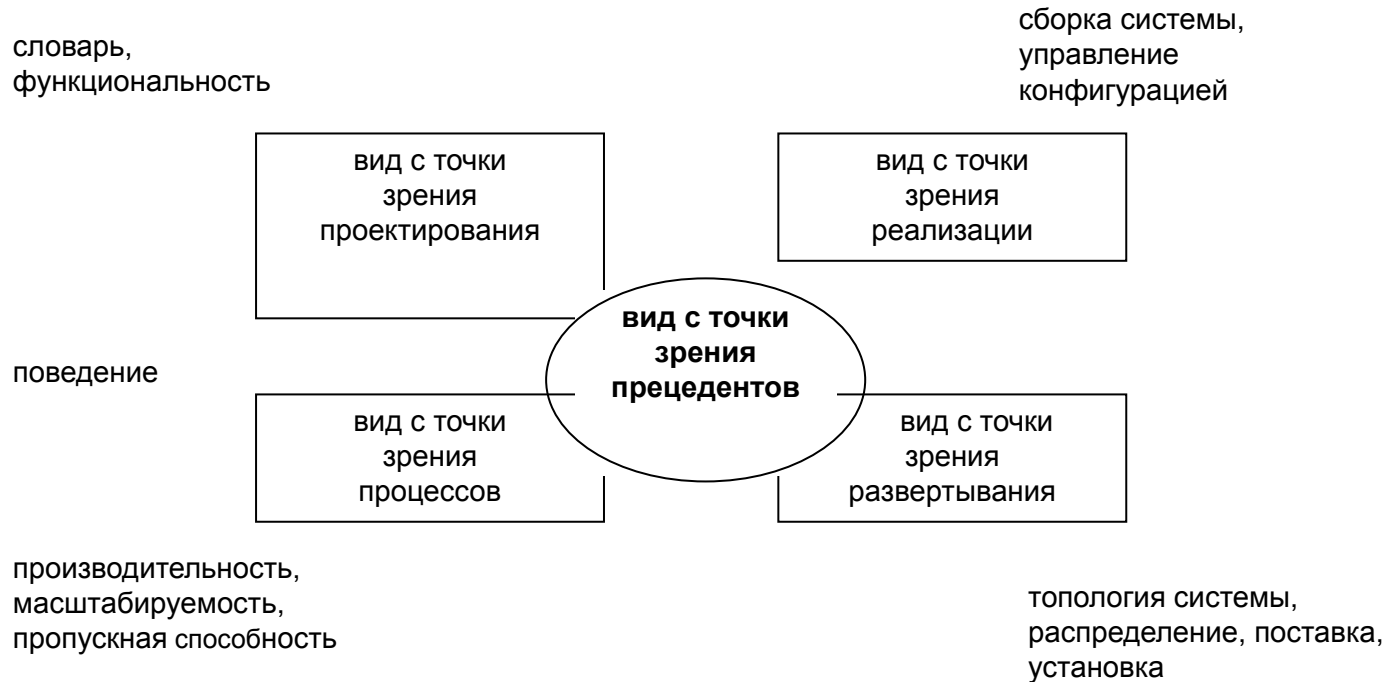
Диаграмма развертывания



Архитектура системы

- Архитектура – это совокупность существенных решений касательно следующих аспектов:
- организации программной системы;
- выбора структурных элементов, составляющих систему, и их интерфейсов;
- поведения этих элементов, специфицированного в кооперациях с другими элементами;
- составления из этих структурных и поведенческих элементов все более и более крупных подсистем;
- архитектурного стиля, направляющего и определяющего всю организацию системы: статические и динамические элементы, их интерфейсы, кооперации и способ их объединения.

Моделирование системной архитектуры

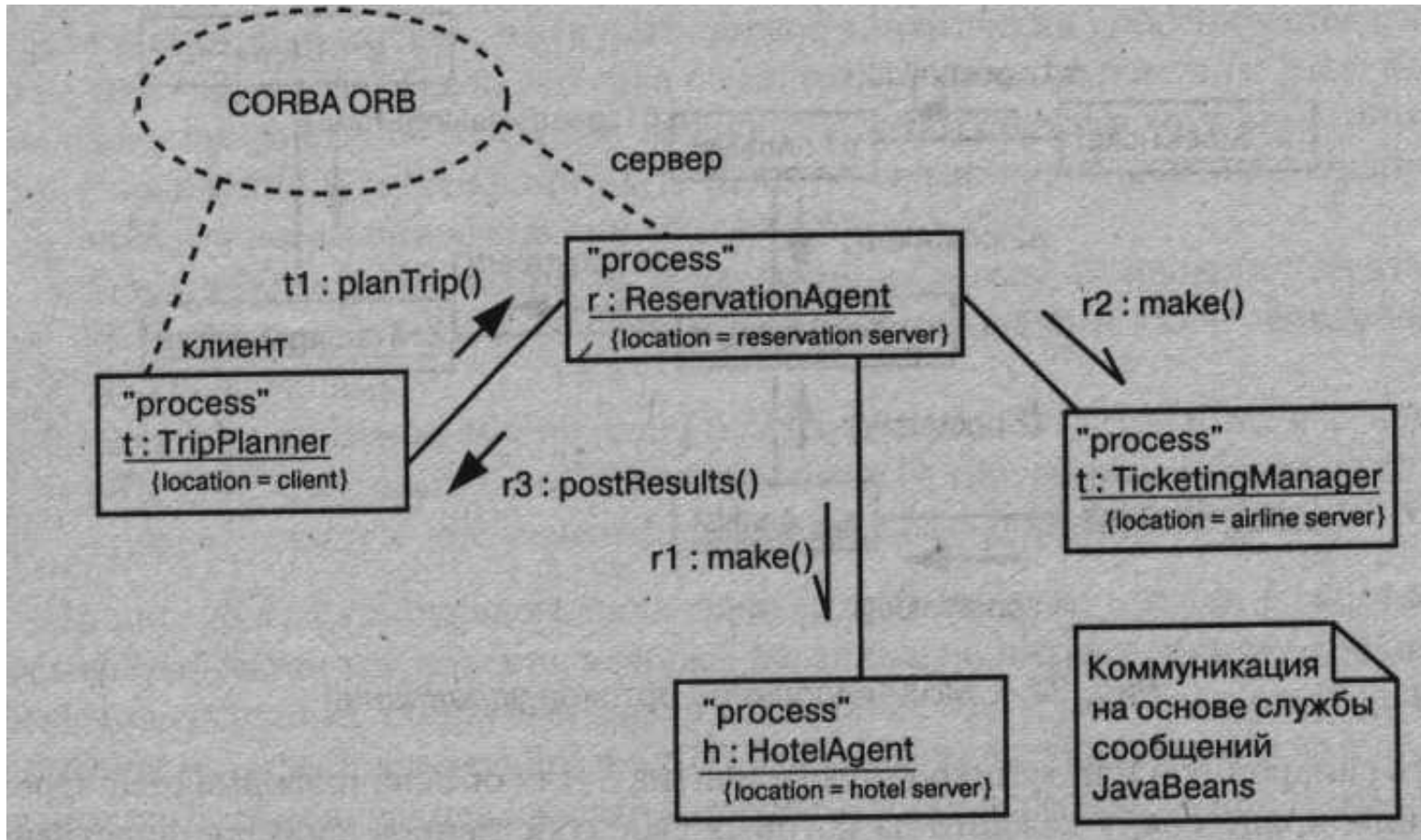


- Вид с точки зрения прецедентов (Use case view) охватывает прецеденты, которые описывают поведение системы, наблюдаемое конечными пользователями, аналитиками и тестировщиками. Этот вид специфицирует не истинную организацию программной системы, а те движущие силы, от которых зависит формирование системной архитектуры. В языке UML статические аспекты этого вида передаются диаграммами прецедентов, а динамические – диаграммами взаимодействия, состояний и действий.

- Вид с точки зрения проектирования (Design view) охватывает классы, интерфейсы и кооперации, формирующие словарь задачи и ее решения. Этот вид поддерживает прежде всего функциональные требования, предъявляемые к системе, то есть те услуги, которые она должна предоставлять конечным пользователям. С помощью языка UML статические аспекты этого вида можно передавать диаграммами классов и объектов, а динамические – диаграммами взаимодействия, состояний и действий.

- Вид с точки зрения процессов (Process view) охватывает нити и процессы, формирующие механизмы параллелизма и синхронизации в системе. Этот вид описывает главным образом производительность, масштабируемость и пропускную способность системы. В UML его статические и динамические аспекты визуализируются теми же диаграммами, что и для вида с точки зрения проектирования, но особое внимание при этом уделяется активным классам, которые представляют соответствующие нити и процессы.

Процессы



- Вид с точки зрения реализации (Implementation view) охватывает компоненты и файлы, используемые для сборки и выпуска конечного программного продукта. Этот вид предназначен в первую очередь для управления конфигурацией версий системы, составляемых из независимых (до некоторой степени) компонентов и файлов, которые могут по-разному объединяться между собой. В языке UML статические аспекты этого вида передают с помощью диаграмм компонентов, а динамические - с помощью диаграмм взаимодействия, состояний и действий.

- Вид с точки зрения развертывания (Deployment view) охватывает узлы, формирующие топологию аппаратных средств системы, на которой она выполняется. В первую очередь он связан с распределением, поставкой и установкой частей, составляющих физическую систему. Его статические аспекты описываются диаграммами развертывания, а динамические - диаграммами взаимодействия, состояний и действий.