

ЛЕКЦИЯ 8

Диаграмма композитной структуры

Диаграмма пакетов

Диаграмма объектов

Композитная структура

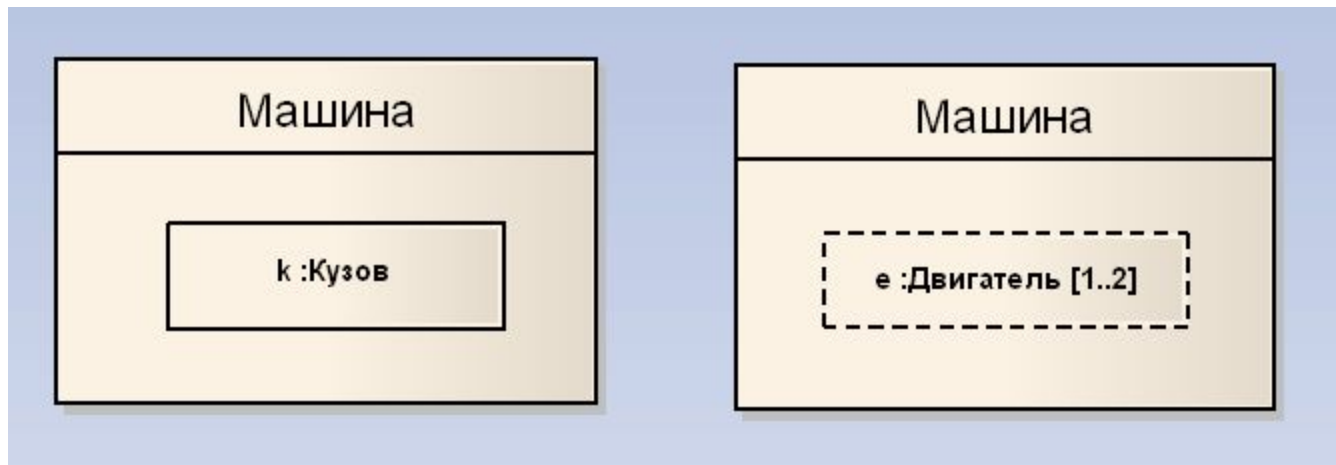
Внутренняя структура – это структура взаимодействующих элементов модели, которые создаются в экземпляре содержащего их классификатора.

Свойство – множество экземпляров, которые являются собственностью содержащего их экземпляра классификатора.

Для представления внутренней структуры классификаторов в общем случае предназначена диаграмма композитной структуры. При этом классификатор, имеющий некоторую внутреннюю структуру, называют также композитным, а его внутреннюю структуру – композитной структурой.

Композитный класс

Часть – свойство, которое является элементом внутренней структуры композитного классификатора, в частном случае – класса.

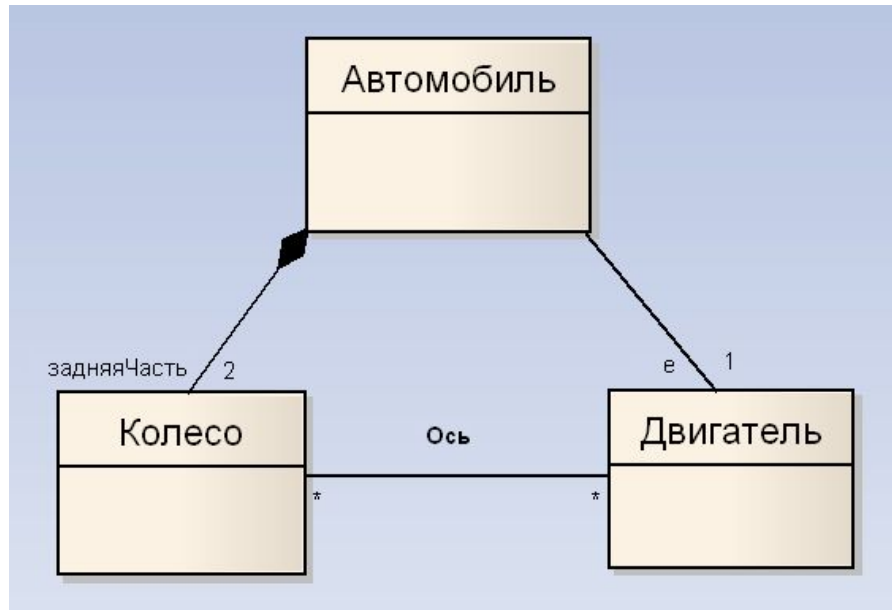


Композитный класс

Символ прямоугольника части или свойства имеет только секцию имени, содержащую строку текста, которая должна удовлетворять следующему синтаксису в БНФ:

$$\langle \text{имя-части} \rangle ::= [\langle \text{собственное имя части} \rangle] [: \langle \text{имя-класса} \rangle] \\ [[\langle \text{кратность} \rangle]] [\langle \text{имя-класса} \rangle]$$

Композитный класс



Соединитель

Соединитель – отношение, которое обеспечивает взаимосвязь или коммуникацию между двумя или более экземплярами классификаторов, в частном случае – экземплярами классов.

Соединитель изображается с использованием нотации для ассоциации. Необязательная строка имени соединителя должна удовлетворять следующему синтаксису:

`<имя-соединителя> ::= ([имя] ':' <имя-класса>) | <имя>`

Конец соединителя является концевой точкой соединителя, которой соединитель присоединяется к соединяемому элементу.

Роли

На диаграммах композитной структуры могут быть также изображены спецификации экземпляров класса. При этом для них должны быть указаны имена ролей, которые они играют в композитном классе.

Строка имени роли в спецификации экземпляра должна удовлетворять следующему синтаксису БНФ:

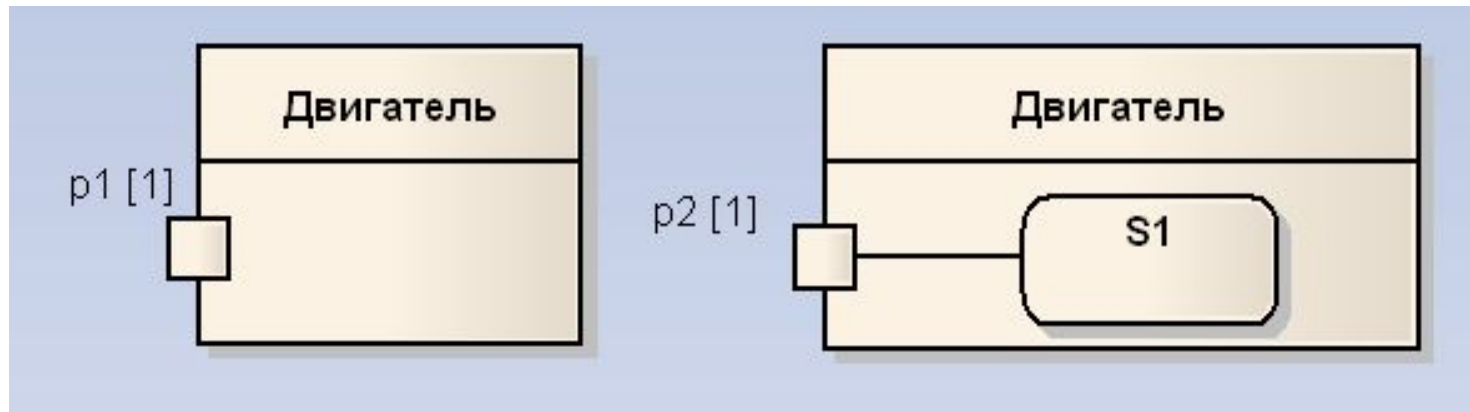
```
<имя-роли-в-спецификации-экземпляра> ::=  
  {<имя>[ '/' <имя-роли>] | '/' <имя-роли>}  
  [ ':' <имя-классификатора> [ ',' <имя-классификатора>]*]
```

Композитная структура



Порт класса

Порт – свойство классификатора, которое специфицирует отдельную точку взаимодействия между этим классификатором и его окружением или между классификатором и его внутренними частями.



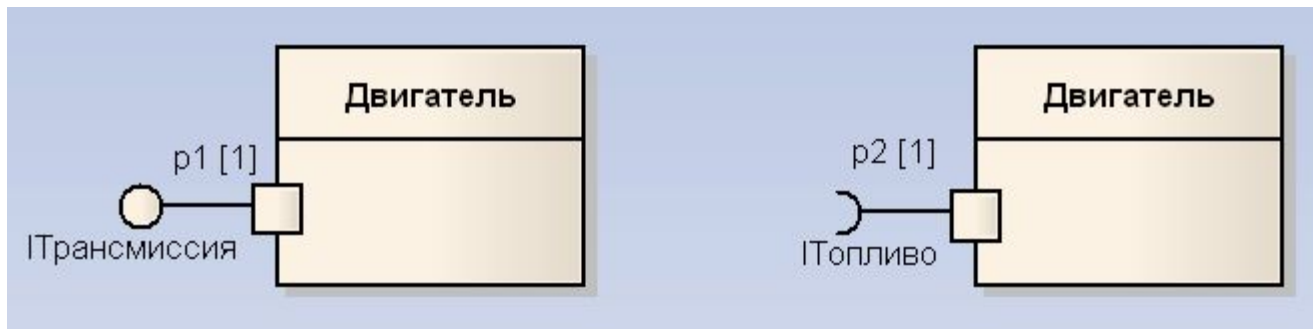
Порт класса

С портами могут быть ассоциированы интерфейсы для спецификации характера взаимодействия, которое может осуществляться через этот порт.

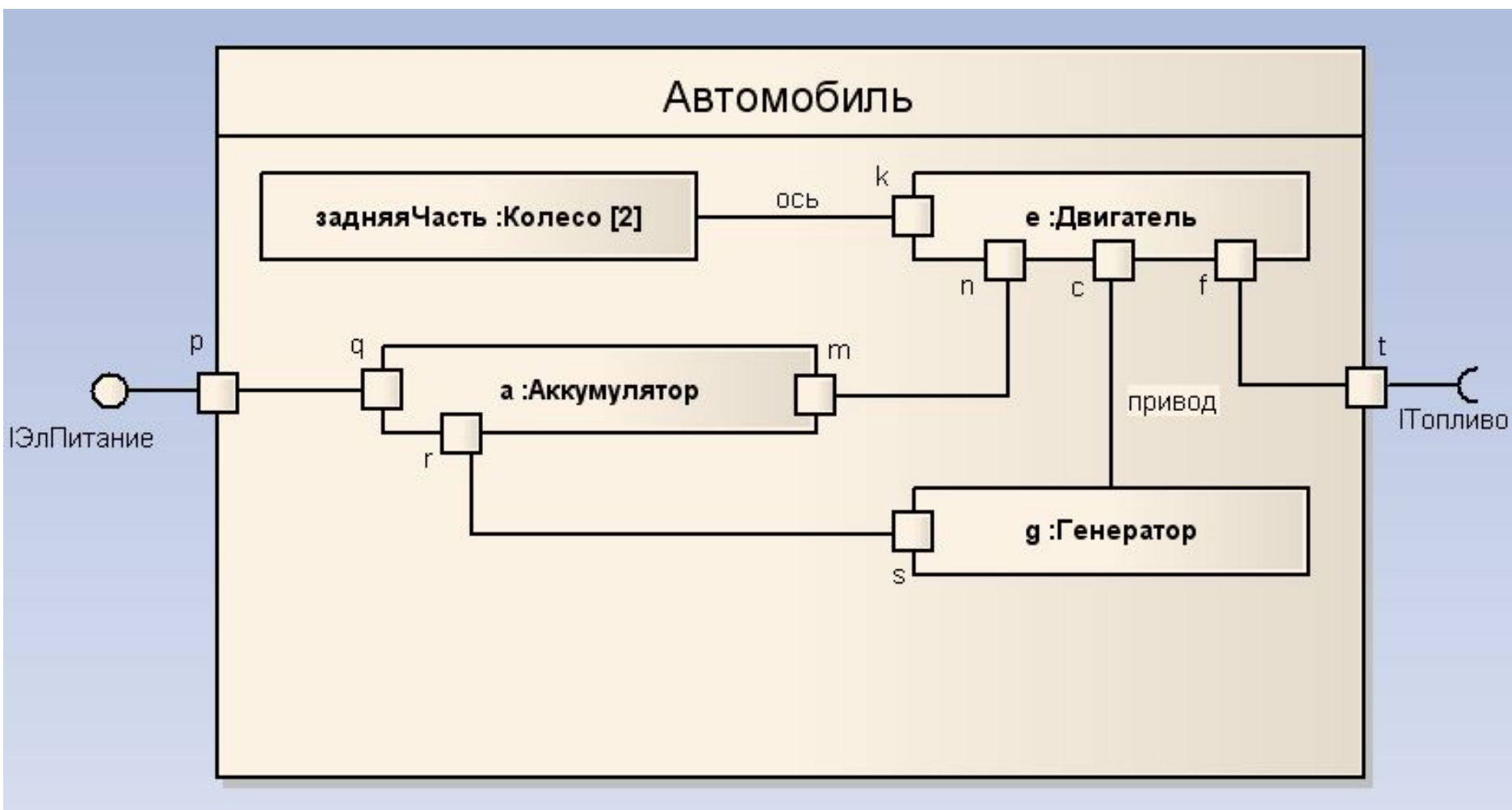
В этом случае все взаимодействия через этот порт специфицируются в форме и предоставляемых и требуемых интерфейсов.

Предоставляемый интерфейс порта характеризует запросы, которые могут быть переданы через этот порт классу от его окружения.

Требуемый интерфейс порта характеризует запросы, которые могут быть переданы от класса к его окружению через этот порт.



Пример



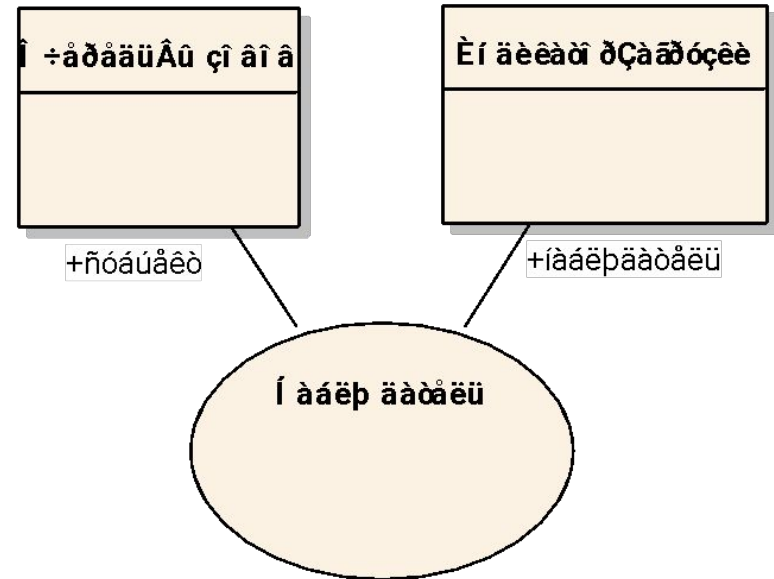
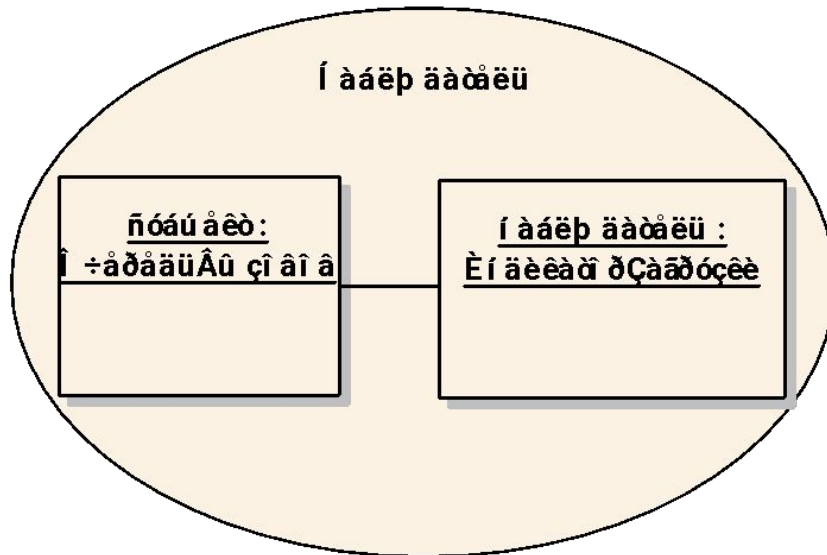
Кооперация

Кооперация — классификатор, предназначенный для описания некоторой структуры элементов или ролей, которые выполняют специализированные функции и совместно производят желаемую функциональность.

Роль кооперации специфицирует требуемое множество характеристик, которые должен иметь соответствующий участник кооперации.

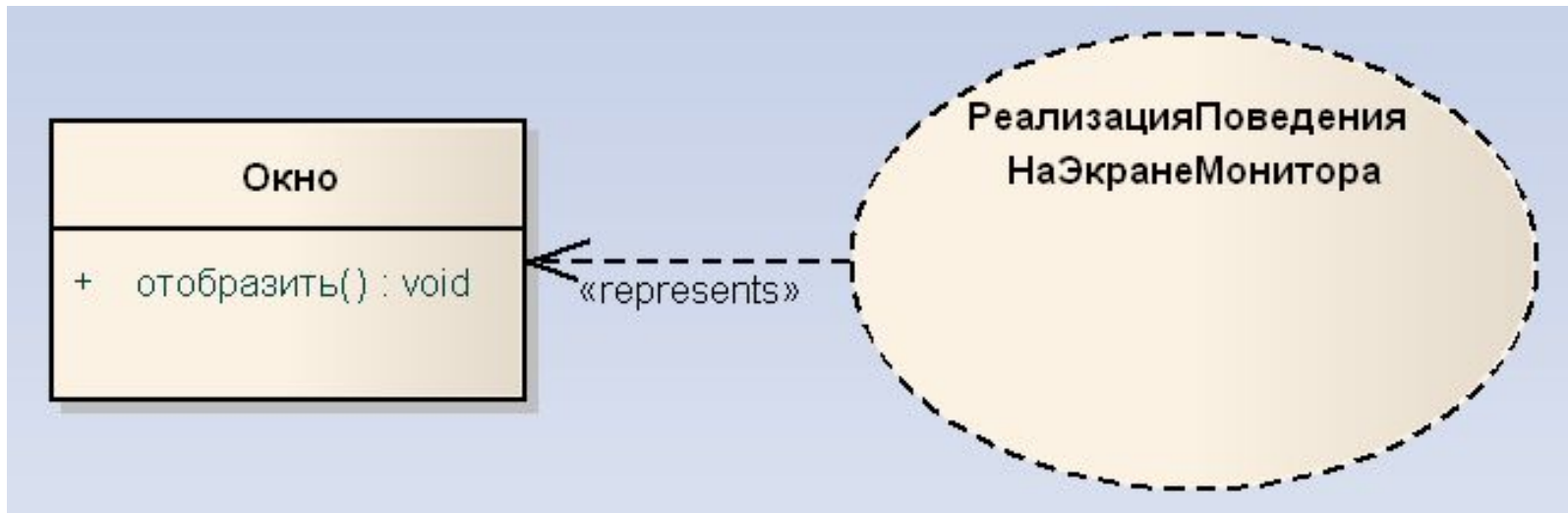
Кооперация

composite structure Èíîîî çèð àÿ äèàäàìî à

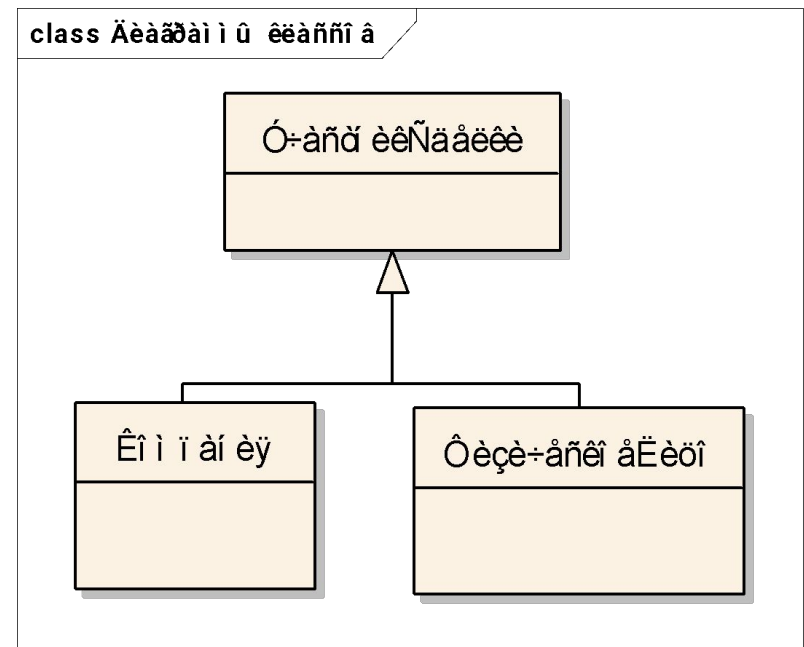
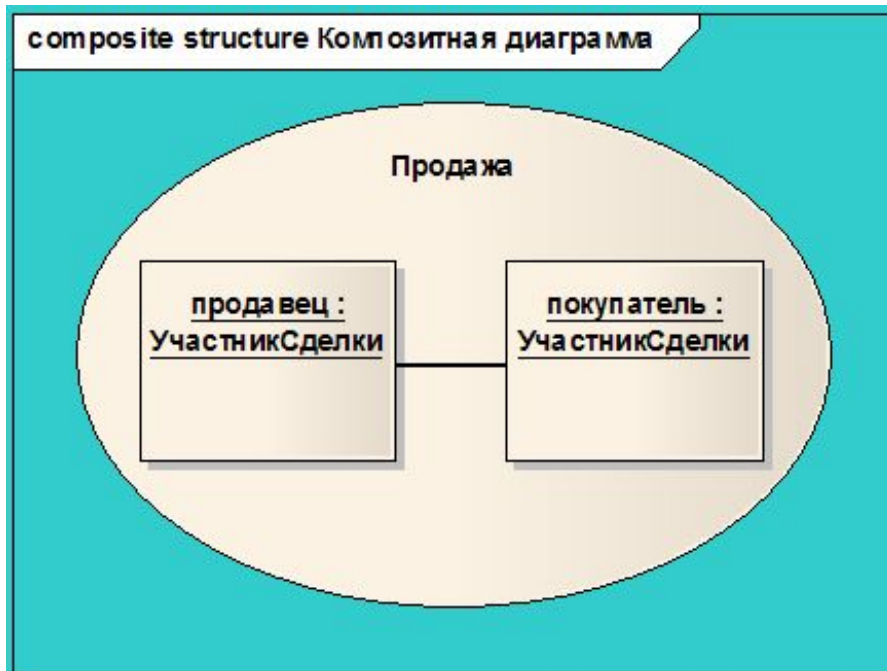


Применение кооперации

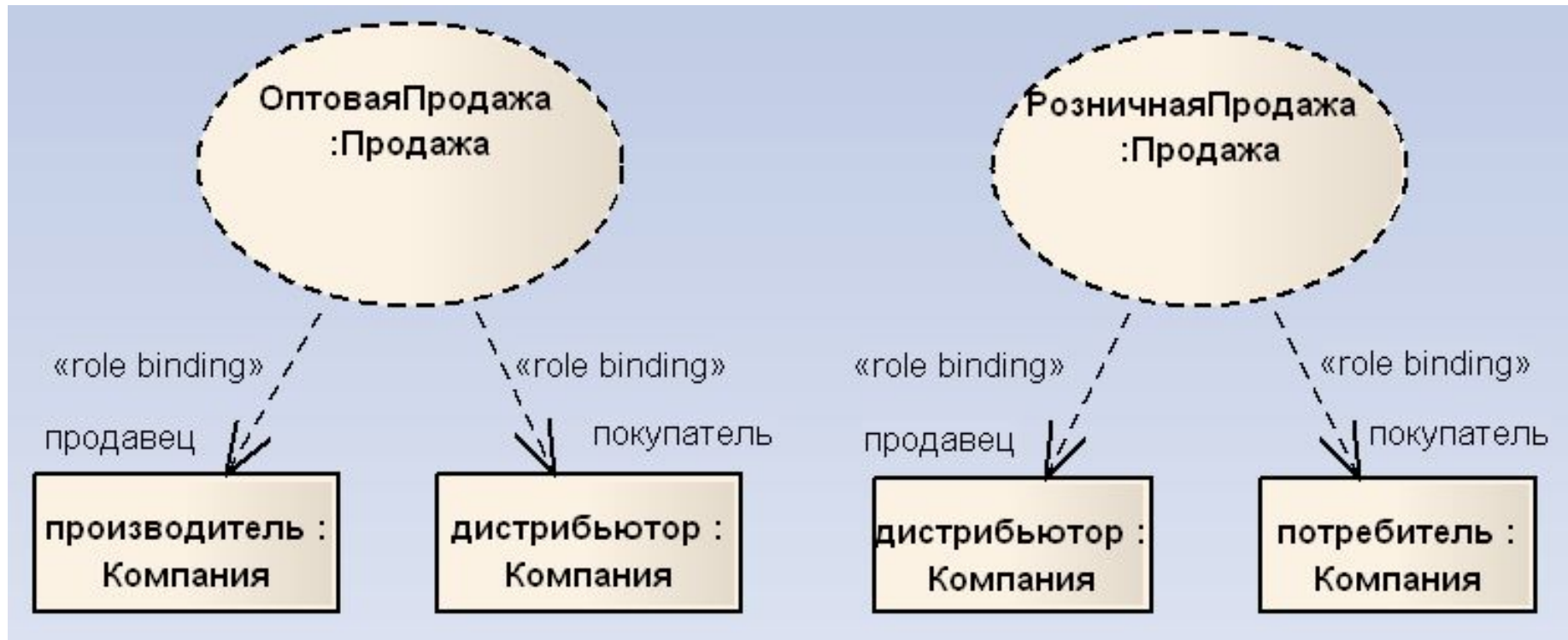
Применение кооперации представляет собой описание реализации кооперации в форме множества взаимодействующих элементов посредством связывания этих элементов с ролями данной кооперации.



Применение кооперации



Применение кооперации



Применение кооперации

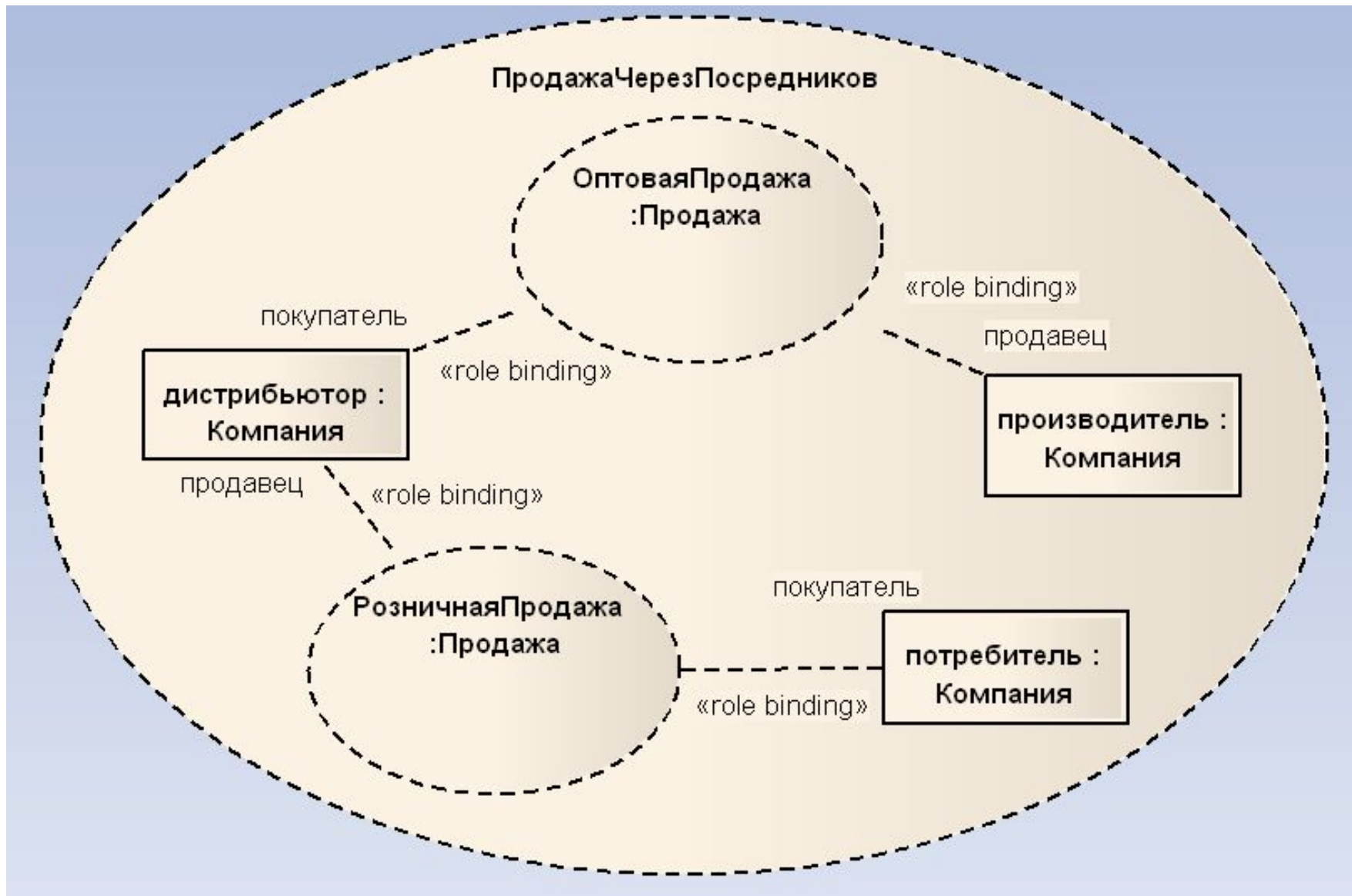


Диаграмма пакетов

Диаграмма пакетов предназначена для представления размещения элементов модели в пакетах и спецификации зависимостей между пакетами и их элементами.

Пакет – элемент модели, используемый для группировки других элементов модели.

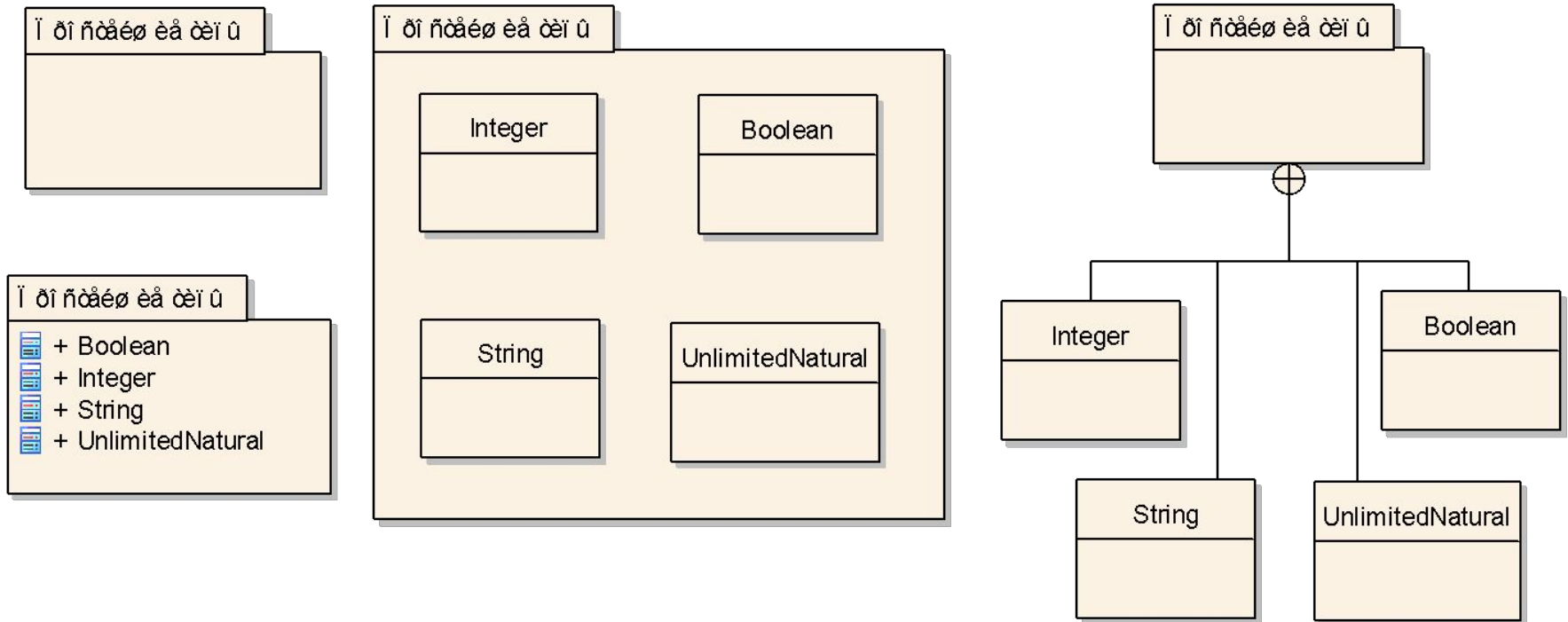
Элементы модели, которые входят в состав некоторого пакета, называются членами этого пакета.

Элементы модели, которые входят в пространство имен некоторого пакета, называются элементами этого пакета.

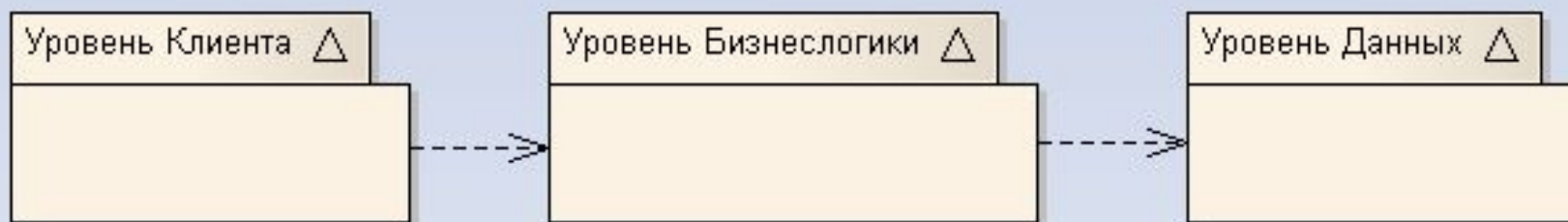
На диаграммах пакетов одни пакеты могут быть вложены в другие пакеты. В этом случае вложенный пакет называется подпакетом, а все элементы подпакета будут также принадлежать любому пакету, для которого рассматриваемый подпакет является вложенным.

Диаграмма пакетов

pkg Æààðàì ì à ÿ àèáä â



Зависимость пакетов



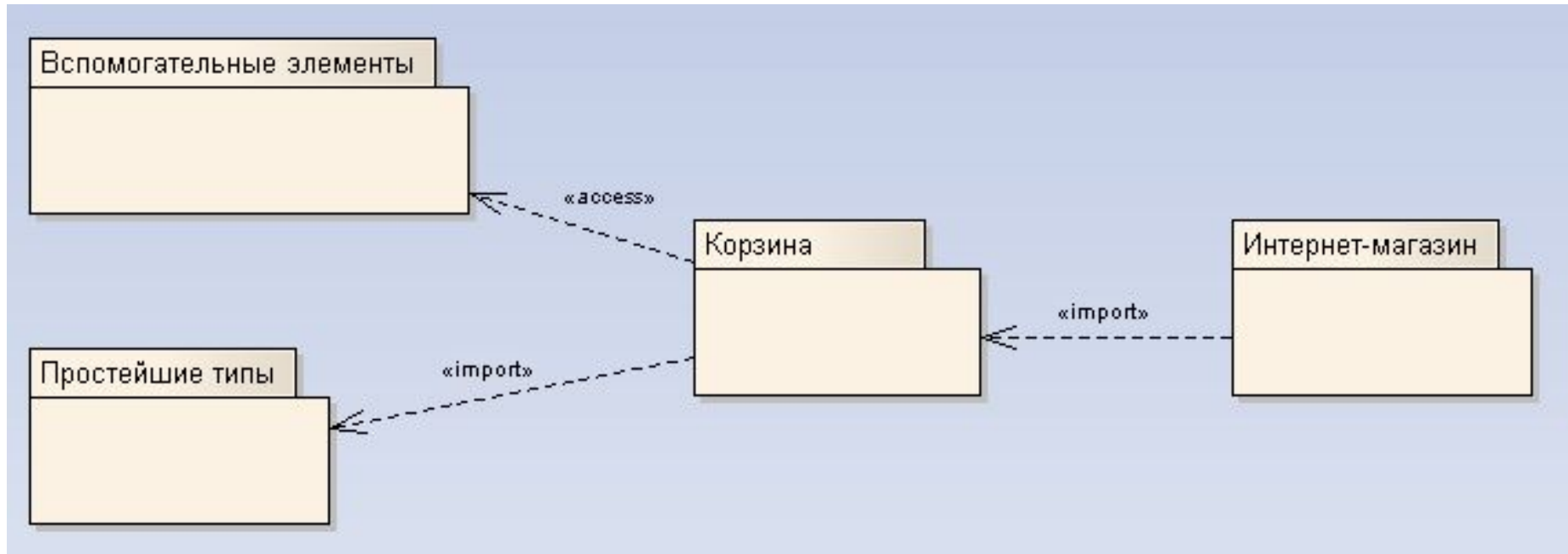
Импорт пакета

Импорт пакета – направленное отношение между пакетами, при котором члены одного пакета могут быть добавлены в пространство имен другого пакета.

Два вида импорта:

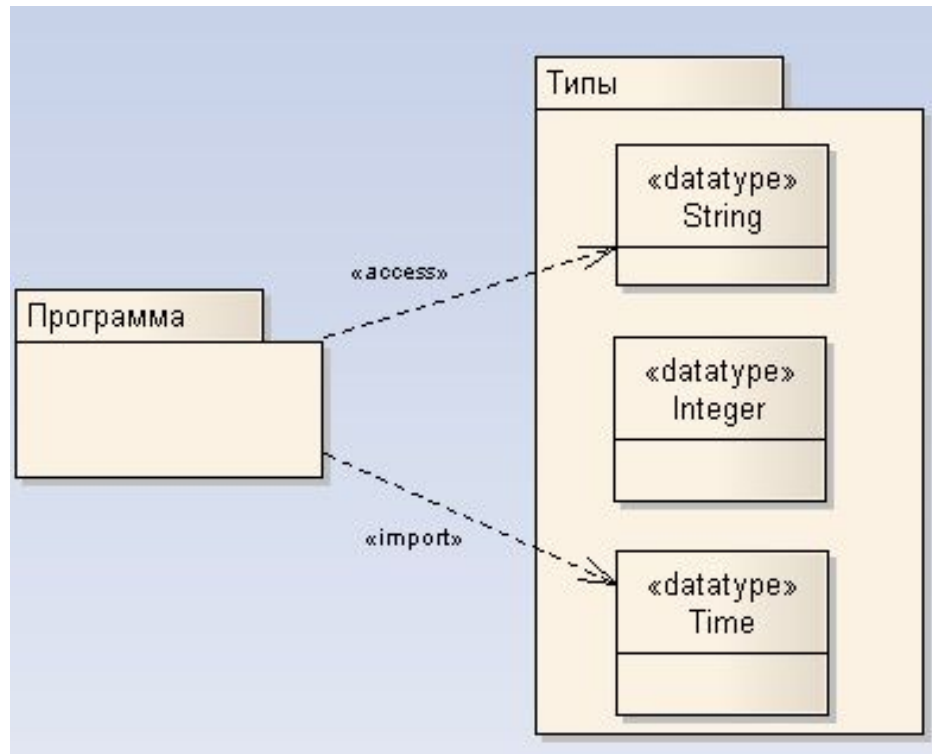
- `<<import>>` - для общедоступного импорта пакета;
- `<<access>>` - для закрытого импорта пакета

Импорт пакета

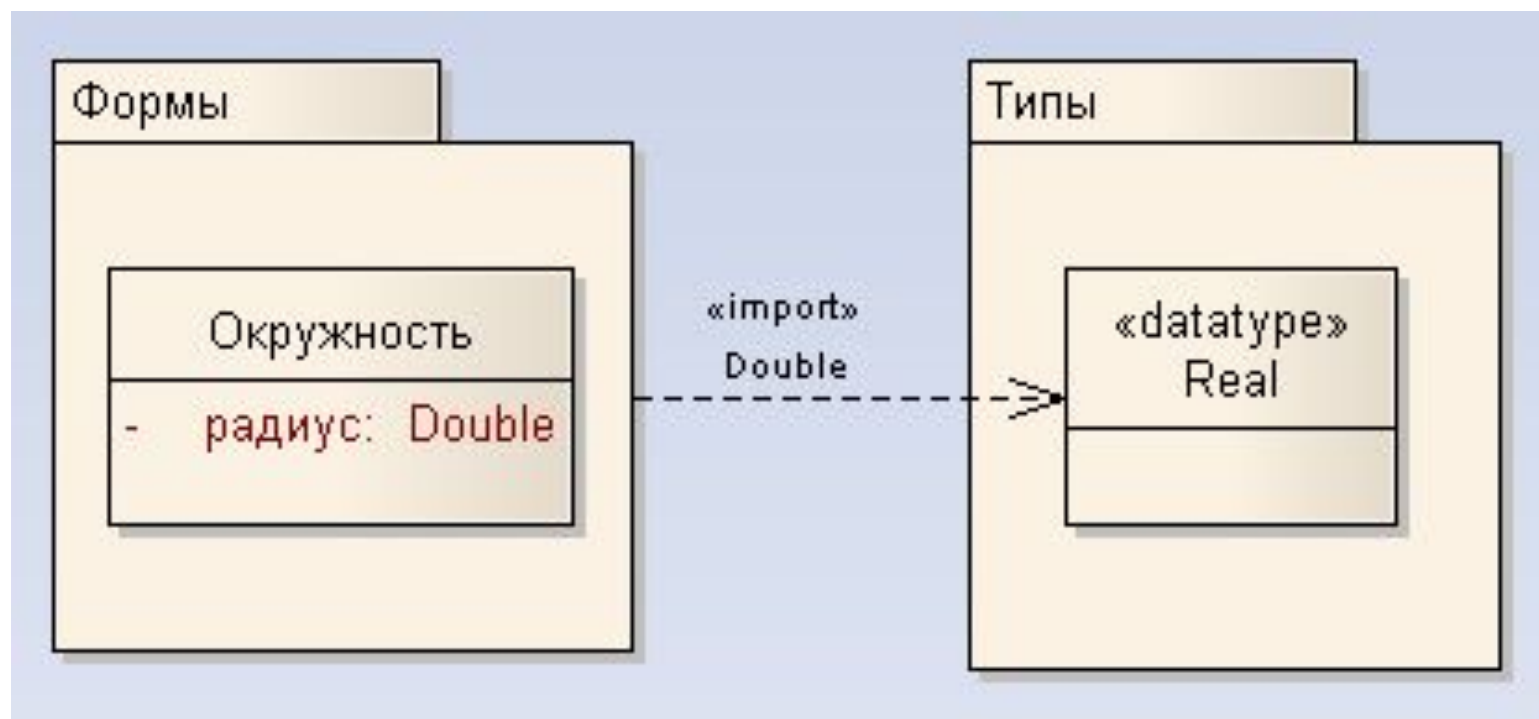


Импорт элемента

Импорт элемента – направленное отношение между импортирующим пространством имен и отдельным элементом пакета, которое позволяет ссылаться на этот элемент с использованием невалифицированного имени.

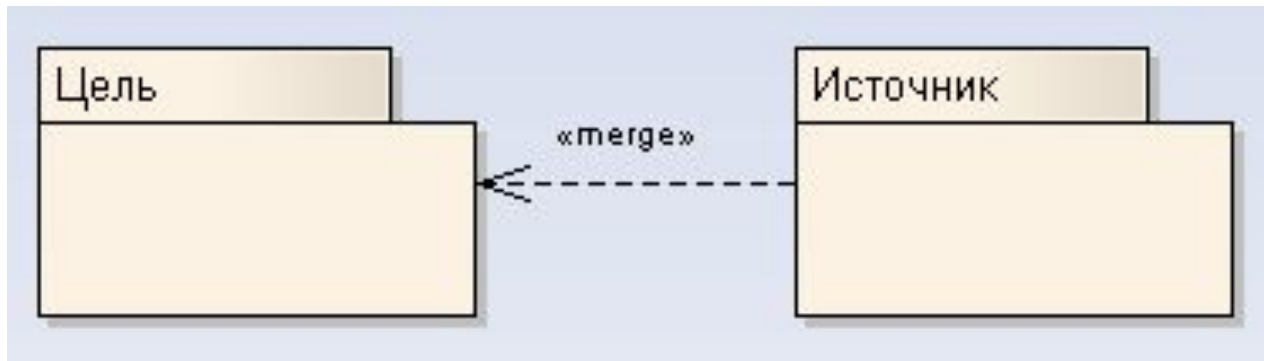


Импорт элемента

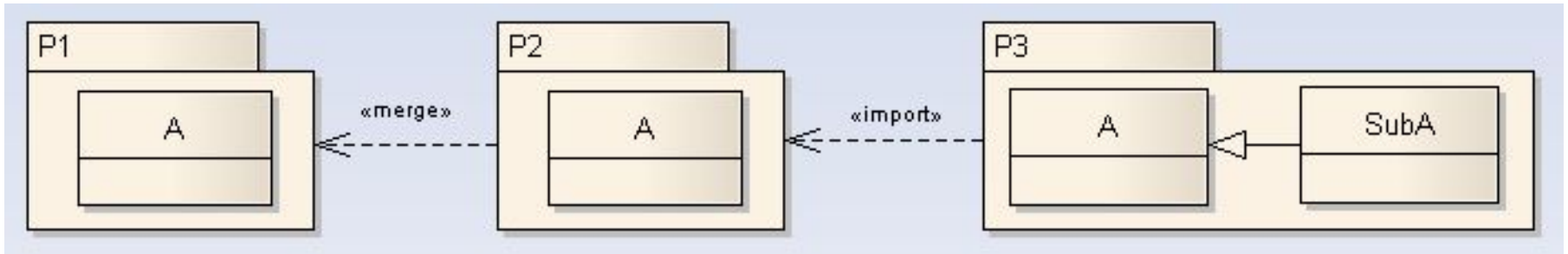


Слияние пакетов

Слияние пакетов – направленное отношение между двумя пакетами, один из которых расширяет свое содержание посредством добавления содержимого другого пакета.



Слияние пакетов



Класс A в пакете P3 (P3::A) представляет результат слияния P1::A в P2::A, а не только лишь приращение P2::A.

Слияние пакетов

Сливаемый пакет – первый операнд слияния, т.е. пакет, который сливается в принимающий пакет и который является целью стрелки слияния на диаграммах.

Принимающий пакет – второй операнд слияния, т.е. пакет, который концептуально содержит результаты слияния и который является источником стрелки слияния на диаграммах.

Результирующий пакет – это пакет, который концептуально результаты слияния.

Сливаемый элемент – элемент модели, который находится в сливаемом пакете.

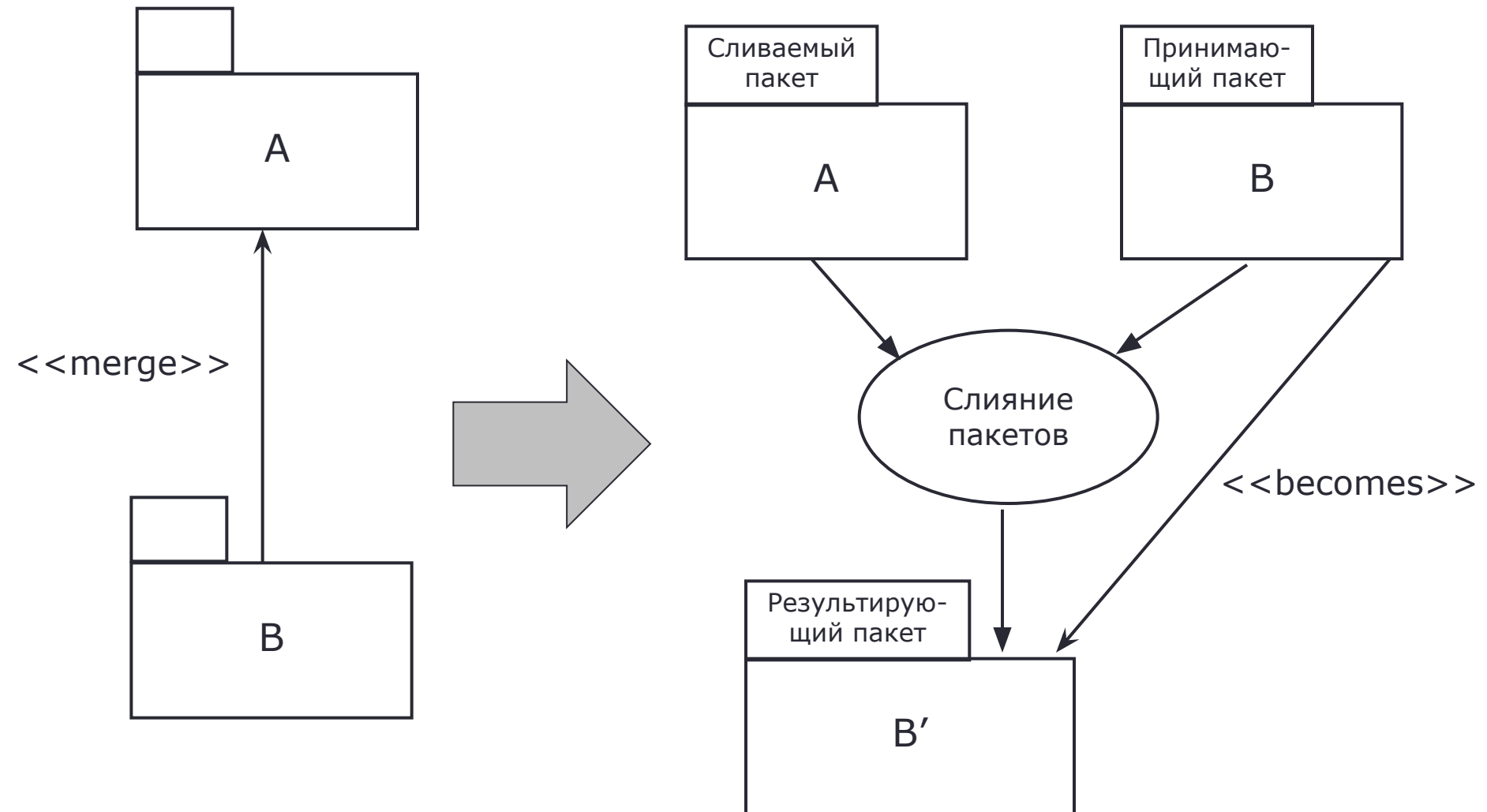
Принимающий элемент – элемент модели, который находится в принимающем пакете.

Результирующий элемент – элемент модели в результирующем пакете после выполнения слияния.

Тип элемента – любой допустимый тип элемента в языке UML 2, например, тип параметра или атрибута.

Метатип элемента – является типом MOF элемента модели, например, Классификатор, Ассоциация, Характеристика.

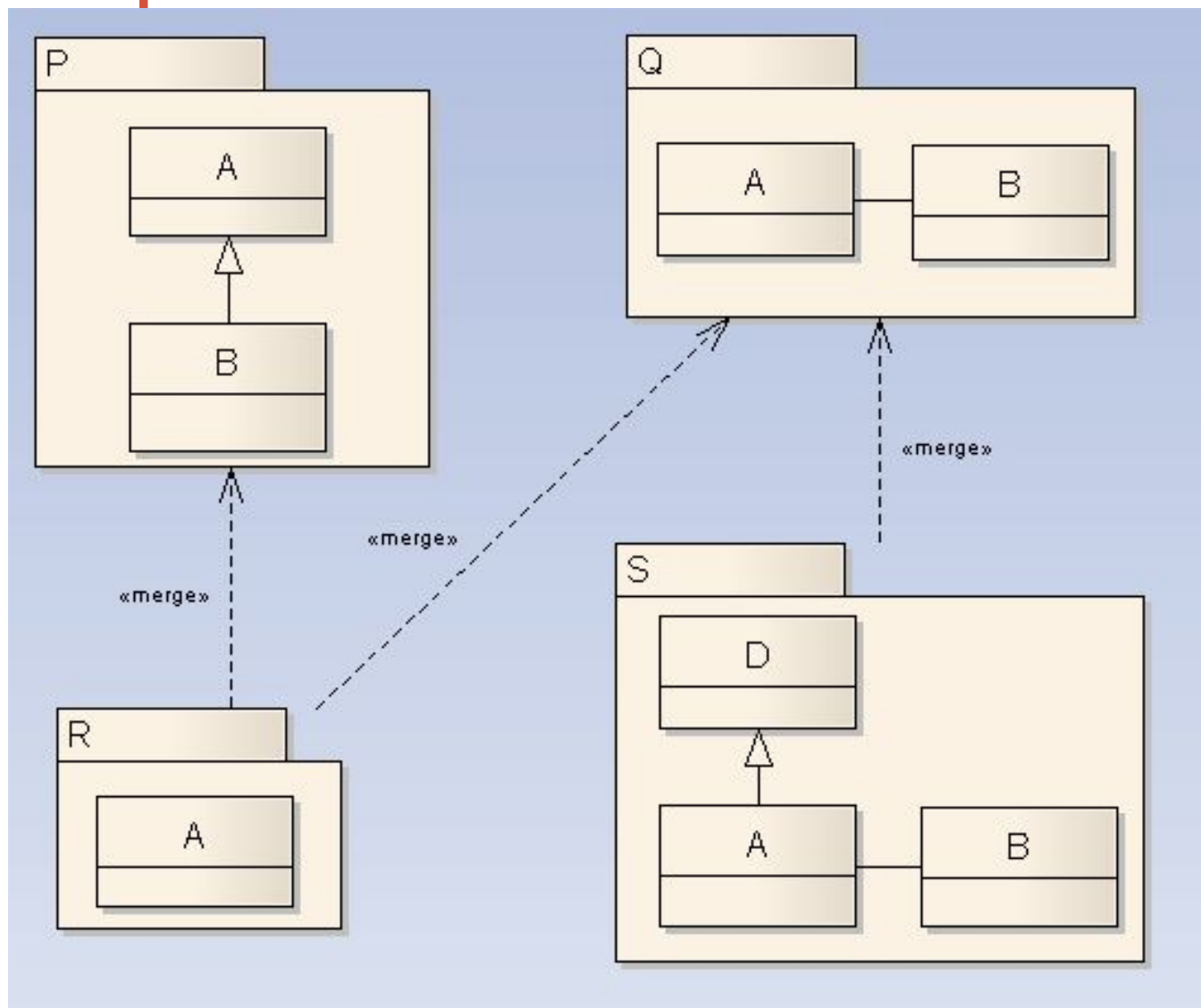
Слияние пакетов



Правила и ограничения слияния

- закрытые элементы пакета не сливаются;
- классы в принимающем пакете, имеющие такие же имена как и классы сливаемого пакета, становятся обобщением сливаемых классов;
- можно обращаться к исходным классам, используя квалифицированное имя с указанием оригинального пакета (имя пакета до слияния);
- классы, которые существуют только в принимающем или только в сливаемом пакете, остаются неизменными и добавляются в результирующий пакет;
- подпакеты внутри сливаемого пакета добавляются в результирующий пакет, если они не существуют в принимающем пакете. В противном случае, для этих пакетов выполняется операция слияния;
- импорты пакетов в сливаемом пакете становятся импортами пакетов в результирующем пакете (отношение обобщения для них не применяется);
- если импортируемые элементы конфликтуют с элементами в принимающем пакете, то элементы принимающего пакета имеют преимущество, а к импортируемым элементам необходимо обращаться, указывая квалифицированное имя.

Пример



Пример

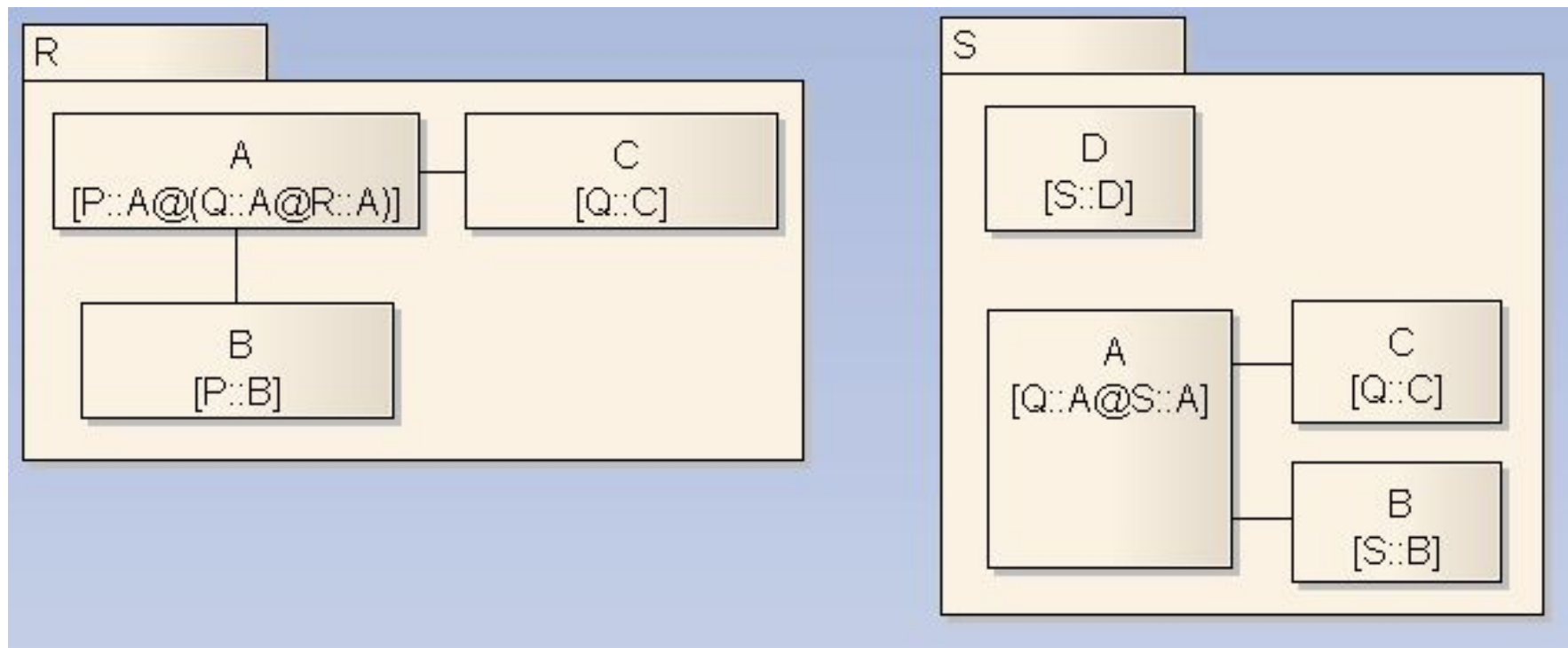


Диаграмма объектов

Диаграмма объектов предназначена для спецификации объектов и связей между ними для фиксированного момента времени.

Объект является отдельным экземпляром класса, который создается на этапе реализации модели или выполнения программы.

Имя объекта представляет собой строку текста записанную в следующем виде:

`<имя-объекта>::=[<собственное-имя-объекта>] | [:<имя-класса>]`

Изображение объектов

Варианты возможных записей имен объектов:

- о: С – для объекта специфицировано собственное имя объекта и имя класса;
- о – для объекта специфицировано только собственное имя объекта;
- :С – для объекта специфицировано только имя класса.

квадрат :
Прямоугольник

квадрат

:Прямоугольник

Спецификация экземпляра

Спецификация экземпляра является элементом модели, который представляет описание произвольной сущности типа классификатора.

Описание может включать:

- Классификацию этой сущности посредством одного или более классификаторов, сущность которых является экземпляром.
- Вид экземпляра, основанный на его классификаторе или классификаторах.
- Спецификацию значений структурных характеристик сущности.
- Необязательную спецификацию того, как вычислять, получать или создавать экземпляр.

Спецификация экземпляра

Имя спецификации экземпляра представляет собой строку текста, записанную в следующем виде:

<имя-спецификации-экземпляра>::=

[<собственное-имя-спецификации>] | [:<имя-класса>]



Слот

Слот предназначен для представления того, что сущность, моделируемая посредством спецификации экземпляра, имеет конкретное значение или значения для некоторой своей структурной характеристики.

Слот может быть изображен текстом в следующем формате:

<слот>::=<имя-характеристики>[:<тип-характеристики>]
<спецификация-значения>

‘=’

мойАдрес :Адрес

имяУлицы = "Садовая"
номерДома = 25
номерКвартиры = 15

:Прямоугольник

вершина = (1,10)
длина = 10
ширина = 20
цветГраницы = черный
цветЗаливки = белый