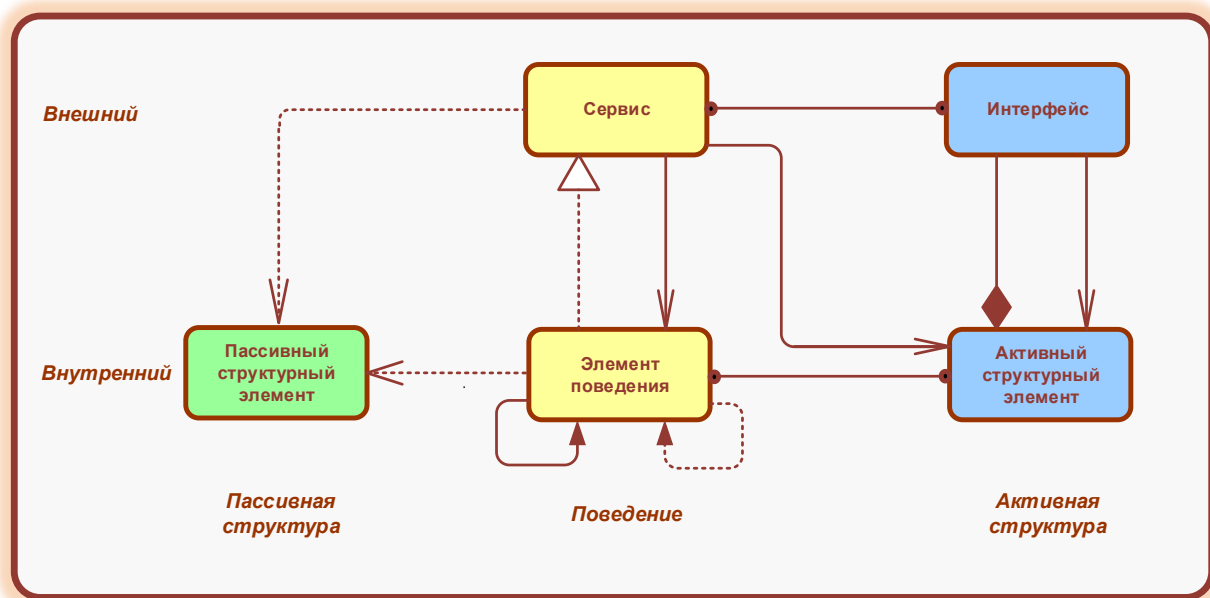


1 2 3 4 5

6 7 8 9 0

Лекция № 3 Отношения



Отношения – это различного рода соединения, связывающие элементы.

Линия, соединяющая две фигуры, устанавливает правила взаимоотношений между типами элементов, которые указываются внутри этих фигур (Рис. 3-1):

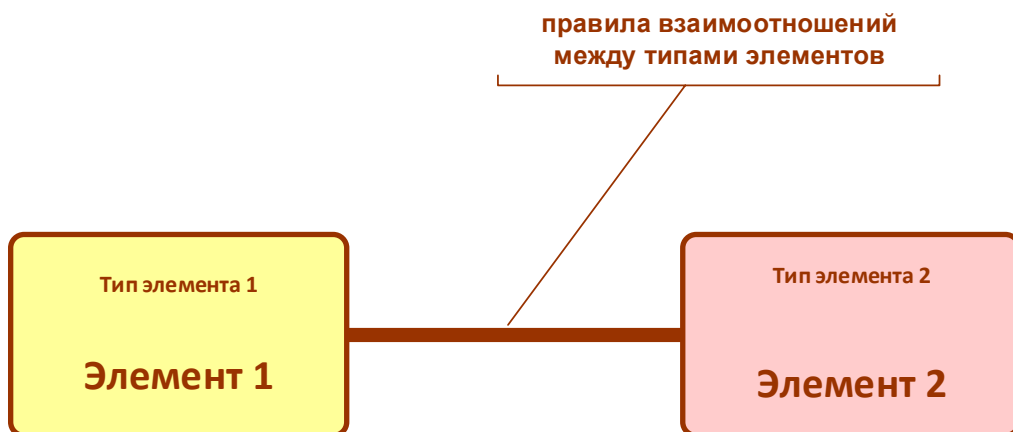


Рис. 3-1. Соединение определяет правила взаимоотношений между типами элементов

Отношения в языке разбиваются на три группы: структурные, динамические и другие.

1. Структурные отношения

Структурные отношения - это отношения, которые моделируют структурные зависимости элементов одного или разных типов.

Всего имеется 7 типов структурных отношений: композиция, объединение, назначение, реализация, использование, доступ и ассоциация.

1.1. Отношение «композиция» показывает, что объект состоит из одного или более других объектов (Рис. 3-2):



Рис. 3-2. Обозначение отношения «композиция»

Объект может быть частью только одной композиции (в отличие от отношения «объединение») (Рис. 3-3):

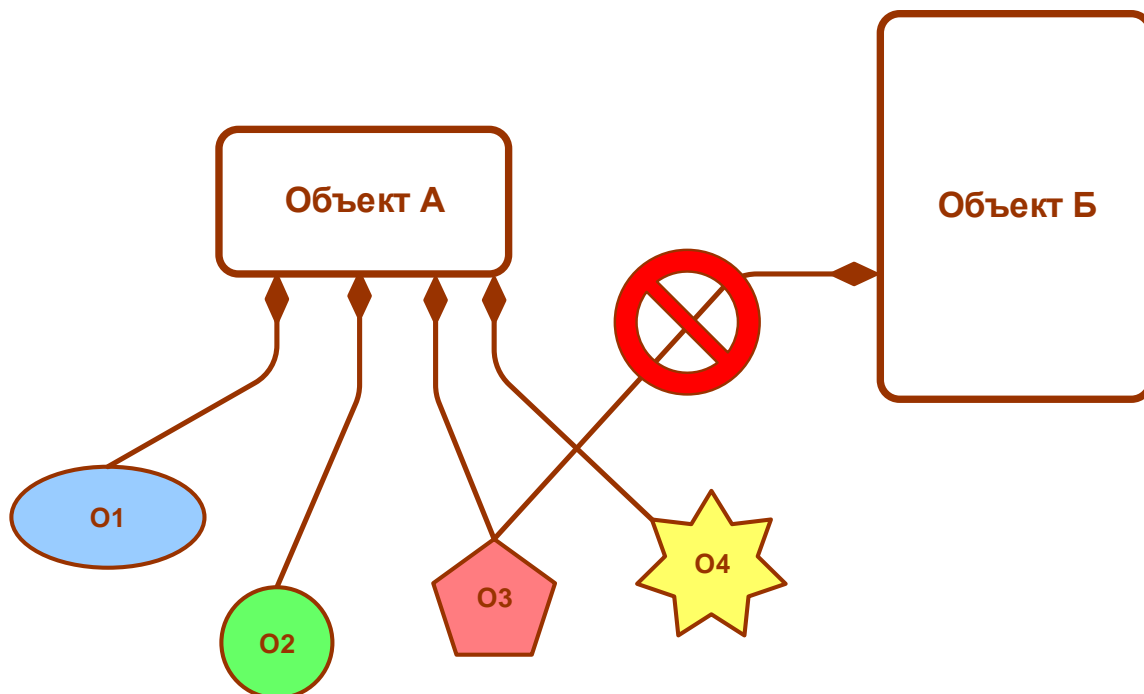


Рис. 3-3. Объект может быть частью только одной композиции

Элемент, соприкасающийся с черным ромбом, является элементом-родителем для элемента на другом конце соединения.

У отношения «композиция» существует альтернативный способ обозначения - вложение (гнездование) элементов (Рис. 3-4):

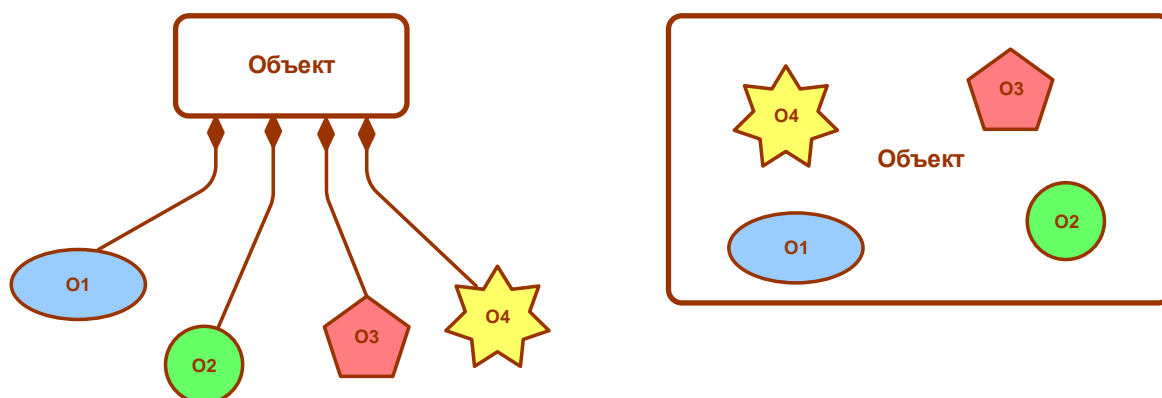


Рис. 3-4: Альтернативные способы обозначения отношения «композиция»

Такой способ более наглядный.

Но при его использовании остается неясным, какие именно отношения между целым и частями?

На рисунке 3-5 показан пример, в котором представлены два способа отображения того факта, что компонент приложений «Финансы» состоит из трех других компонентов приложений:

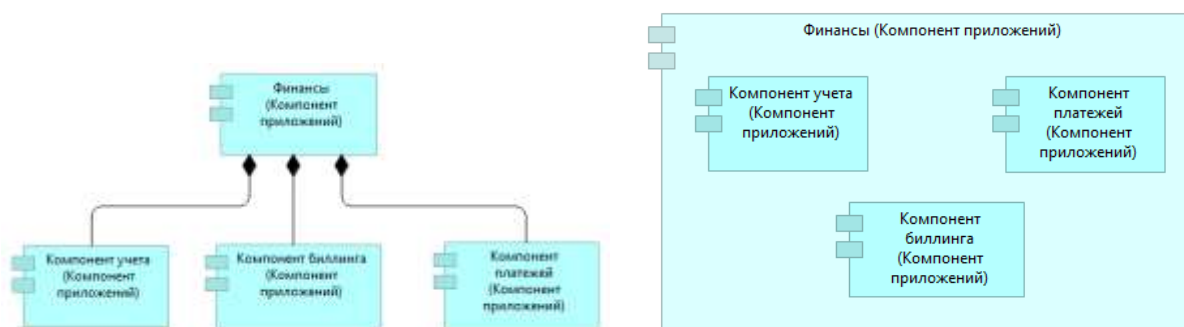


Рис. 3-5. Пример отношения «композиция»

Следует отметить, что в языке только у трех отношений имеется альтернативный способ отображения посредством вложения.

Это отношения «композиция», «объединение» и «назначение».

1.2. Отношение «объединение» показывает, что элемент образует группу из ряда других объектов (Рис. 3-6):



Рис. 3-6. Обозначение отношения «объединение»

Объект может быть частью более чем одного объединения (в отличие от отношения «композиция») (Рис. 3-7):

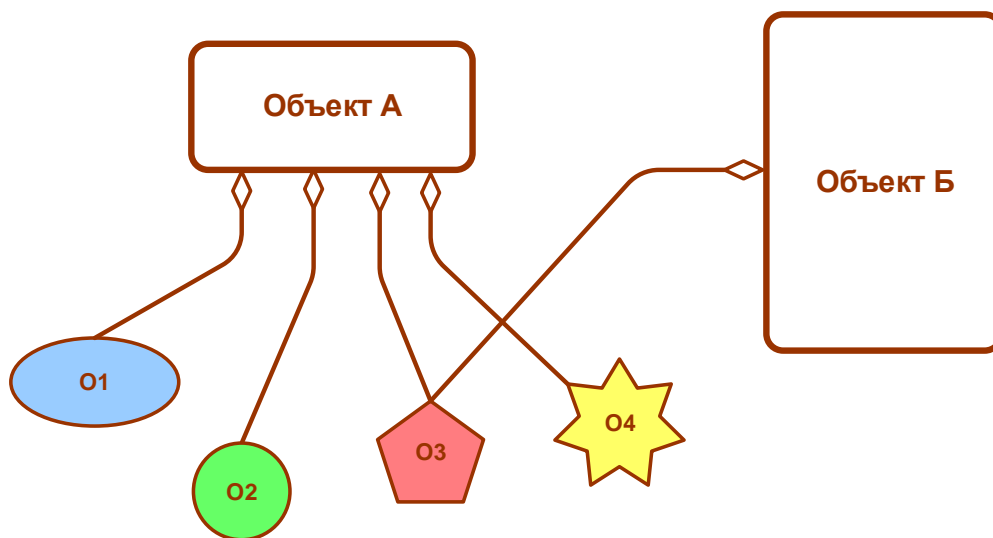


Рис. 3-7. Объект может быть частью более чем одного объединения

Отношение «объединение» всегда возможно между двумя экземплярами элементов одного типа.

Говорят, что отношение «композиция» – это сильное владение. Это отношение «целое – часть». В этом смысле отношение «объединение» является слабым владением.

У отношения «объединение» существует альтернативный способ обозначения - вложение элементов (Рис. 3-8):

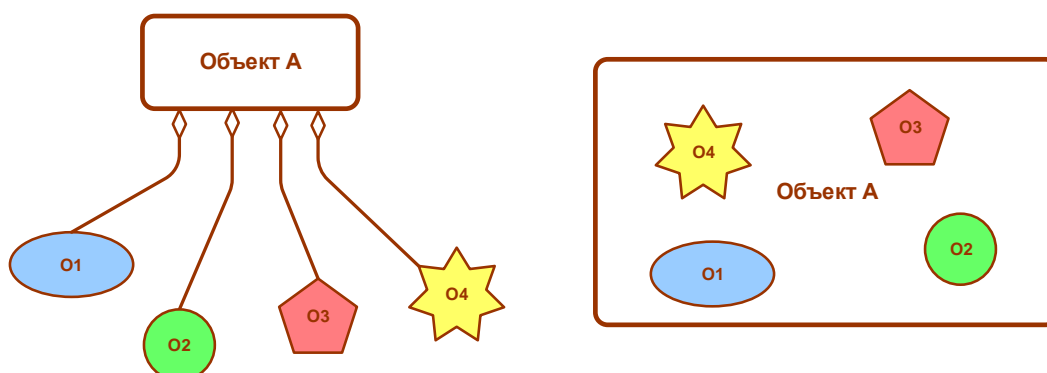


Рис. 3-8: Альтернативные способы обозначения отношения «объединение»

В примере показаны два способа отображения того факта, что продукт «Страхование Авто» объединяет Полис (элемент «Контракт») и два бизнес-сервиса (Рис. 3-9):



Рис. 3-9. Пример отношения «объединение»

1.3. Отношение «назначение» связывает активные элементы с единицами поведения, которые выполняются этими элементами, или бизнес-исполнителей с бизнес-ролями, которые выполняются исполнителями (Рис. 3-10):



Рис. 3-10. Обозначение отношения «назначение»

Отношение «назначение» может соединять:

- Бизнес-исполнителя с бизнес-ролью,
- Бизнес-роль с бизнес-процессом или функционалом,
- Совместную бизнес-деятельность с бизнес-взаимодействием,
- Бизнес-интерфейс с бизнес-сервисом,
- Компонент приложений с функционалом приложений,
- Совместную работу приложений с взаимодействием приложений,
- Интерфейс приложений с сервисом приложений,
- Площадку с бизнес-объектом, представлением или бизнес-исполнителем.

У отношения «назначение» также существует альтернативный способ обозначения - вложение элементов.

В примере показаны два способа выражения отношения «назначение» (Рис. 3-11):

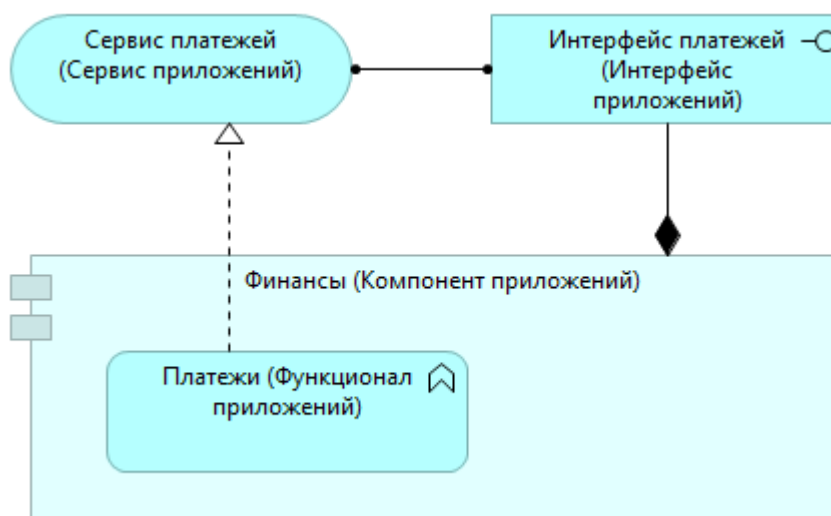


Рис. 3-11. Пример отношения «назначение»

Функционал приложений «Платежи» назначается компоненту приложений «Финансовое приложение» (способ вложения) и сервис приложений «Сервис платежей» назначается интерфейсу приложений «Интерфейс платежей» (явное связывание).

1.4. Отношение «реализация» связывает логическую сущность с более конкретной сущностью, которая ее реализует (Рис. 3-12):



Рис. 3-12. Обозначение отношения «реализация»

Элемент, соприкасающийся со стрелкой, - это элемент, создаваемый элементом на другом конце соединения.

Отношение «реализация» показывает, как логические (более абстрактные) сущности, такие, как сервисы, реализуются средствами более конкретных (реальных) сущностей.

Например, процесс/функционал реализует сервис, объект данных реализует бизнес-объект или артефакт реализует компонент приложений.

Пример показывает два способа использования отношения «реализация» (Рис. 3-13):

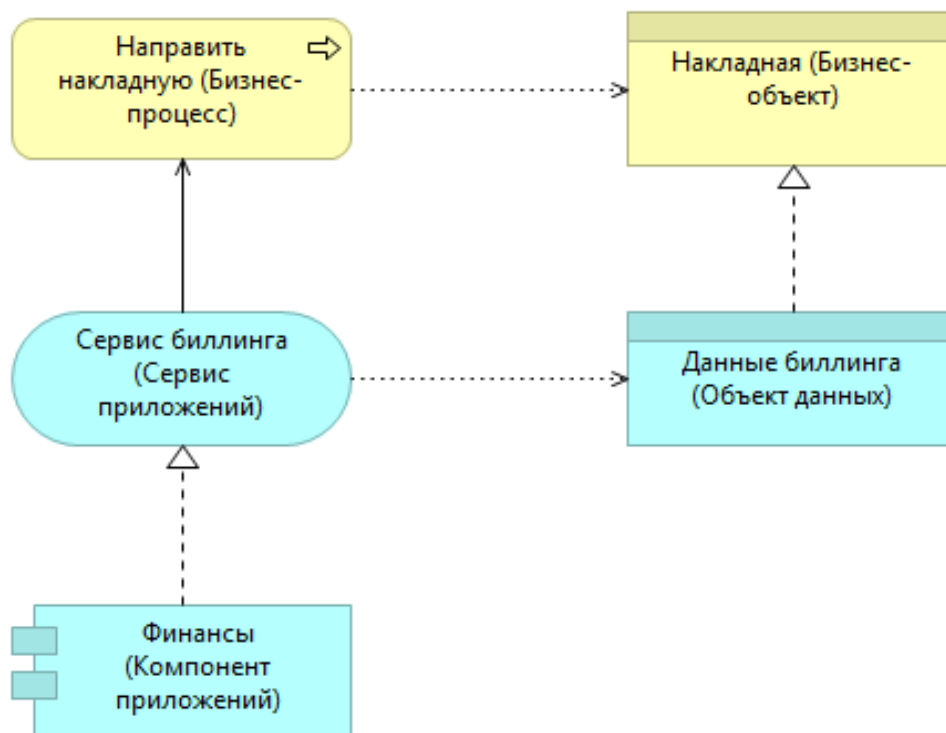


Рис. 3-13. Пример отношения «реализация»

Компонент приложений «Финансы» реализует сервис приложений «Сервис биллинга», а объект данных «Данные биллинга» реализует бизнес-объект «Накладная».

1.5. Отношение «использование» моделирует использование сервисов процессами, функционалами или взаимодействиями, а также доступ к интерфейсам ролями, компонентами или совместными деятельностью/работами (Рис. 3-14):



Рис. 3-14. Обозначение отношения «использование»

Элемент, соприкасающийся со стрелкой, использует элемент на другом конце соединения.

Отношение «использование» применяется как между структурными элементами, так и между элементами поведения.

Пример на рисунке 3-15 иллюстрирует применение отношения «использование»: интерфейс приложений (в данном случае интерфейс CRM-системы) используется служащим фронт-офиса, в то время как, сервис приложений «Сервис изменения информации по клиенту» используется бизнес-процессом «Изменить адрес»:

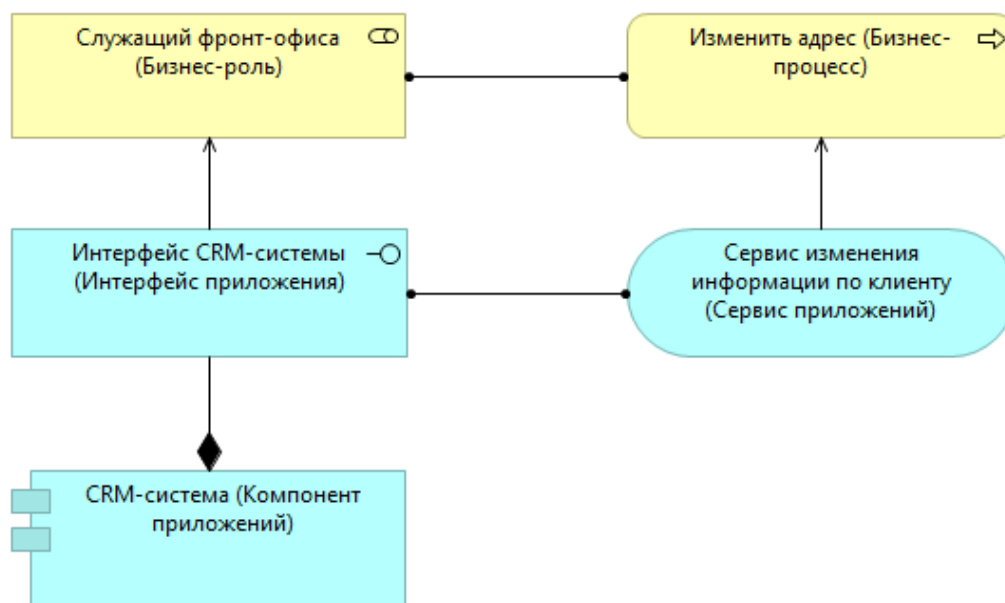


Рис. 3-15. Пример отношения «реализация»

1.6. Отношение «доступ» моделирует доступ элементов поведения к бизнес-объектам или объектам данных (Рис. 3-16):



Рис. 3-16. Обозначение отношения «доступ»

Отношение «доступ» показывает, что процесс, функционал, взаимодействие, сервис или событие «делают что-то» с бизнес-объектом или объектом данных. Например, создают новый объект, читают данные с объекта, пишут или модифицируют объект данных, удаляют объект.

Отношение «доступ» может быть также использовано для указания того, что объект просто ассоциирован с поведением.

Например, отношение «доступ» моделирует ситуацию, когда информация приходит с событием или информация становится доступной как часть сервиса. Если в обозначении отношения имеется стрелка, то она показывает направление потока информации.

В примере показано использование отношения «доступ» (Рис. 3-17):



Рис. 3-17. Пример отношения «доступ»

Процесс «Создать накладную» пишет/создает бизнес-объект «Накладная», а процесс «Направить накладную» читает бизнес-объект «Накладная».

1.7. Отношение «ассоциация» моделирует отношение между объектами, которое не охватывается другим, более характерным отношением (Рис. 3-18):



Рис. 3-18. Обозначение отношения «ассоциация»

Если необходимо связать два элемента, но точно неизвестно каким образом, то можно применить данное отношение.

В основном оно используется для моделирования отношений между бизнес-объектами или объектами данных, которые не моделируются отношениями «объединение», «композиция» или «специализация».

Также используется для связывания пассивных структурных элементов с другими элементами: бизнес-объект с образом объекта, образ объекта со смысловым значением, бизнес-сервис с ценностью.

В примере показано несколько вариантов использования отношения «ассоциация» (Рис. 3-19):

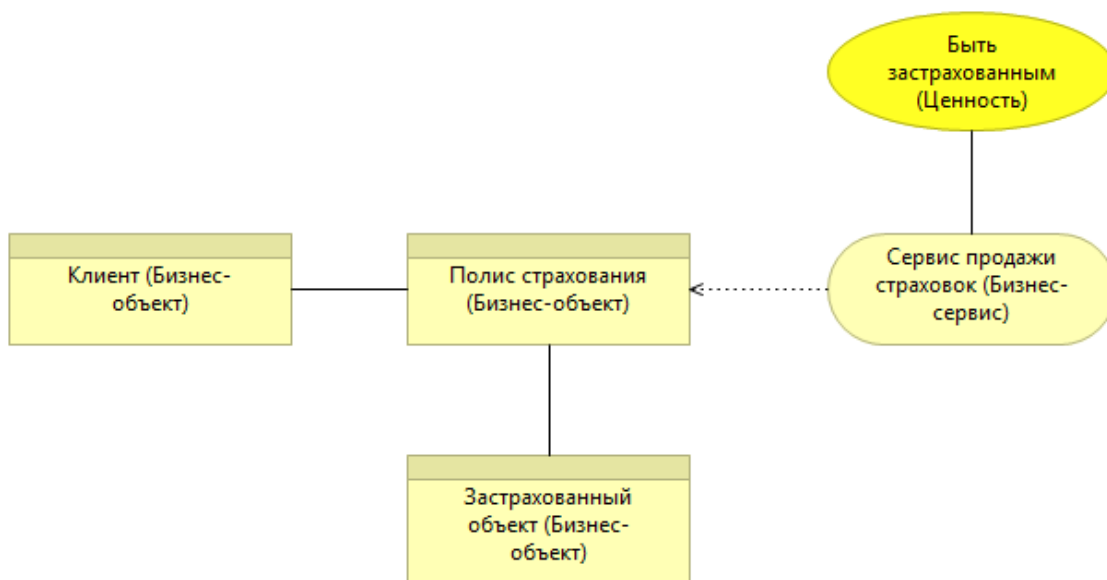


Рис. 3-19. Пример отношения «ассоциация»

2. Динамические отношения

Динамические отношения - это отношения, которые используют для моделирования зависимостей между элементами поведения.

Всего имеется два типа динамических отношений: запуск и передача.

2.1. Отношение «запуск» описывает временные или причинные отношения между процессами, функционалами, взаимодействиями и событиями (Рис. 3-20):



Рис. 3-20. Обозначение отношения «запуск»

Отношение «запуск» используется для моделирования причинных отношений между элементами поведения в процессе.

В примере показано, что отношения запуска используются для моделирования причинных зависимостей между подпроцессами и/или событиями (Рис. 3-21):

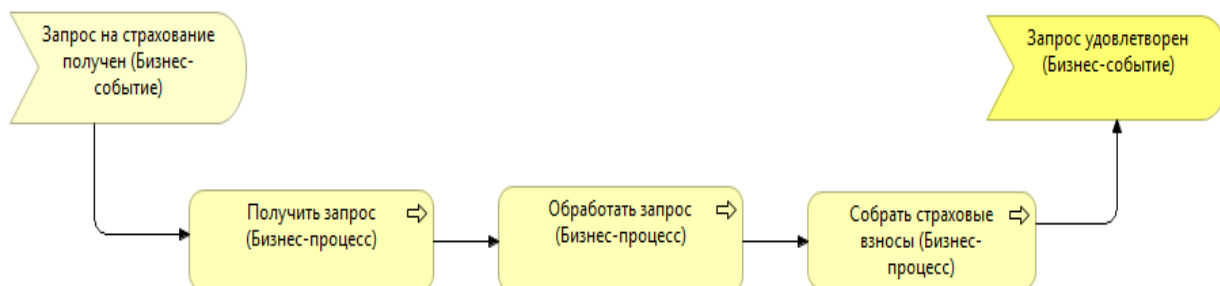


Рис. 3-21. Пример отношения «запуск»

2.2. Отношение «передача» описывает обмен или передачу, например, информации или ценности между процессами, функционалами, взаимодействиями и событиями (Рис. 3-22):



Рис. 3-22. Обозначение отношения «передача»

Отношение «передача» используется для моделирования передачи (потока), например, информации между элементами поведения в процессе.

Отношение «передача» не подразумевает причинной или временной зависимости.

В примере бизнес-функционал «Оценка претензий» направляет решения о претензиях бизнес-функционалу «Урегулирование претензий» (Рис. 3-23):

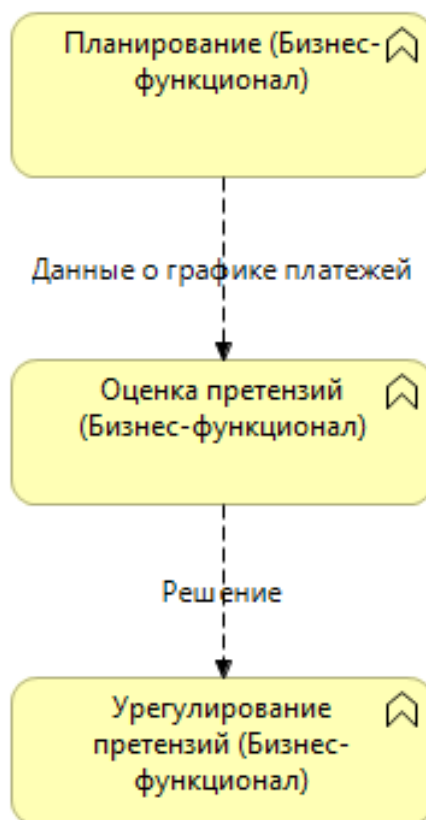


Рис. 3-23. Пример отношения «передача»

Для того, чтобы принять решение о порядке урегулирования претензии, бизнес-функционал «Оценка претензий» использует данные о платежах, полученные от бизнес-функционала «Планирование».

3. Другие отношения

Другие отношения - отношения, которые не входят в первые две группы.

Всего имеется три типа других отношений: группирование, связывание-разветвление, специализация.

3.1. Отношение «группирование» обозначает объекты одного или разных типов, которые объединяются на основе некоторого общего признака (Рис. 3-24):

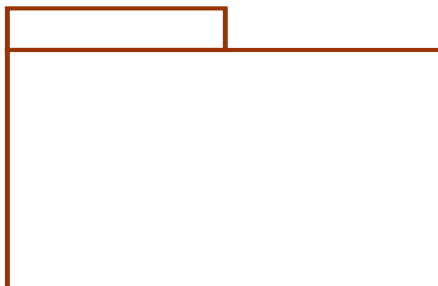


Рис. 3-24. Обозначение отношения «группирование»

В отличие от отношений «объединение» или «композиция» у отношения «группирование» нет «общего» объекта, в отношении которого сгруппированные объекты формируют часть.

Отношение «группирование» используется только для того, чтобы показать графически, что у элементов модели имеется что-то общее.

Это способ сделать представление более ясным для пользователя.

Внешне выглядит как элемент, но это отношение.

Элементы модели могут принадлежать различным (перекрывающимся) группам.

В примере показано, что отношение «группирование» используется для группирования объектов, которые относятся к одной информационной области – финансовому администрированию (Рис. 3-25):

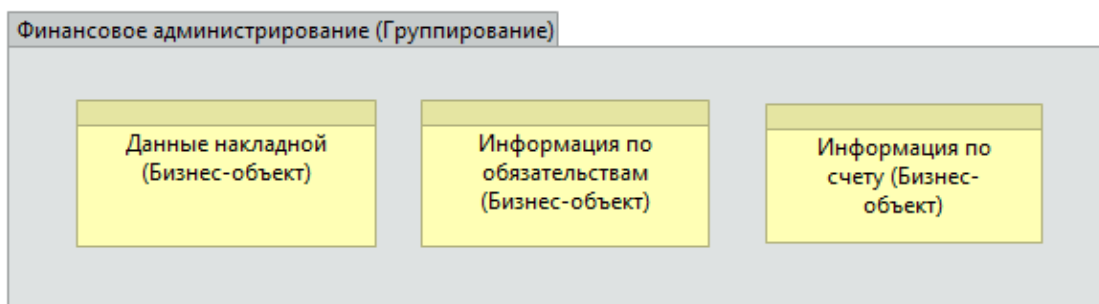


Рис. 3-25. Пример отношения «группирование»

3.2. Отношение «связывание-разветвление» используется для связывания динамических отношений одного типа (Рис. 3-26):



Рис. 3-26. Обозначение отношения «связывание-разветвление»

Отношение «связывание-разветвление» используется в ряде ситуаций для связывания динамических отношений одного типа (запуск или передача). Например, для обозначения разветвлений или соединений.

В примере отношение «связывание-разветвление» используется для обозначения выбора (Рис. 3-27):

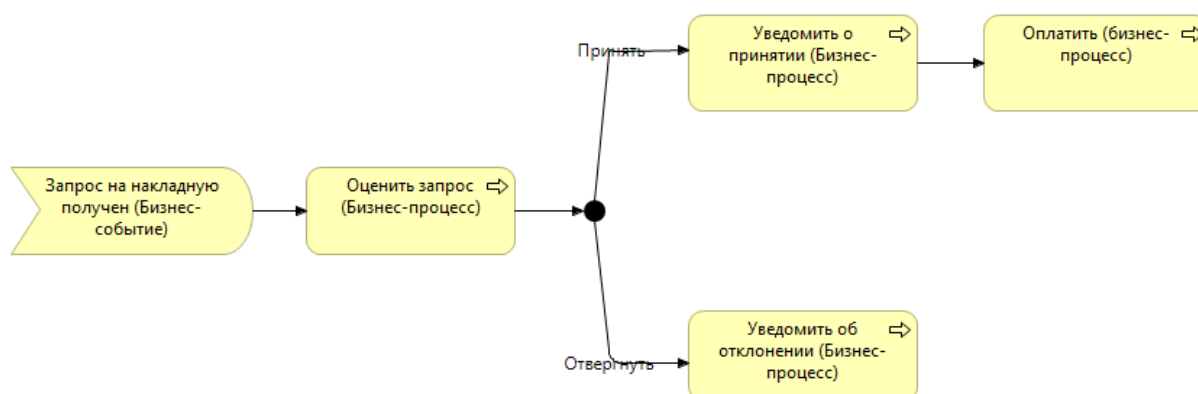


Рис. 3-27. Пример отношения «связывание-разветвление»

В случае принятия запроса выполняется процессы «Уведомить о принятии» и «Оплатить», а если запрос отвергается, то выполняется процесс «Уведомить об отклонении».

3.3. Отношение «специализация» показывает, что объект является специализацией другого объекта (Рис. 3-28):



Рис. 3-28. Обозначение отношения «специализация»

Данное отношение занимает особое положение.

Оно говорит, что элемент является разновидностью (конкретизацией) другого элемента.

Отношение «специализация» может соотносить любой экземпляр элемента с другим экземпляром элемента того же типа.

Отношение «специализация» всегда возможно между экземплярами одного типа.

Пример иллюстрирует использование отношения «специализация» для понятия «бизнес-процесс» (Рис. 3-29):



Рис. 3-29. Пример отношения «специализация»

Экземпляр процесса «Оформить страховку на поездку» и экземпляр процесса «Оформить страховку на багаж» являются специализацией более общего экземпляра процесса «Оформить страховку».

4. Производные отношения

4.1. Производные структурные отношения.

Структурные отношения формируют важную категорию отношений для описания общей связности.

На рисунке 3-30 структурные отношения приведены в порядке возрастания их «силы»:

- ассоциация – слабое структурное отношение;
- композиция – сильнейшее структурное отношение.



Рис. 3-30. Сила структурных отношений

Частью определения языка является правило, гласящее, что два отношения, которые соединяют промежуточный элемент, могут быть объединены и заменены слабым из этих отношений (Рис. 3-31):

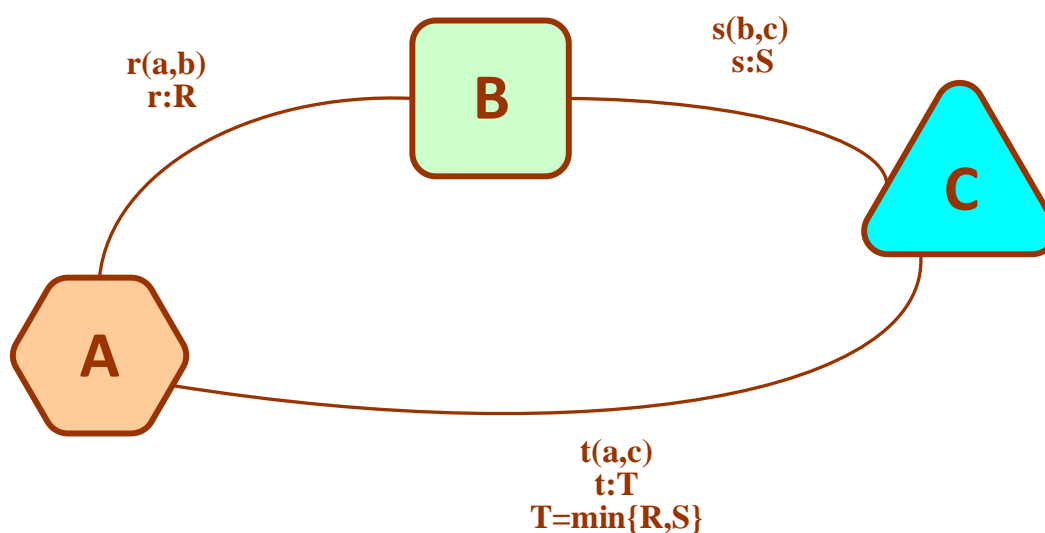


Рис. 3-31. Правило объединения и замены отношений

Если два структурных отношения $r:R$ и $s:S$ разрешены между элементами a, b и c таким образом, что $r(a, b)$ и $s(b, c)$, то структурное отношение $t:T$ $t(a, c)$ также разрешено и тип T равен слабейшему из R и S .

Транзитивное применение этого свойства позволяет заменить цепочку структурных отношений (с промежуточными элементами модели) слабейшим структурным отношением в цепочке.

Используя правило объединения и замены отношений, становится возможным определять производные (косвенные) отношения, которые существуют между элементами модели, что может быть полезно для проведения определенных видов анализа (Рис. 3-32):

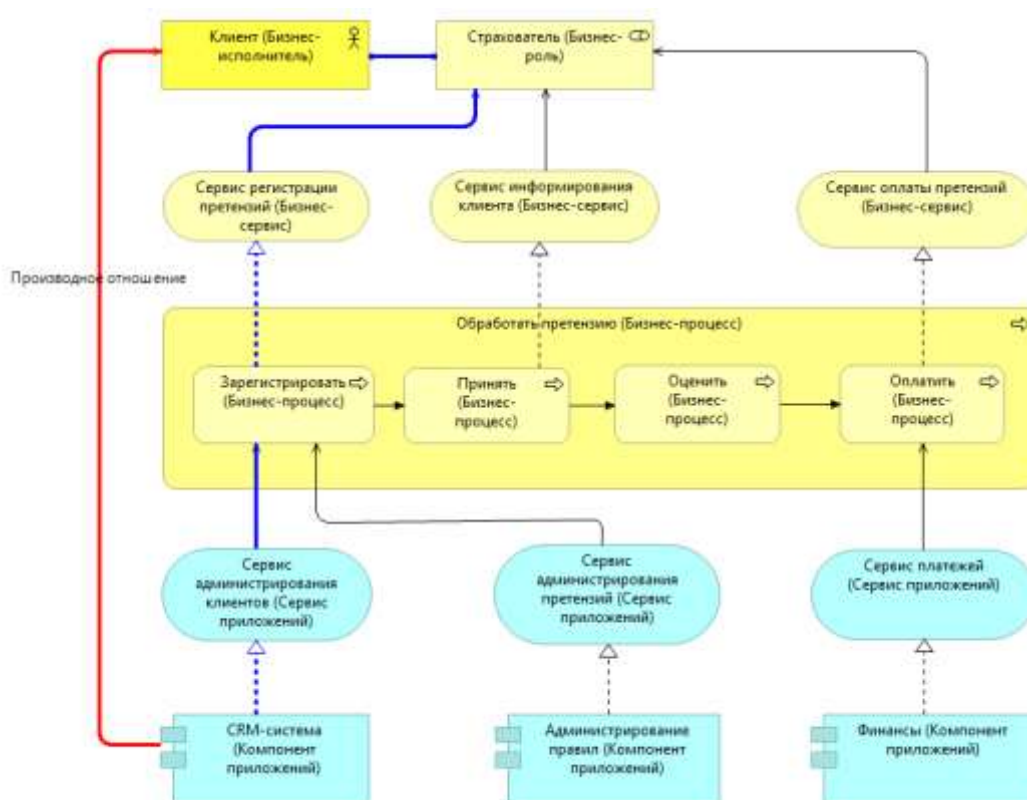


Рис. 3-32. Пример производных структурных отношений [4,5]

Например, каким будет воздействие на клиента при отказе работы CRM-системы? От элемента «CRM-система (Компонент приложений)» к элементу «Клиент (Бизнес-исполнитель)» может быть выведено производное отношение «использование» (красная стрелка слева) из цепочки отношений назначение – использование - реализация – использование – реализация, выделенных синим цветом.

Таким образом, использование клиентом сервиса «Сервис регистрации претензий» станет невозможным при отказе CRM-системы. А, например, использование сервиса «Сервис оплаты претензий» останется возможным, поскольку не существует производного структурного отношения между CRM-системой и сервисом «Сервис оплата претензий».

4.2. Производные динамические отношения.

Для двух динамических отношений «запуск» и «передача» применимы следующие правила (Рис. 3-33):

- начальная и/или конечная точка отношений «запуск» или «передача» между элементами поведения (например, процессами или функционалами) может быть перенесена к активным структурным элементам (например, бизнес-исполнителям или компонентам приложений), которые назначены (приписаны) к ним;
- начальная и/или конечная точка отношений «запуск» или «передача» между элементами поведения может быть перенесена к сервисам, которые они реализуют.

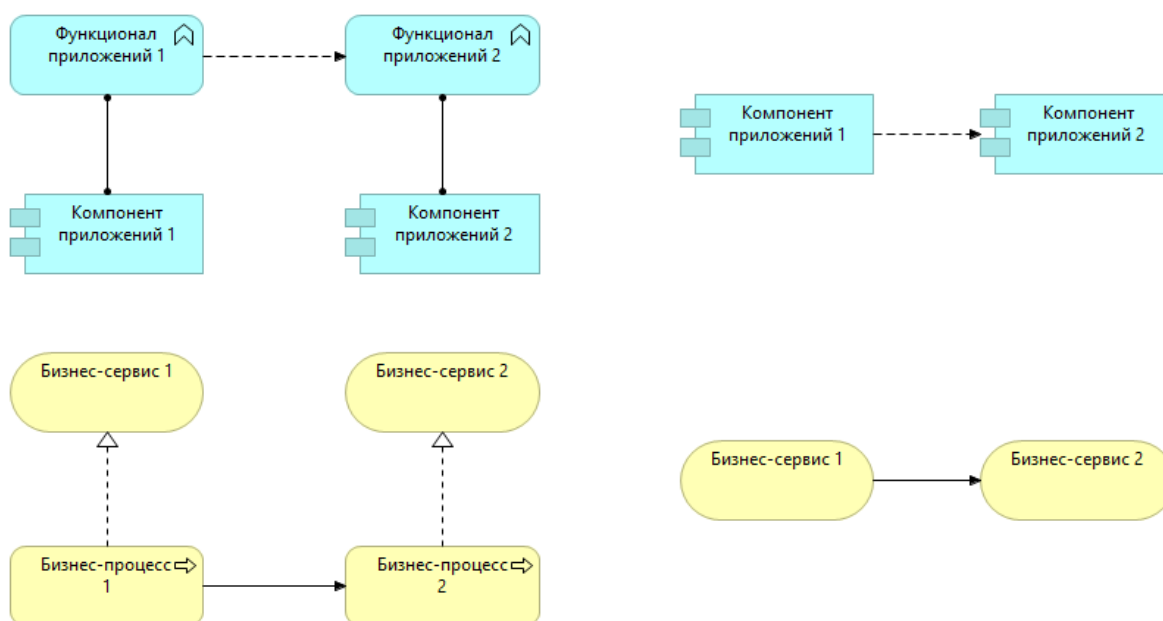


Рис. 3-33. Пример производных динамических отношений