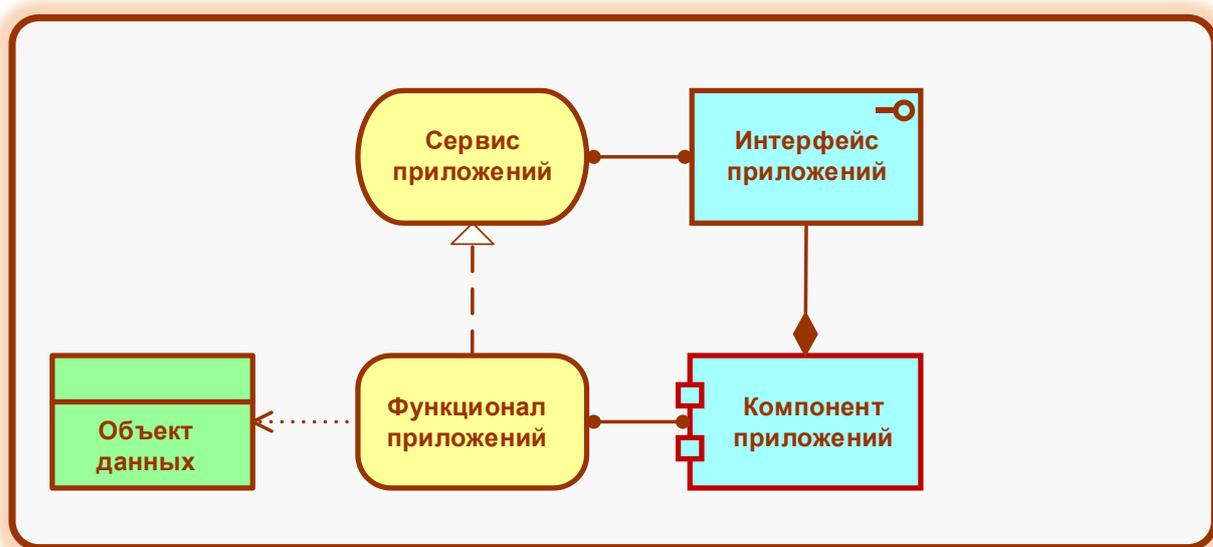




Лекция № 4

Элементы слоя приложений.

Моделирование архитектуры приложений и данных



1. Мета модель слоя

На рис. 4-1 представлена метамодель слоя приложений:

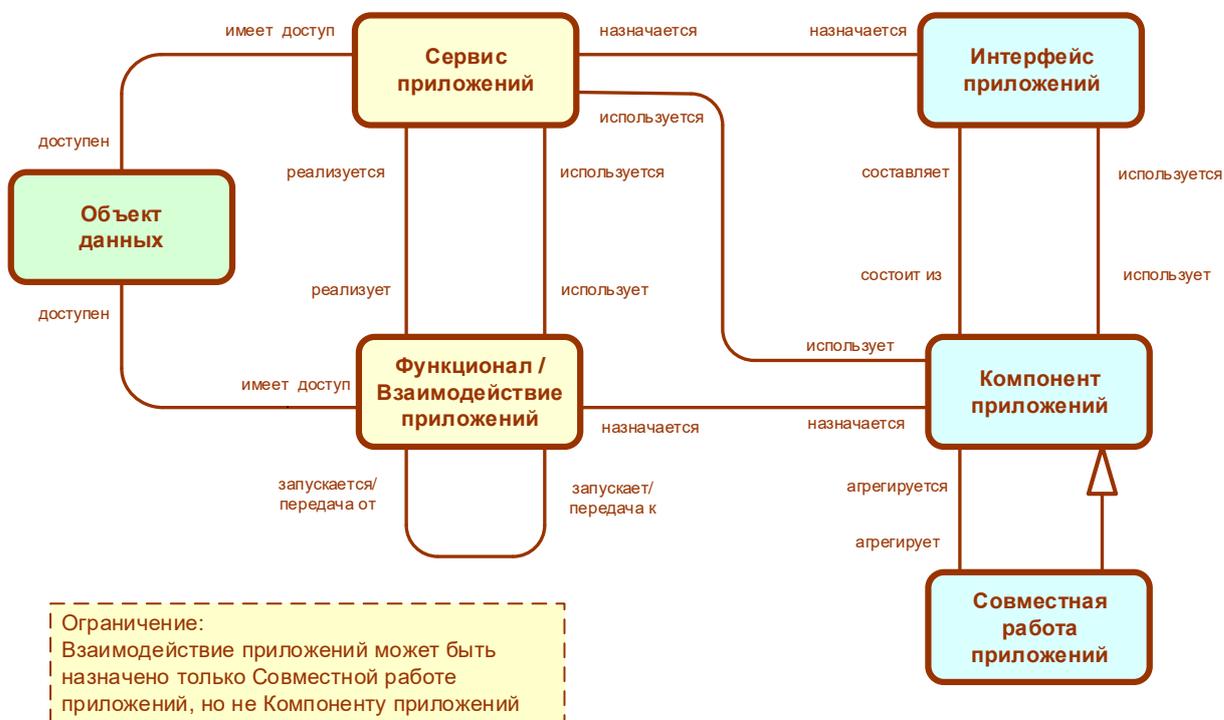


Рис. 4-1. Мета модель слоя приложений [4]

Ранее отмечалось, что все слои имеют сходную базовую структуру (назовем ее базовым шаблоном), соответствующую обобщенной метамодели, и эта базовая структура включает следующие 5 элементов: Исполнитель – Функционал – Интерфейс – Сервис – Объект (Рис. 4-2):

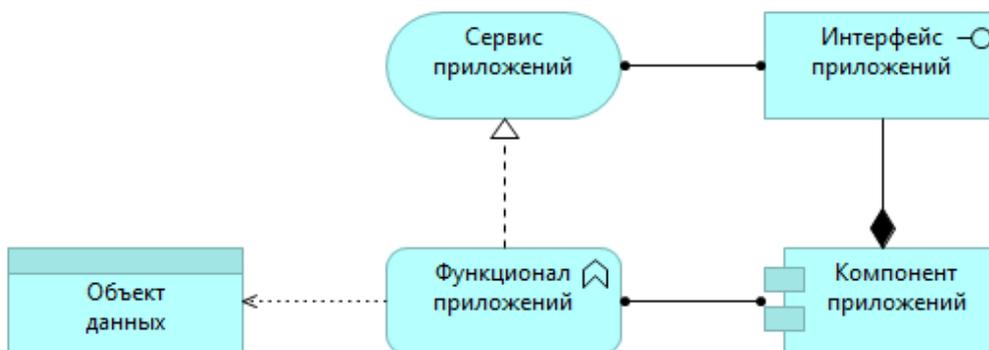


Рис. 4-2. Базовый шаблон слоя приложений

В слое приложений Исполнителем является компонент приложений, Функционалом - функционал приложений (внутреннее поведение компонента приложений), Интерфейсом является интерфейс приложений (способ, посредством которого компонент приложений предлагает себя бизнесу или другому приложению), Сервисом – сервис приложений (видимое поведение компонента приложений) и Объектом – объект данных (сущность, с или над которой действует функционал приложений).

Функционал приложений может создавать, читать, писать, изменять и удалять объект данных. Объект данных может и не существовать.

2. Активные структурные элементы

2.1. Компонент приложений определяется как модульная, развертываемая и заменяемая часть программного обеспечения системы, которая инкапсулирует свое поведение и данные и показывает их через набор интерфейсов (Рис. 4-3):

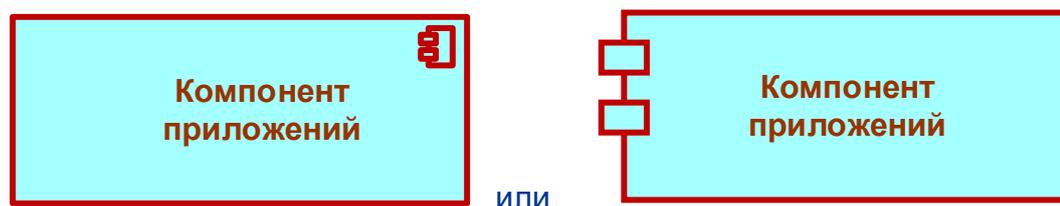


Рис. 4-3. Обозначение элемента «Компонент приложений»

Основным активным структурным элементом слоя приложений является компонент приложений.

Это понятие используется для моделирования любой структурной сущности в слое приложений: это не только компоненты программного обеспечения, которые могут быть частью одного или более приложений, но и целых, законченных приложений, подприложений или информационных систем.

Компонент приложений выполняет поведение через один или более функционалов приложений.

Он инкапсулирует свое содержание – его функциональность доступна только через набор интерфейсов приложений.

Компонент приложений может быть назначен одному или более функционалам приложений, бизнес-процессам или бизнес-функционалам.

У компонента приложений имеется один или более интерфейсов приложений, которые раскрывают его функциональность. Интерфейсы приложений других компонентов приложений могут быть использованы компонентом приложений.

Предпочтительное название элемента – существительное.

На рисунке 4-4 представлены типичные, наиболее часто используемые отношения между компонентом приложений и другими элементами:

- Компонент приложений реализует сервис приложений;
- У компонента приложений может быть один или более интерфейсов приложений;
- Объект данных создается, читается, модифицируется или разрушается компонентом приложений (или функционалом приложений);
- Компонент приложений может быть частью совместной работы приложений (через отношение «объединение»);
- Компонент приложений может состоять из нескольких компонентов приложений (через отношение «композиция»);
- Компонент приложений может быть назначен функционалу приложений;
- Компонент приложений может использовать инфраструктурный сервис;
- Между компонентами приложений могут иметь место потоки данных (через отношение «доступ»).

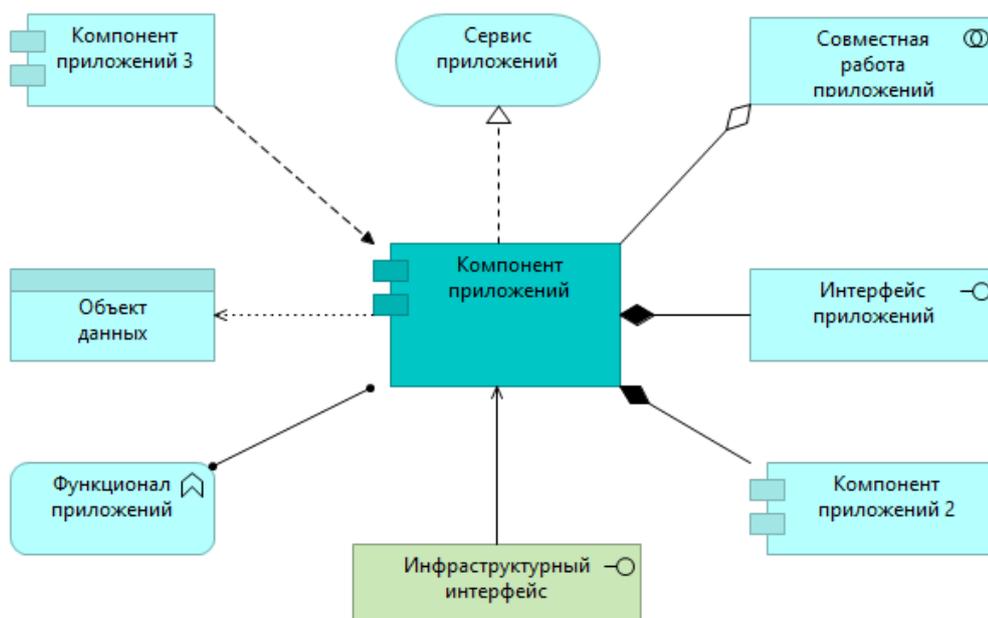


Рис. 4-4. Типичные отношения элемента «Компонент приложений» с другими элементами [14]

В примере показано, что приложение «Финансы» представляется компонентом приложений, состоящим из двух компонентов приложений – «Учет» и «Биллинг» (Рис. 4-5):

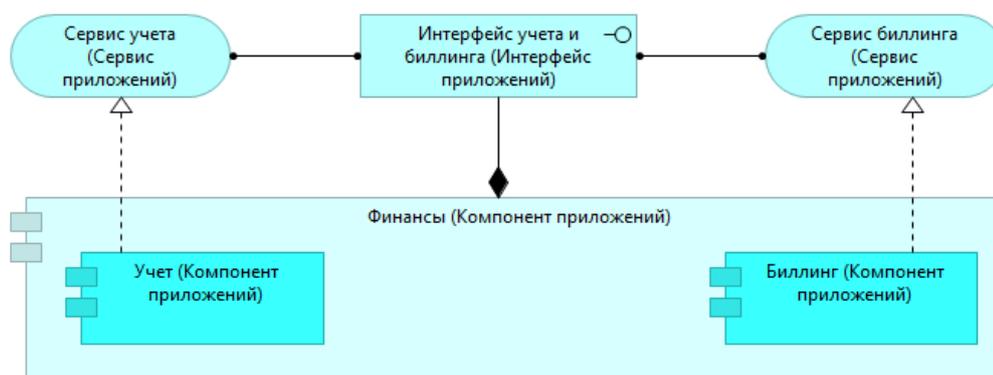


Рис. 4-5. Пример элемента «Компонент приложений»

Компонент приложений «Учет» реализует сервис приложений «Сервис учета», компонент приложений «Биллинг» реализует сервис приложений «Сервис биллинга». Эти сервисы доступны через общий интерфейс учета и биллинга, который является частью приложения «Финансы».

2.2. Совместная работа приложений определяется как объединение двух или более компонентов приложений, которые совместно работают для выполнения коллективного поведения (Рис. 4-6):



Рис. 4-6. Обозначение элемента «Совместная работа приложений»

Совместная работа приложений определяет - какие компоненты приложений взаимодействуют, чтобы выполнить некоторую работу, задачу.

Совместная работа приложений обычно используется для моделирования логического или временного сотрудничества компонентов приложений и не является отдельной сущностью.

Совместная работа приложений может быть назначена одному или более взаимодействиям компонент или бизнес-взаимодействиям, которые моделируют объединенное поведение.

Интерфейс приложений может быть использован совместной работой приложений и совместная работа приложений может включать интерфейсы приложений.

Предпочтительное название элемента – существительное.

В примере показаны два компонента приложений («Учет» и «Биллинг»), которые сотрудничают для администрирования транзакций. Данное сотрудничество моделируется элементом совместная работа приложений «Администрирование транзакций» (Рис. 4-7). Эта совместная работа приложений выполняется взаимодействием приложений «Администрировать транзакции».

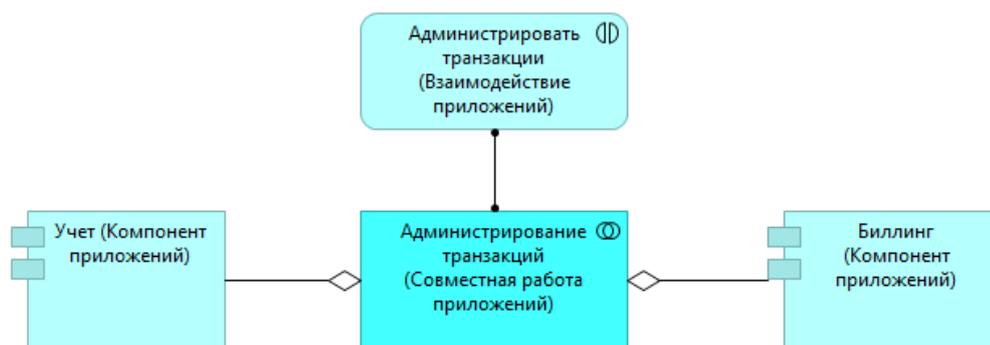


Рис. 4-7. Пример элемента «Совместная работа приложений»

2.3. Интерфейс приложений определяется как точка доступа, в которой сервис приложений становится доступным пользователю или другому компоненту приложений (Рис. 4-8):

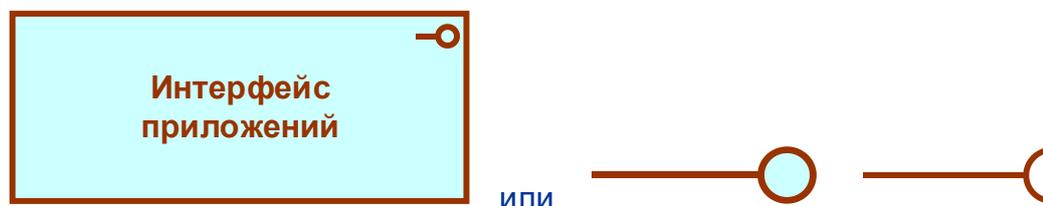


Рис. 4-8. Обозначение элемента «Интерфейс приложений»

Интерфейс приложений определяет, каким образом к функциональности компонента приложений может получить доступ другой компонент

(предоставляемый интерфейс —○) или какую функциональность запрашивает компонент от окружения (запрашиваемый интерфейс —○).

Интерфейс приложений раскрывает сервис приложений окружению.

Сервис приложений может быть представлен через различные интерфейсы.

Предпочтительное название элемента – существительное.

На рисунках 4-9 и 4-10 показаны два способа представления ситуации, в которой компонент приложений «Учет» предоставляет интерфейс приложений «Обмен данными (предоставляемый)», компонент приложений «Биллинг» запрашивает данный интерфейс через интерфейс приложений «Обмен данными (запрашиваемый)»:

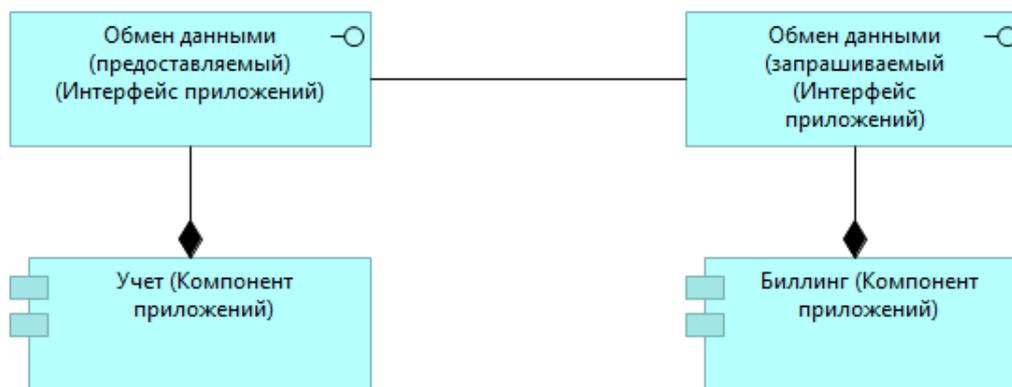


Рис. 4-9. Пример элемента «Интерфейс приложений»

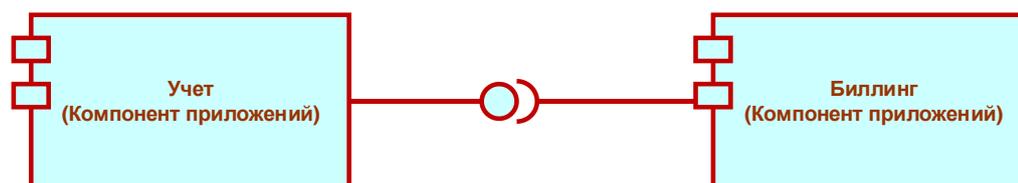


Рис. 4-10. Пример элемента «Интерфейс приложений» (альтернативное представление)

3. Элементы поведения

Различают внешнее поведение компонентов приложений в виде сервисов приложений и внутреннее поведение этих компонентов в виде функционалов приложений, которые реализуют эти сервисы.

3.1. Функционал приложений определяется как элемент поведения, группирующий автоматизированное поведение, которое может быть выполнено компонентом приложений (Рис. 4-11):



Рис. 4-11. Обозначение элемента «Функционал приложений»

Функционал приложений представляет (описывает) внутреннее поведение компонента приложений.

Если это поведение раскрывается внешне, то это осуществляется через один или более сервисов.

Функционал приложений может реализовать один или более сервисов.

Функционал приложений может использовать сервисы приложений других функционалов приложений и инфраструктурные сервисы.

Функционал приложений может иметь доступ к объекту данных.

Функционалу приложений может быть назначен компонент приложений.

Предпочтительное название элемента – отглагольное существительное.

В примере внутреннее поведение компонента приложений «Финансы» моделируется функционалом приложений «Финансовое администрирование», состоящее из двух подфункционалов (Рис.4-12). Эти подфункционалы реализуют сервисы приложений, доступные пользователям приложения:

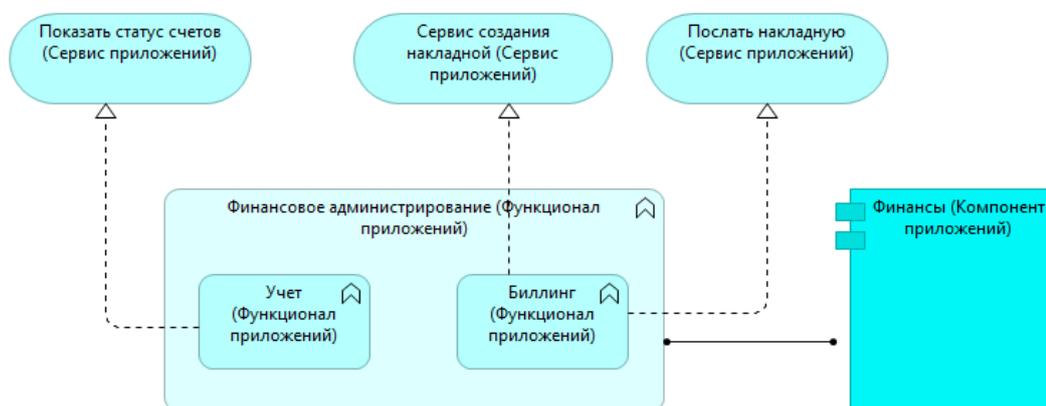


Рис. 4-12. Пример элемента «Функционал приложений»

3.2. **Взаимодействие приложений** определяется как элемент поведения, описывающей поведение совместной работы приложений (Рис. 4-13):



Рис. 4-13. Обозначение элемента «Взаимодействие приложений»

Взаимодействие приложений описывает коллективное поведение, которое выполняется компонентами, участвующими в совместной работе приложений. Совместная работа приложений может быть назначена взаимодействию приложений.

Взаимодействие приложений может реализовать сервис приложений.

Сервисы приложений и инфраструктурные сервисы могут быть использованы взаимодействием приложений.

Взаимодействие приложений может иметь доступ к объектам данных.

Предпочтительное название элемента – глагол.

В примере показано, что два компонента приложений («Учет» и «Биллинг») финансовой подсистемы совместно выполняются (работают) для администрирования транзакций (совместная работа приложений «Администрирование транзакций») (Рис. 4-14). Это поведение моделируется посредством элемента взаимодействие приложений «Администрировать транзакции», которое назначено элементу совместная работа приложений «Администрирование транзакций»:

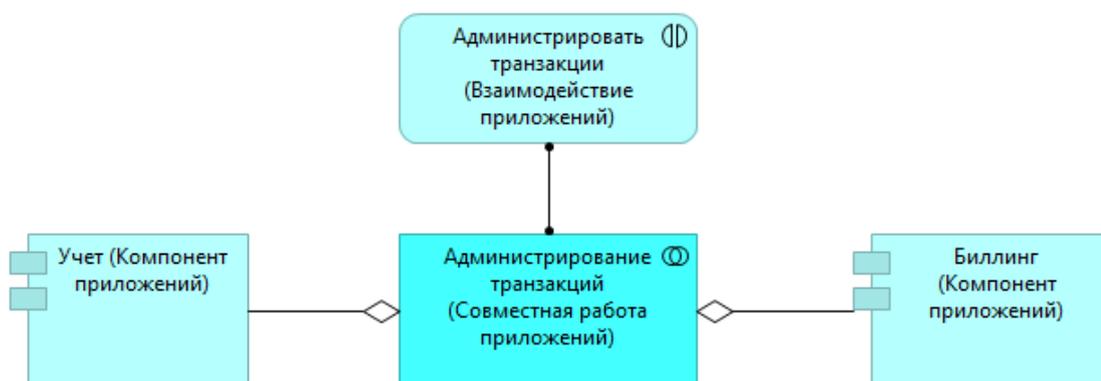


Рис. 4-14. Пример элемента «Взаимодействие приложений»

3.3. Сервис приложений определяется как сервис, демонстрирующий (раскрывающий) автоматизированное поведение (Рис. 4-15):

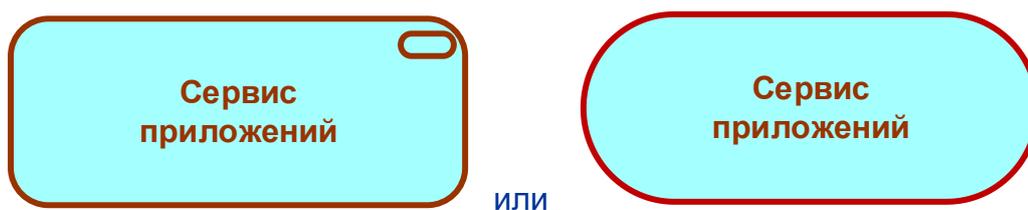


Рис. 4-15. Обозначение элемента «Сервис приложений»

Сервис приложений раскрывает функциональность компонентов приложений их окружению.

Сервис приложений является значимым с точки зрения окружения; он предоставляет единицу функциональности, полезную пользователям. Эта функциональность доступна через один или более интерфейсов приложений.

Сервис приложений реализуется функционалом приложений или взаимодействием приложений.

Сервис приложений может быть использован функционалами приложений или бизнес-процессами, бизнес-функционалами, бизнес-взаимодействиями.

Сервису приложений может быть назначен интерфейс приложений.

Сервис приложений может иметь доступ к объектам данных.

Предпочтительное название элемента – отглагольное существительное либо в названии может явно использоваться слово «сервис».

На рисунке 4-16 представлены типичные, наиболее часто используемые отношения между сервисом приложений и другими элементами:

- Сервис приложений реализуется функционалом приложений или компонентом приложений;
- Сервис приложений используется компонентом приложений;
- Сервис приложений может использоваться другим сервисом приложений;
- Сервис приложений имеет доступ к объекту данных (сервис создает, читает, модифицирует или ликвидирует объект данных);
- Сервис приложений может состоять из других сервисов приложений и может использовать другие сервисы приложений.



Рис. 4-16. Типичные отношения элемента «Сервис приложений» с другими элементами [14]

В примере на рисунке 4-17 показано, что сервис приложений «Сервис обработки транзакций» реализуется функционалом приложений «Учет» и доступен другим компонентам через интерфейс приложений «API обработки транзакций»:

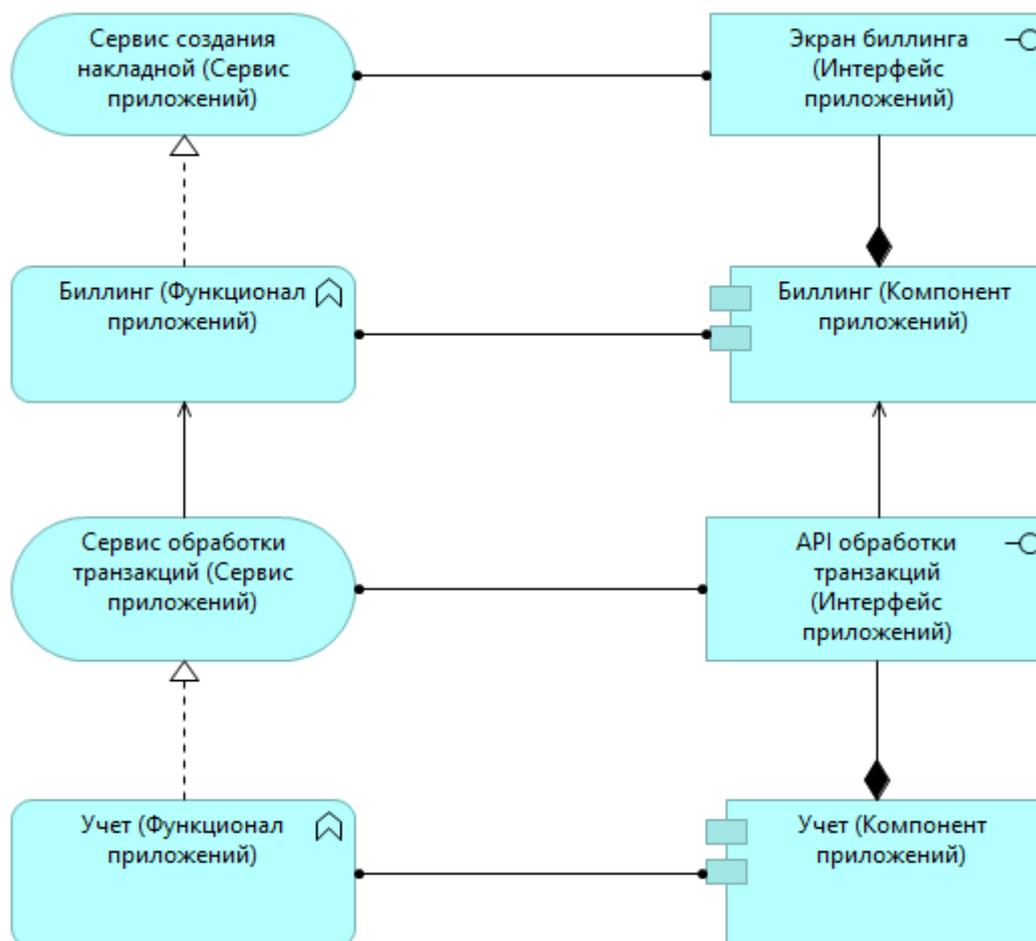


Рис. 4-17. Пример элемента «Сервис приложений»

Функционал приложений «Учет» выполняется компонентом приложений «Учет». Сервис приложений «Сервис обработки транзакций» используется функционалом приложений «Биллинг», который выполняется компонентом приложений «Биллинг». Функционал приложений «Биллинг» предлагает сервис приложений «Сервис создания накладной», который может быть использован для поддержки бизнес-процессов. Данный сервис доступен через интерфейс приложений «Экран биллинга».

4. Пассивные структурные элементы

4.1. Объект данных определяется как пассивный элемент, подходящий для автоматизированной обработки (Рис. 4-18):



Рис. 4-18. Обозначение элемента «Объект данных»

Объект данных – это абстрактное понятие, создаваемое или используемое функционалом приложений.

Объект данных - это самостоятельная часть информации с понятным значением не только для уровня приложений, но и для бизнеса.

Типичные примеры объектов данных – записи различного рода о клиентах, заказах, документах и т.п.

Функционал приложений производит операции с объектом данных.

Объект данных может реализовать бизнес-объект.

Объект данных может быть реализован артефактом.

Доступ к объекту данных могут иметь функционал приложений, взаимодействие приложений и сервис приложений.

У объекта данных могут быть отношения ассоциации, специализации, объединения и композиции с другими объектами данных.

Предпочтительное название элемента – существительное.

На рисунке 4-19 представлены типичные, наиболее часто используемые отношения между объектом данных и другими элементами:

- Объект данных создается, читается, модифицируется или ликвидируется компонентом приложений, сервисом приложений или функционалом приложений;
- У объекта данных могут быть специализации;
- Объект данных реализуется артефактом;
- Объект данных может входить в другие объекты данных (отношение «объединения»);
- Объект данных может состоять из других объектов данных (отношение «композиция»).

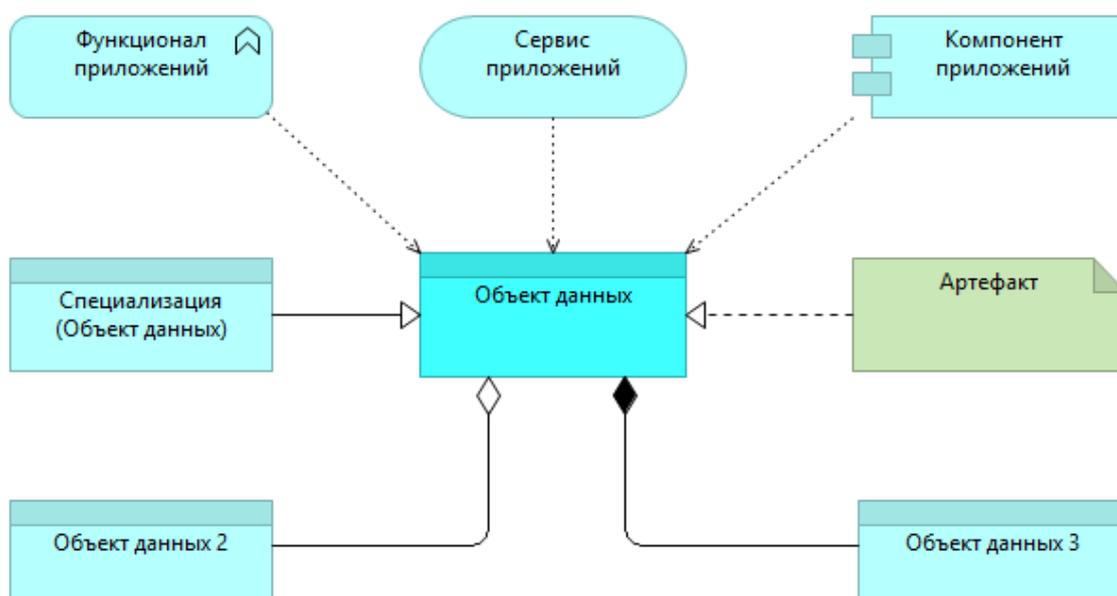


Рис. 4-19. Типичные отношения элемента «Объект данных» с другими элементами [14]

В примере показано, что два функционала приложений «Учет» и «Биллинг» взаимодействуют через сервис приложений «Обработка транзакций», который изменяет объект данных, содержащий данные по транзакциям (Рис. 4-20):

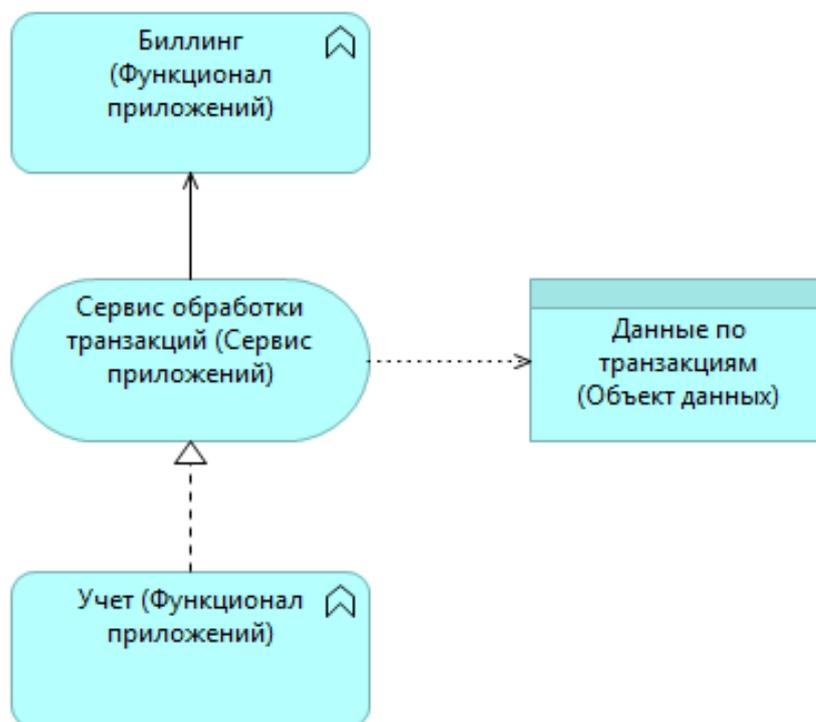


Рис. 4-20. Пример элемента «Объект данных»