

Подходы проектирования
архитектурных артефактов
с использованием принципов
Composable architecture

Дмитрий Клецких

О чем поговорим ?

ИТ-ландшафт и роль корпоративной архитектуры



«ИС-центричный» подход к описанию ИТ-ландшафта



Composable Architecture

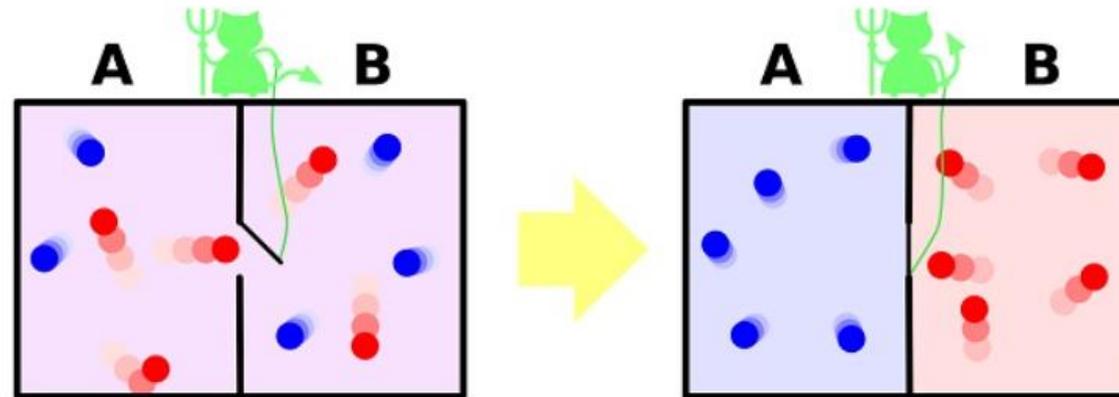


Представление ИТ-ландшафта с использованием принципов
Composable architecture



Энтропия и Демон Максвелла

- Энтропия - это функция беспорядка в системе
- Согласно второму началу термодинамики все в этом мире должно со временем разрушаться и приходить в беспорядок (в замкнутых системах энтропия либо остаётся неизменной, либо возрастает)
- Возможен ли переход замкнутой системы в состояние с меньшей энтропией?
 - Да, с помощью [Демона Максвелла](#), который может разделить горячие и холодные молекулы в сосуде и таким образом перевести систему в более упорядоченное состояние с меньшей энтропией



Энтропия и Демон Максвелла

В каждой шутке есть доля правды...



Корпоративные архитекторы - это
Демоны Максвелла для IT ландшафта

Архитектура —

это фундаментальный способ управления изменениями в организации, включающий принципы и стандарты, объекты управления, их отношения и связи

В эпоху глобальных мировых процессов трансформации
АРХИТЕКТУРА - это и есть СТРАТЕГИЯ

Видение развития ИТ-ландшафта

- Видение развития ИТ-ландшафта включает в себя реализацию бизнес и технологических трансформаций, необходимых для достижения стратегических целей банка, максимально прозрачно и предсказуемо
- Ключевой архитектурный артефакт - Strategic Architecture, учитывающий Business Strategy и Business Capabilities



Драйверы трансформации ИТ-ландшафта

1. Импортозамещение и технологическая независимость [изменения внешней среды]
2. Омниканальность [бизнес-стратегия]
3. Движение в сторону neobank [бизнес-стратегия]
4. Желание вернуть согласованность между бизнес-стратегией и ИТ-стратегией [управляемость, наблюдаемость, устойчивость, гибкость, скорость, стоимость]
5. Борьба с возросшей энтропией ИТ ландшафта [повышение надежности и снижение стоимости]



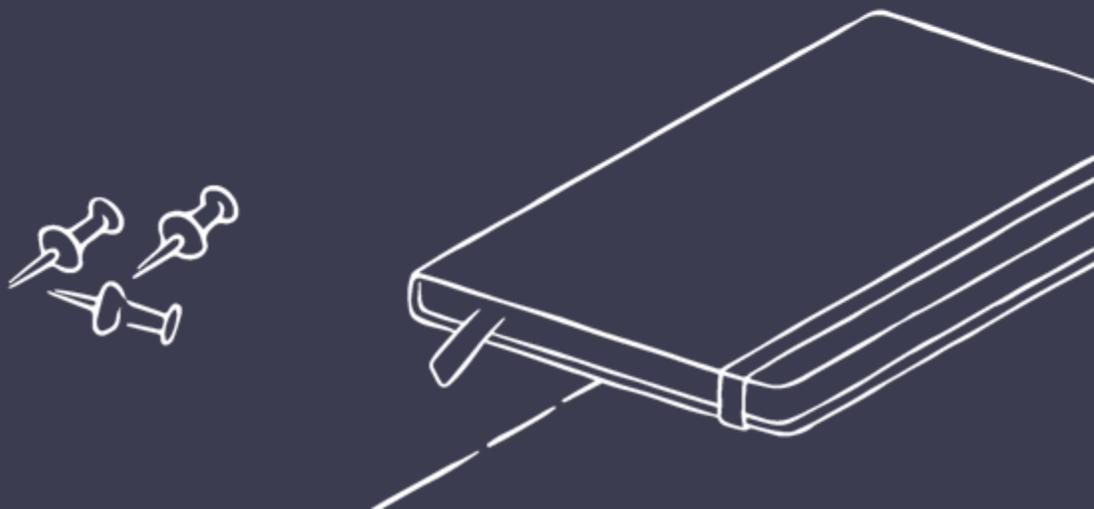
ИС-центричный подход к описанию ИТ-ландшафта

01 Хронология появления программных архитектур

02 ИТ-системы

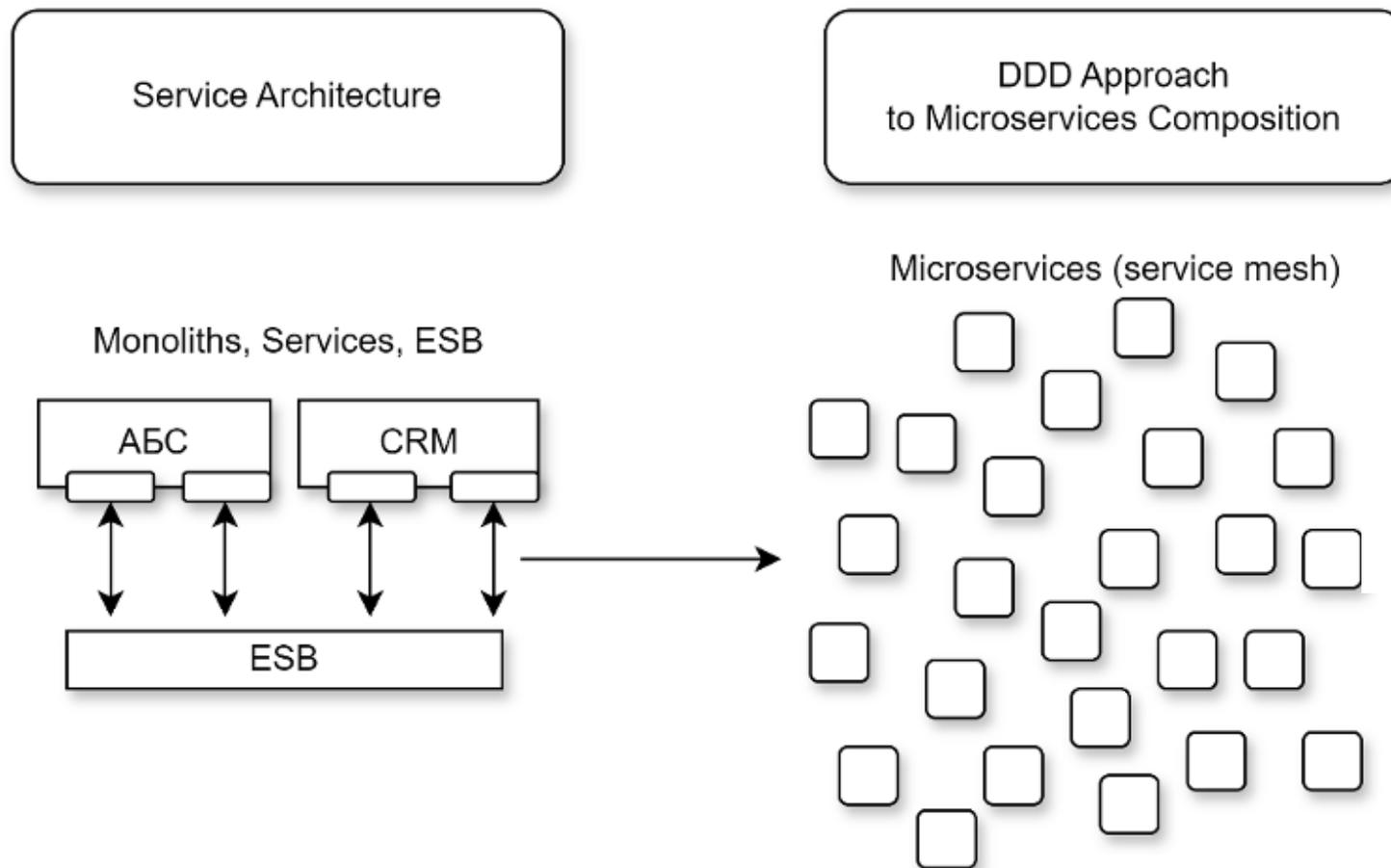
03 Как создаются ИТ-системы ?

04 Проблемы ИС-центричного подхода к описанию ИТ-ландшафта



Хронология появления архитектурных стилей

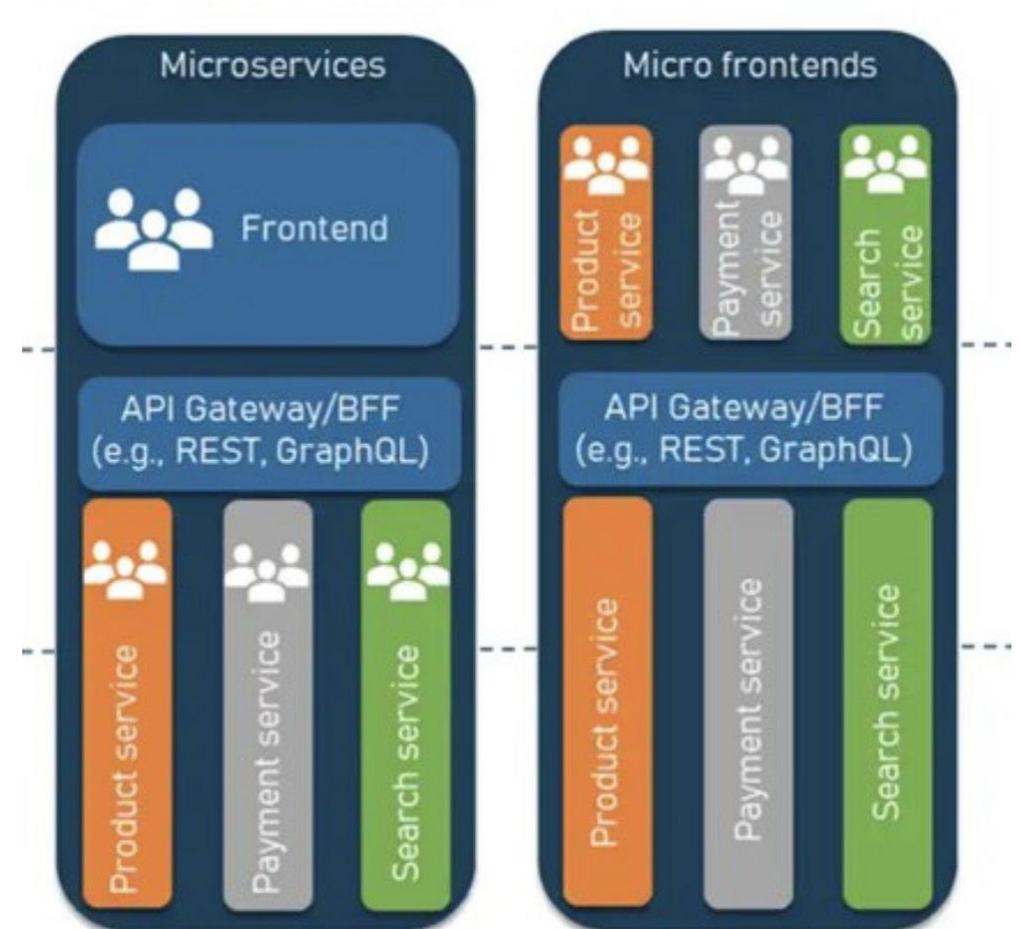
Каждый архитектурный стиль решает свои задачи



Микросервисная архитектура

Очевидные преимущества

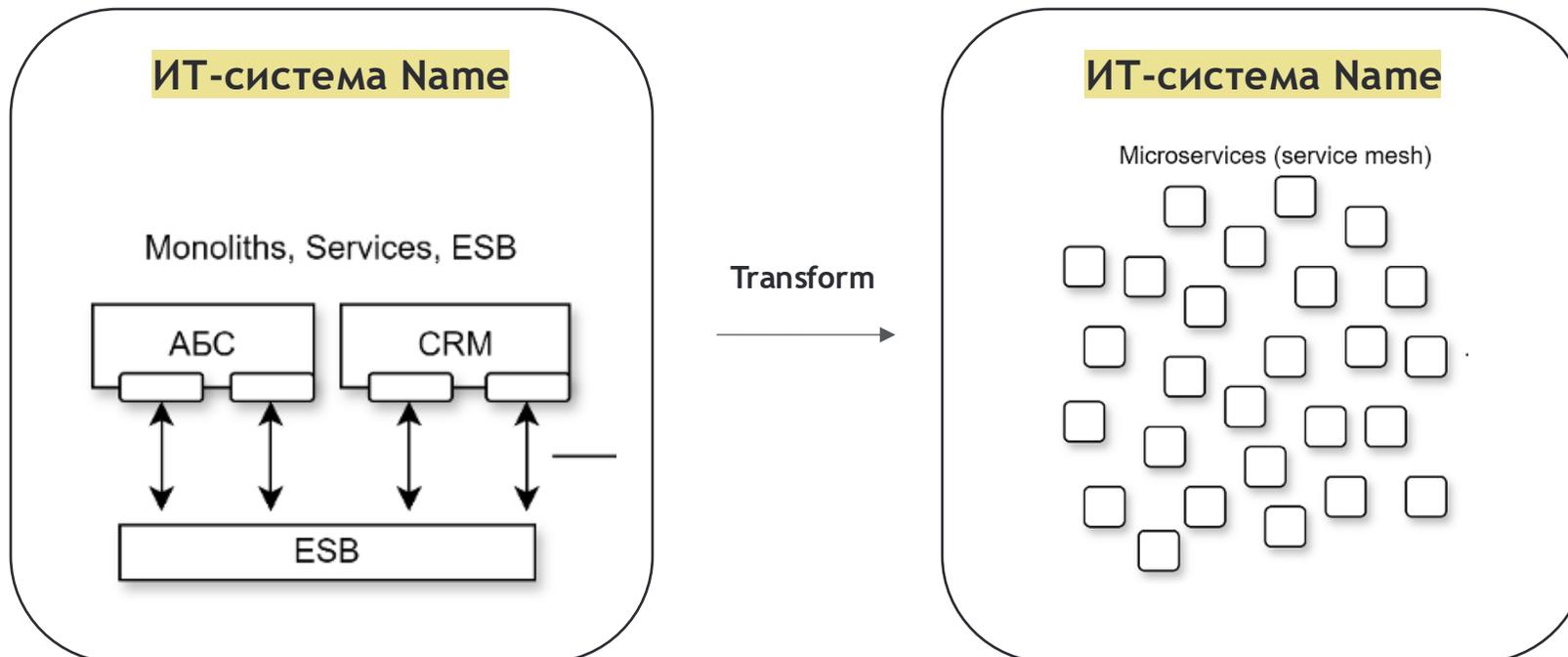
- Построение небольших продуктовых команд - идеально подходит под agile-трансформацию
- Установка на продуктивную среду по готовности задачи (CI/CD, feature toggles, fast TTM)
- Простое горизонтальное масштабирование
- Сервисы меньше и понятнее для разработчика
- Сервисы изолируются в отдельных процессах/контейнерах
- Проще использовать новые технологии
- Для каждого сервиса можно настроить свое инфраструктурное окружение, подобрать ресурсы, наиболее подходящие для решения поставленной задачи



А причем тут ИТ-системы ?

На самом деле ИТ-системы всегда служили инструментом управления сложностью

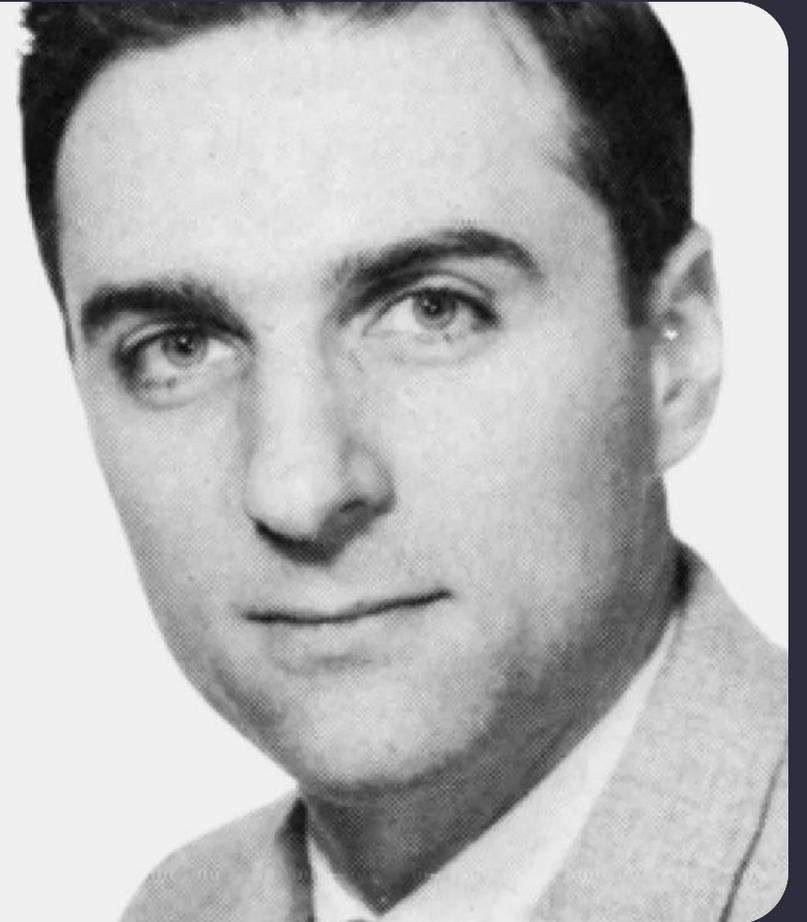
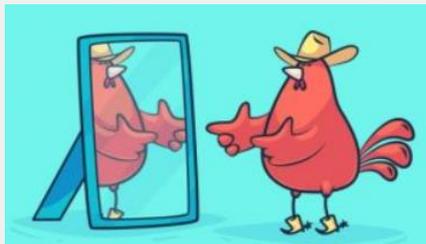
- ИТ-система - это ключевой логический объект управления в ИТ-процессах, одновременно объект архитектурного проектирования, производства, последующего сопровождения и учета затрат
- При этом совершенно неважно, какой архитектурный стиль был выбран при проектировании - ИТ-система инкапсулирует сложность реализации реальных объектов (например, микросервисов)



Как мы обычно создаем новые ИТ-системы ?

Conway's Law

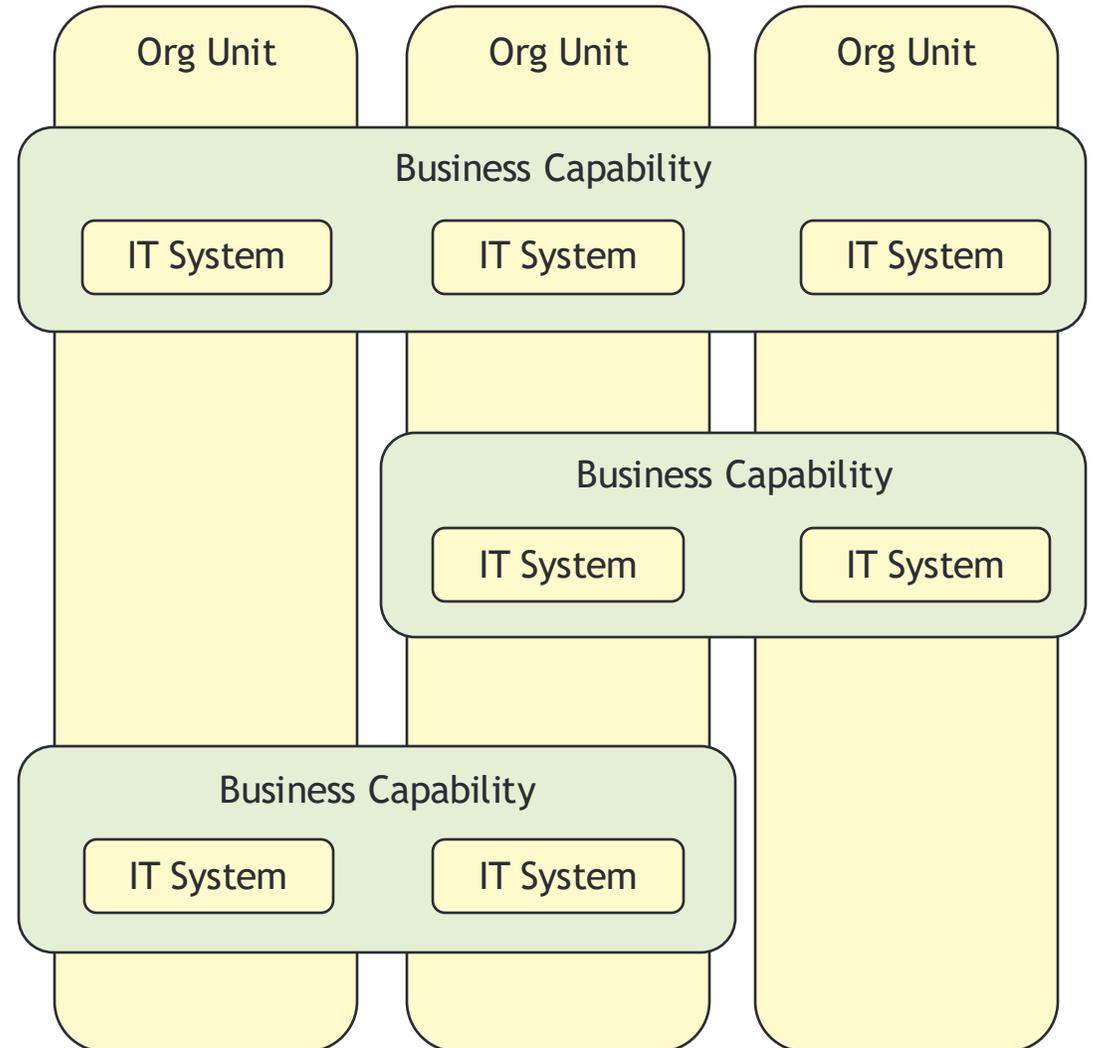
Any organization that designs a system will produce a design whose structure is a copy of the organization's internal communication structure.



Проблема

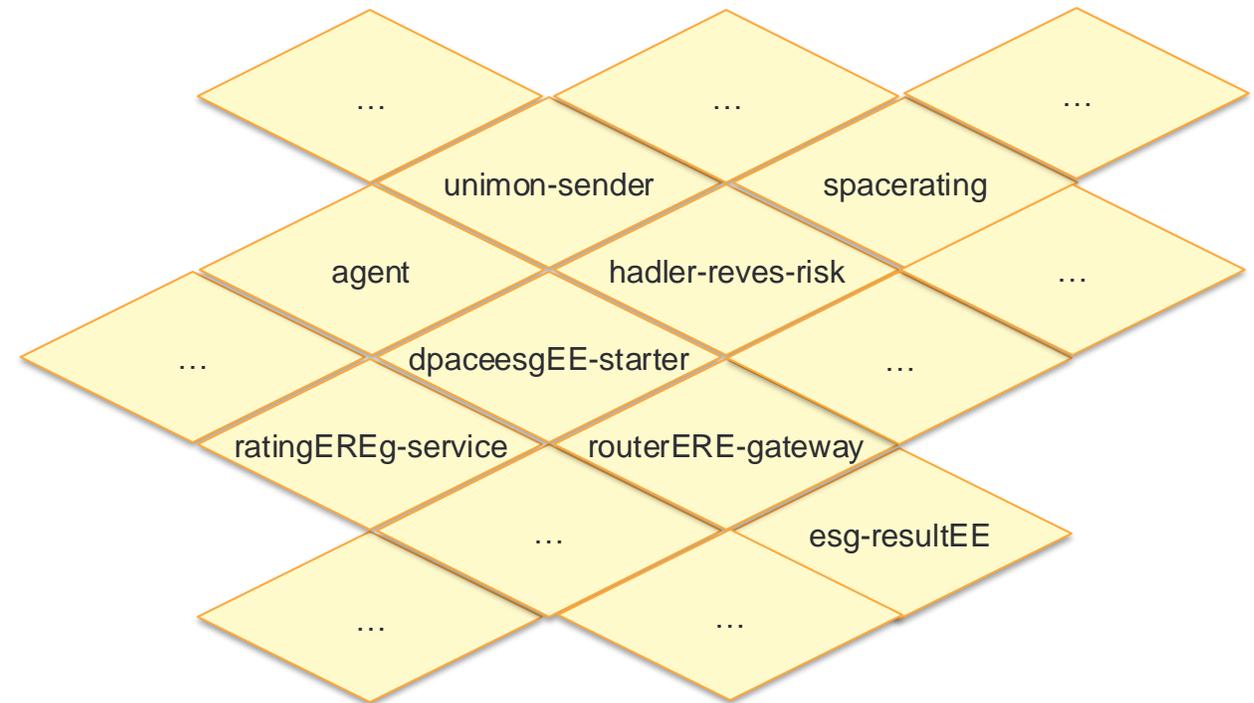
Что стало не так с ИТ-системами ?

- Закон Конвея приводит к «феодализации» ИТ-ландшафта - сначала вокруг Доменов, а в пределе вокруг доменных Команд - в ИТ-системах «паркуется» несвойственная функциональность под решение конкретных задач Доменов
- Развитие продуктов приводят к появлению все большего количества микросервисных архитектур, что увеличивает число ИТ-систем - в пределе один микросервис = одна ИТ-система
- ИТ-система перестает эффективно решать ключевую архитектурную задачу по снижению энтропии ИТ-ландшафта



Уже сложно управлять текущим количеством компонентов и связей микросервисов

- ~1000 ИТ-систем
- ~100 сервисов в одной ИТ-системе
- больше 100 точек взаимодействия/API
- до N-кварталов на синхронизацию ожиданий команд при запуске сложных интеграционных продуктов
- зрелые команды УЖЕ сами озаботились решением задач описания своих системных архитектур, внедрением различных inventory-инструментов





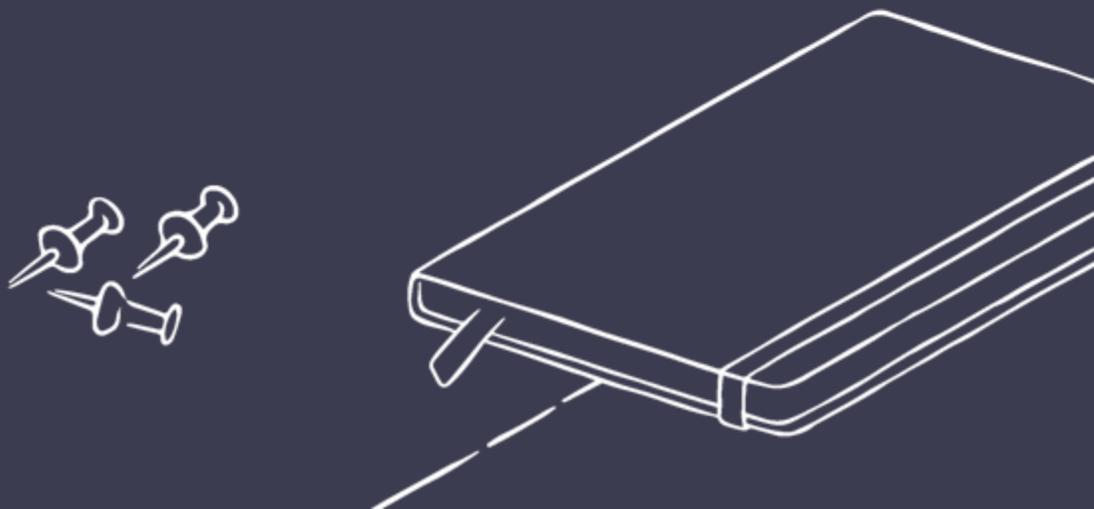
Composable architecture

01 Продолжение эволюции архитектурных стилей

02 Концепция Composable Business от Gartner

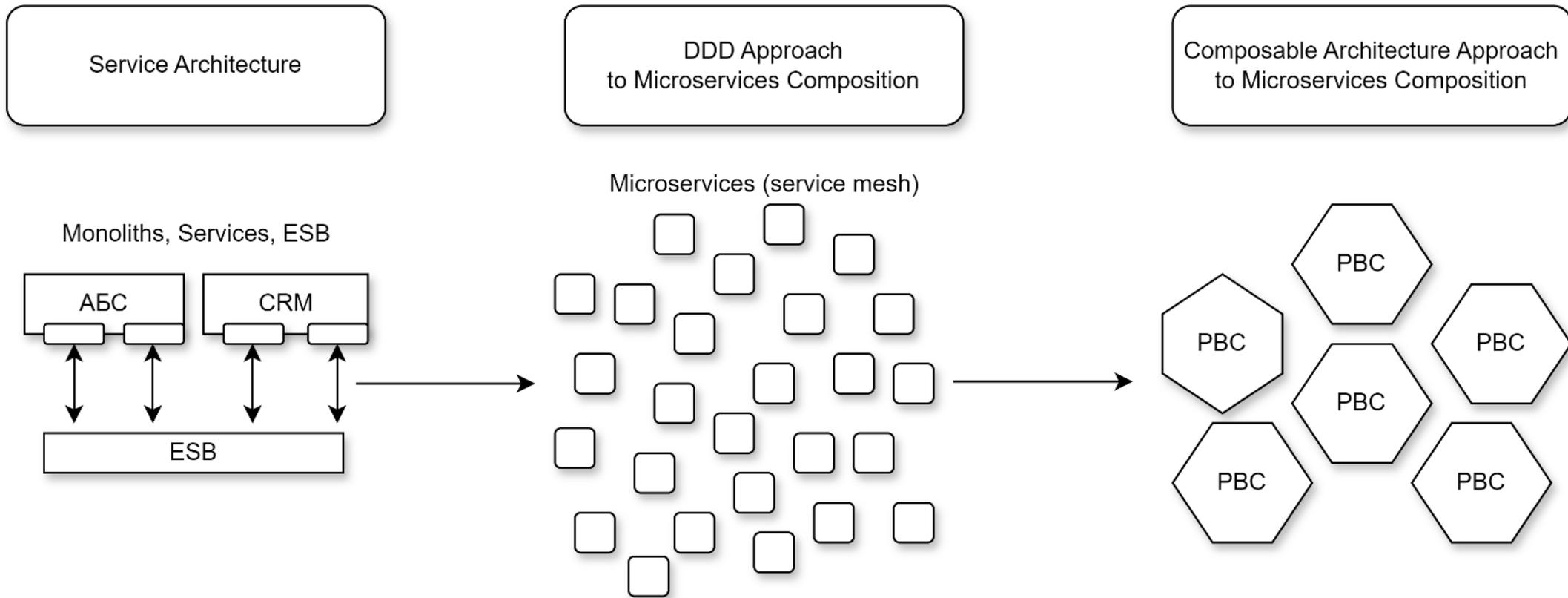
03 Что такое Package Business Capabilities (PBC) ?

04 Достоинства композитной архитектуры



Продолжение эволюции архитектурных стилей

Composable Architecture - обсудим далее, что это такое



Тренды 2021

Gartner Top Strategic Technology Trends for 2021



2021

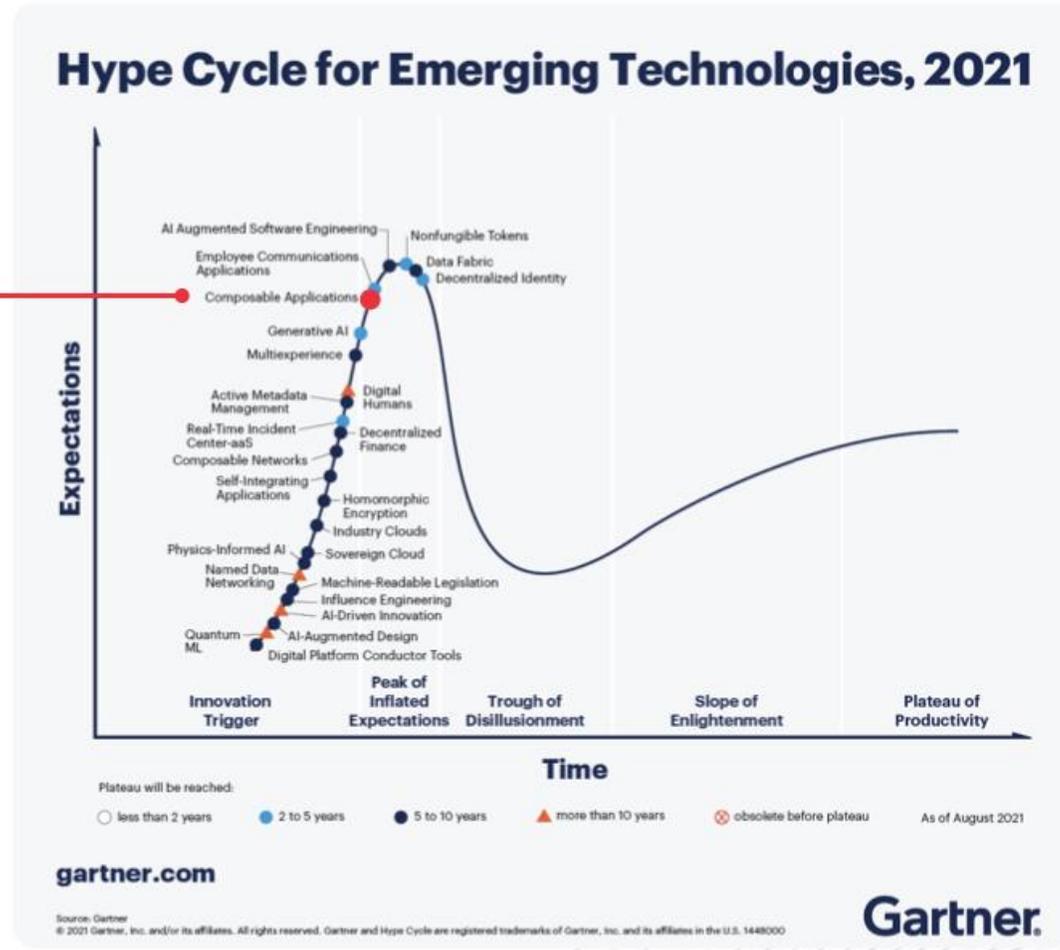


“By 2023, organizations that have adopted a composable approach will outpace competition by 80% in the speed of new feature implementation.”

Тренды 2021

Gartner Top Strategic Technology Trends for 2021

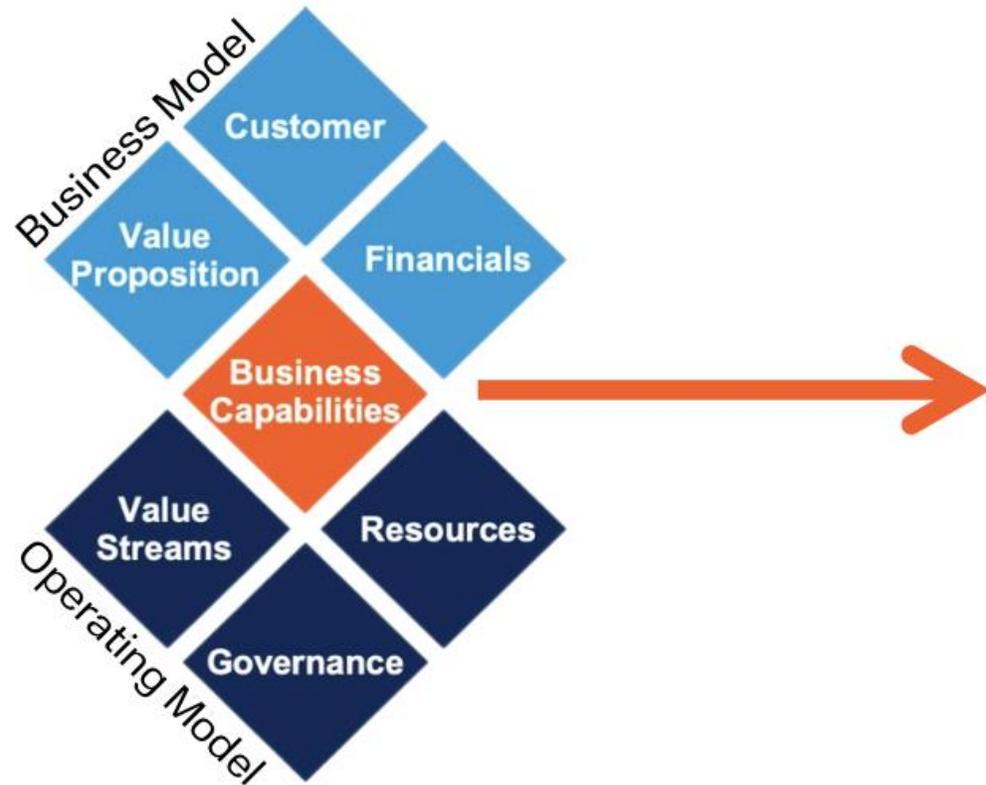
Composable Applications



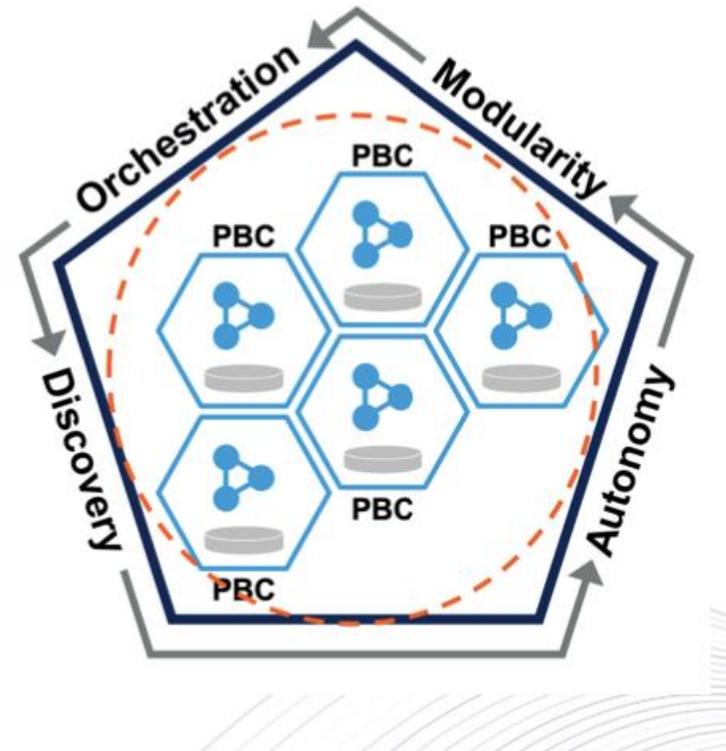
Composable business

Концепция Gartner

Composable Business



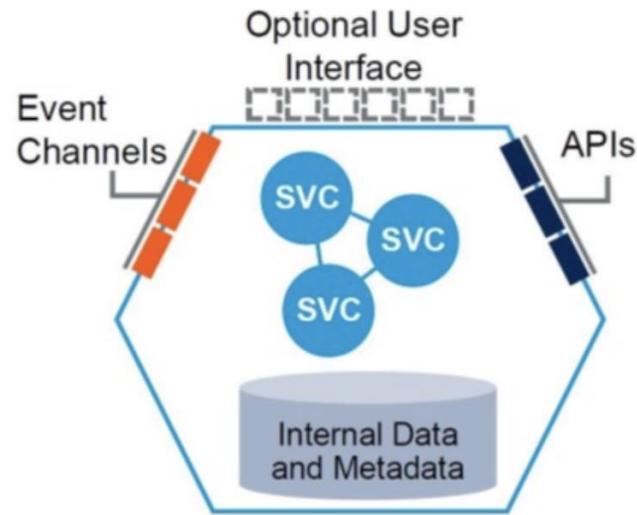
Composable Applications



Packaged Business Capabilities (PBC)

Композиционный блок - новый ключевой объект управления в ИТ-процессах

- Реализует четко определенную независимую бизнес-способность (business capability)
- Обеспечивает управление определенным бизнес объектом (business API) или предоставление информации (data API)
- Является объектом управления, инкапсулирующим данные, логику и процессы
- Достаточно автономный, скрывает свое внутреннее устройство, предоставляя четко определенный контракт (API, SLA)



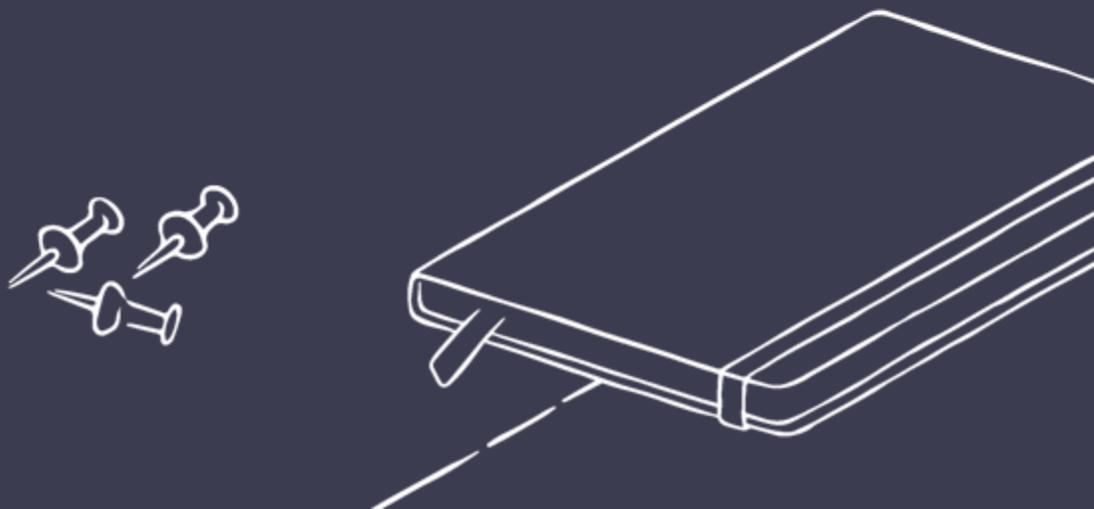
- **Целостность** - Дизайн ИТ-ландшафта в соответствии с иерархией бизнес способностей. Единая ответственность за каждую бизнес способность.
- **Автономность** - Данные и процессы скрыты от потребителя бизнес-ценности внутри PBC. Взаимодействие с PBC только через опубликованные интерфейсы. Минимизация внешних зависимостей.
- **Надежность** - наблюдаемость и управляемость

Достоинства композитной архитектуры

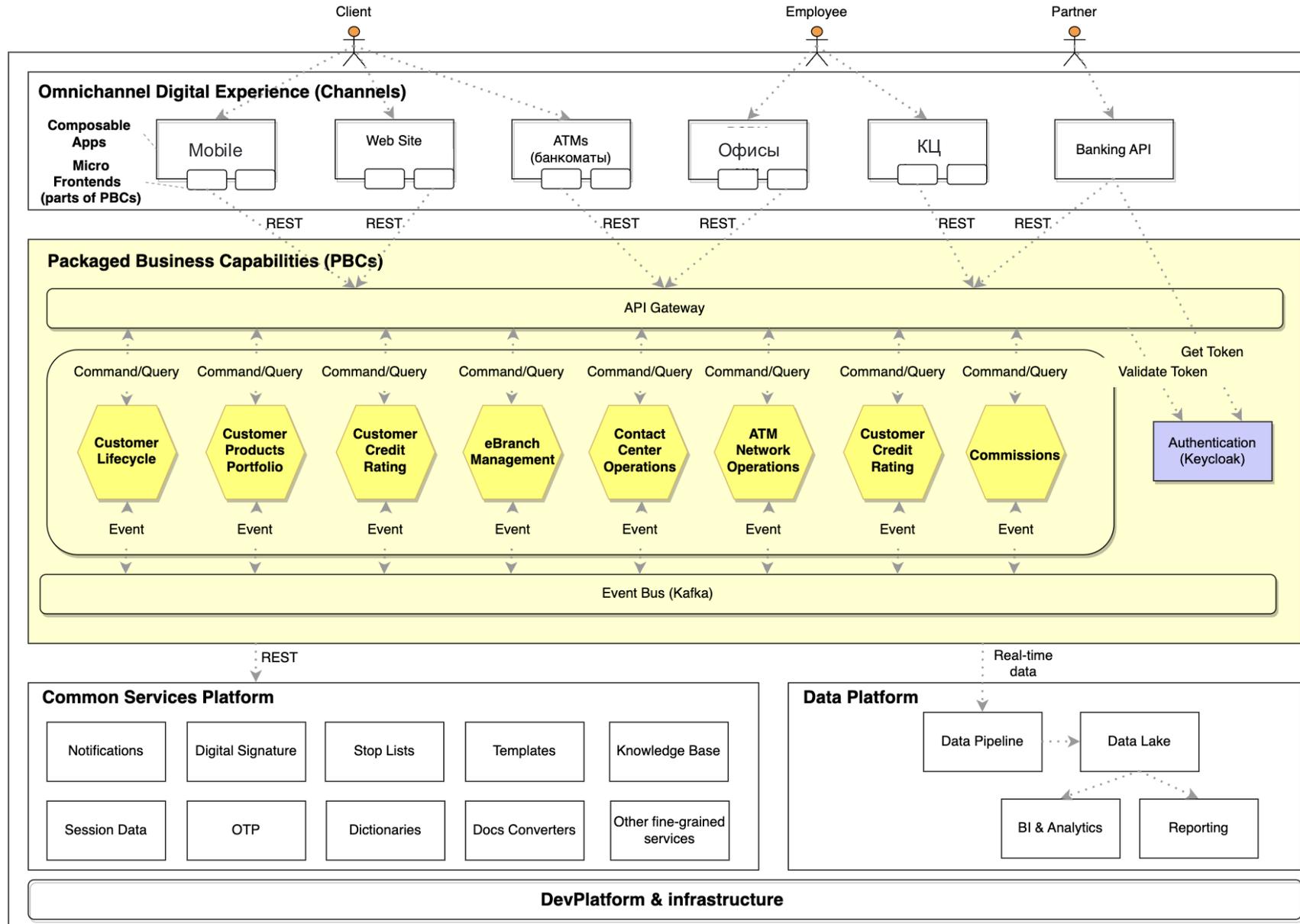
1. Улучшение наблюдаемости и управляемости, снижение сложности внесения изменений в ИТ-ландшафт
2. Выравнивание бизнес-стратегии и бизнес-способностей с ИТ-ландшафтом
3. Четкое разделение ответственности и улучшение возможности повторного использования
4. Снижение сложности интеграций и меж-командного взаимодействия за счет инкапсуляции
5. Снижение количества межсистемных API в 5-10 раз (группировка API по бизнес способностям в API Portal)
6. Решение текущей проблемы “Конфигурационная единица = система”, когда драйвером создания новой системы порой является отсутствие нужной КЕ



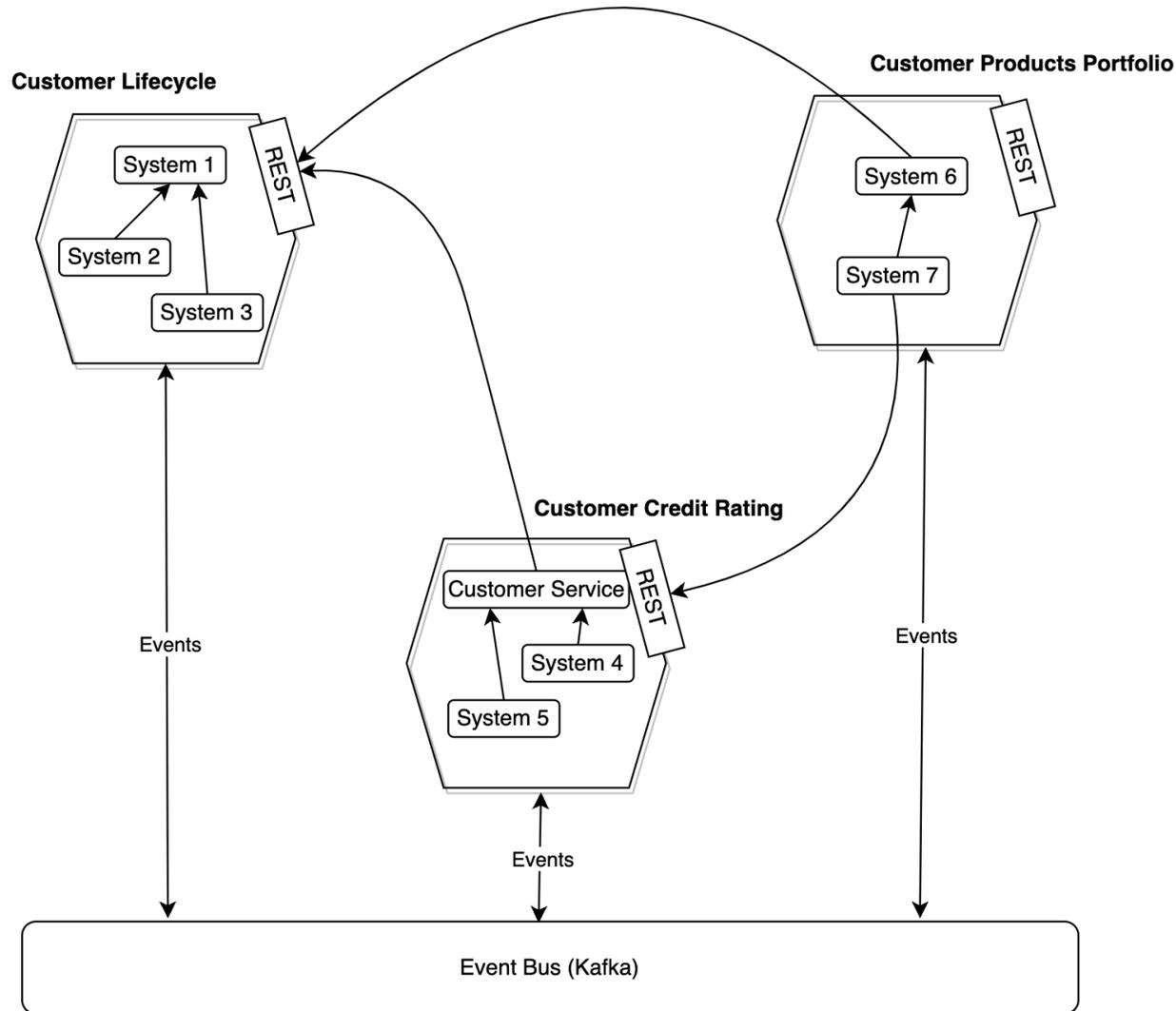
Новое представление ИТ-ландшафта с использованием принципов Composable architecture



Новое представление целевого ИТ-ландшафта



Новое представление целевого ИТ-ландшафта



- Удобная группировка ИС, но не по функциональному принципу (зачастую организационному), а в соответствии с принципами Business Capability
- Новая возможность повышения абстракции - снижения сложности - для принятия стратегических решений (и без потери качества из-за «отсутствия бизнес-процессов»)

Внедрение Composable Architecture

Архитектурные работы

1. Review and expand Business Capabilities map (inputs are Business vision, Business Goals, Business Capabilities)
2. Build mapping of current IT Systems on Business Capabilities
3. Build list of PBCs and its mapping on Business Capabilities
4. Build mapping of current IT Systems on PBCs (review meaning and boundaries of IT Systems)
5. Build Composable Architecture IT Landscape
6. Build transition plan
7. Select a domain for proof of concept

Спасибо за внимание !

Москва 2024