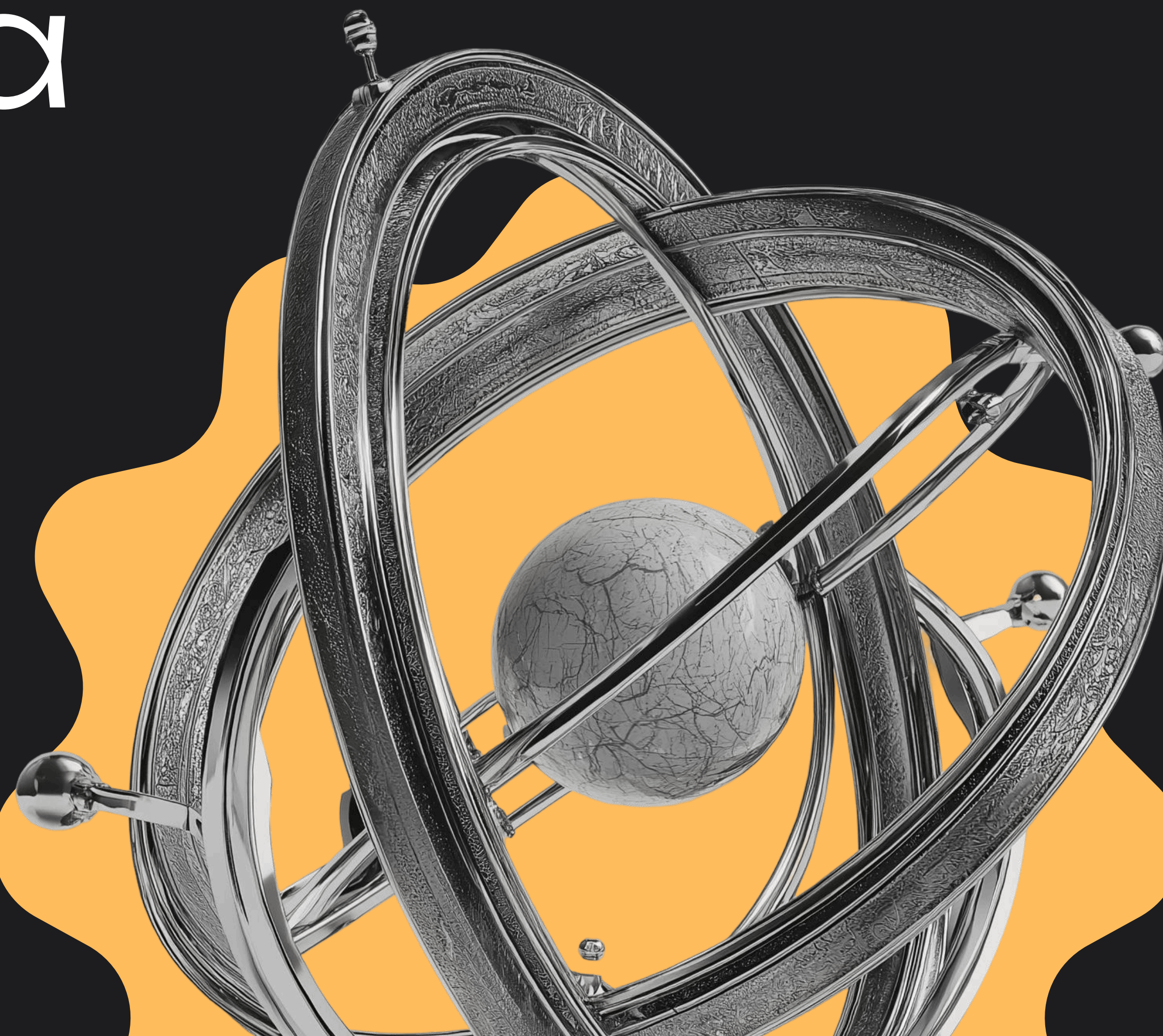


# Технологическая Платформа

Octointegro Platform

Как мы строим независимую  
инженерную экосистему



# Содержание

Экономический эффект

Цели платформы

Компоненты

Compute

IDP

Data Platform

Application Performance  
Management

Ситуационный центр

Integration Platform

Machine Learning

AI-агенты

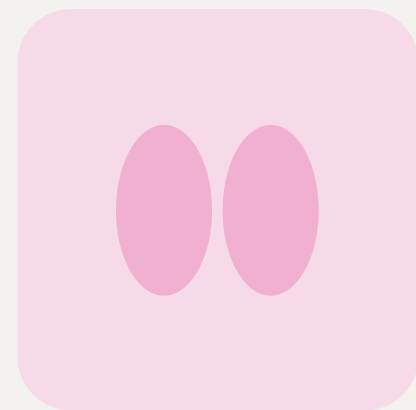
AI Dev Pipeline

Design System

Центры компетенции и SSDLC

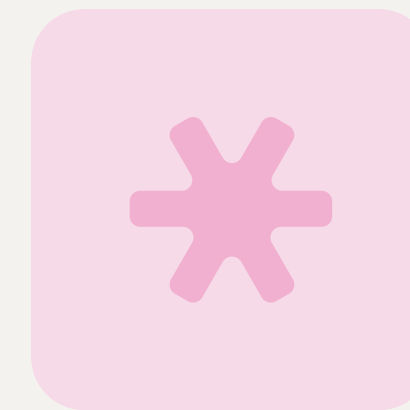
Результаты

# Цели платформы



## Технологическая независимость от вендоров

Платформа полностью построена на open-source решениях, что исключает лицензионные ограничения и обеспечивает полный контроль над технологическим стеком и его развитием.



## Единые стандарты и повторяемые паттерны

Унифицированные шаблоны для всех сервисов и окружений. Infrastructure as Code обеспечивает тиражирование одного решения по всей организации.



## Безопасность

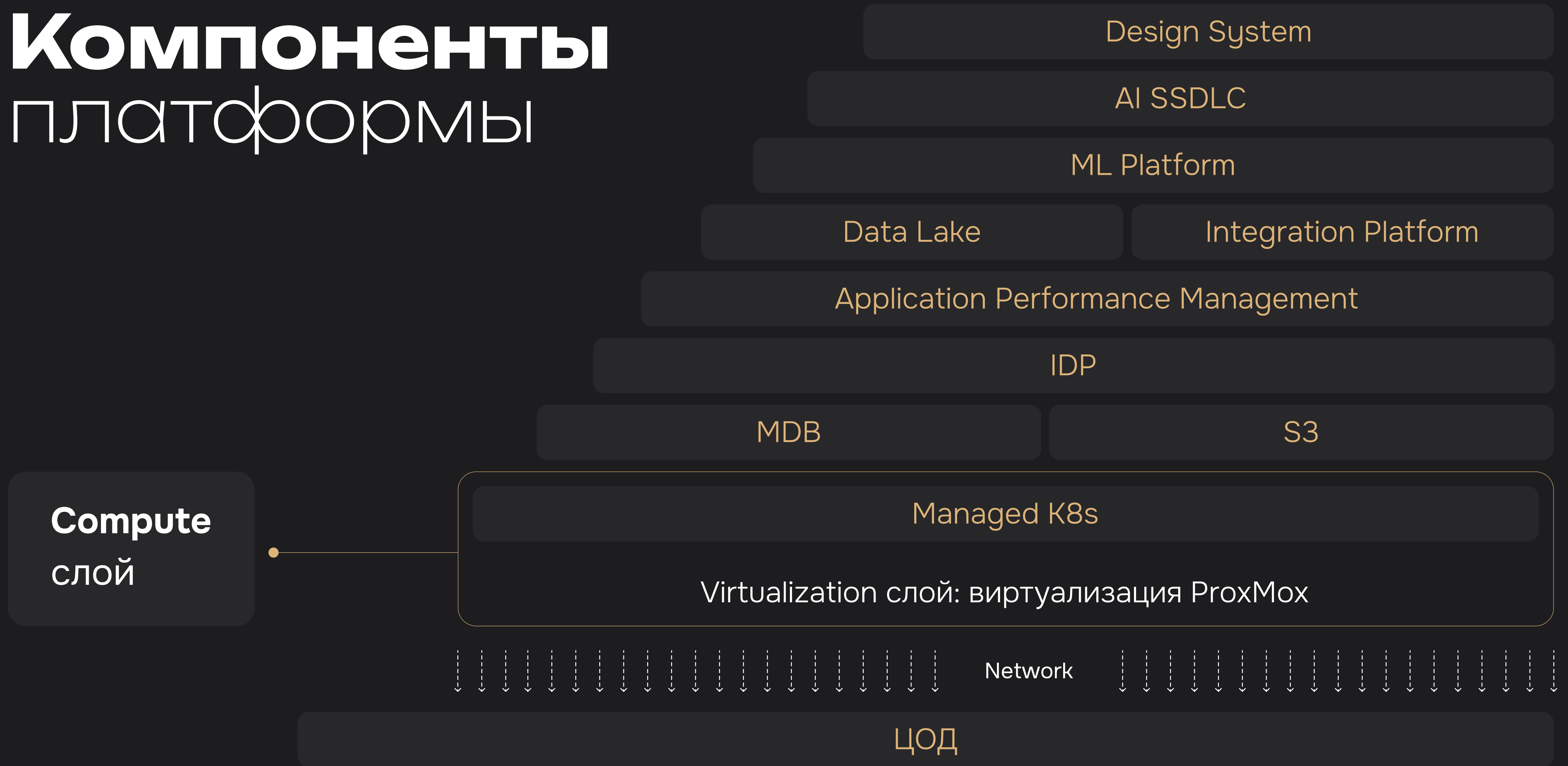
Встроенные механизмы RBAC, сетевой сегментации и compliance-контроля обеспечивают защиту данных и соответствие корпоративным стандартам безопасности на всех уровнях платформы.



## Developer Self-Service

Продуктовые команды получают полную автономию: самостоятельно создают окружения, деплоят сервисы и управляют ресурсами – без ожидания и блокирующих зависимостей.

# Компоненты платформы



# Compute СЛОЙ

## Преимущества



### Стабильность

Минимизация простоев и оптимизация мощностей под пиковые нагрузки



### Предсказуемость

Стандартизированное развёртывание окружений – команды фокусируются на продукте, а не инфраструктуре



### Масштабируемость

Гибкая адаптация к росту бизнеса без дополнительных вложений в сложные настройки

## Ключевые возможности

- \* Надежное и простое управление вычислительными ресурсами для бесперебойной работы бизнес-процессов
- \* Стандартизированное и оперативное развёртывание любых окружений
- \* Гибкая масштабируемость без дополнительных действий
- \* Контейнерная оркестрация:  
Единая среда выполнения для продуктовых и платформенных сервисов с декларативным управлением
- \* Адаптация под любые среды – виртуализация или bare metal



и др.

- \* Работает везде – в вашем ЦОД или на арендованных серверах
- \* Реальный опыт – внедрения на уровне СНГ во всех типах инфраструктур

# IDP

## Проблема 🙄

Продуктовые команды тратят время на ожидание инфраструктурных ресурсов и согласование окружений

## Решение 😊

- ✳ Сокращение времени от идеи до работающего окружения
- ✳ Разработчики сосредоточены на продукте, а не на инфраструктуре
- ✳ Прозрачность: все ресурсы и сервисы компании в одном каталоге

## Инструменты

### Managed k8s

удобное управление инфраструктурой для полного контроля и оптимизации

### MDB

Managed Database по кнопке, автоматическое бэкапирование, репликация, failover из коробки

### PaaS

One-click deployment для окружений любой сложности



### Мониторинг и наблюдаемость

Автоматическая проверка соответствия политикам сетевой безопасности, compliance и аудит



### Управление доступами

Встроенные механизмы управления доступами на основе ролевой модели RBAC

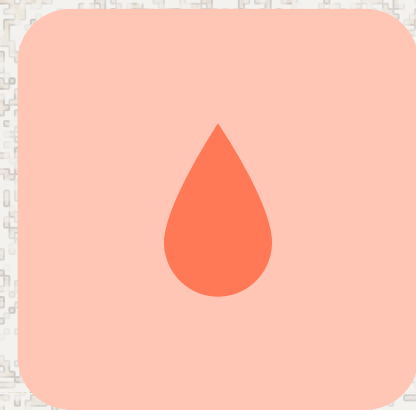


### Встроенные проверки безопасности

Автоматическая проверка соответствия политикам сетевой безопасности, compliance и аудит

# Data Platform

Данные как актив: единая платформа обеспечивает управляемый, прозрачный и переиспользуемый доступ к данным для всех команд.



## Data Lake

Консолидация разнородных данных. Несколько хранилищ, объединённых под разные задачи:

- \* исторические данные;
- \* медиафайлы;
- \* потоковые события;
- \* структурированные записи;



## Каталог данных

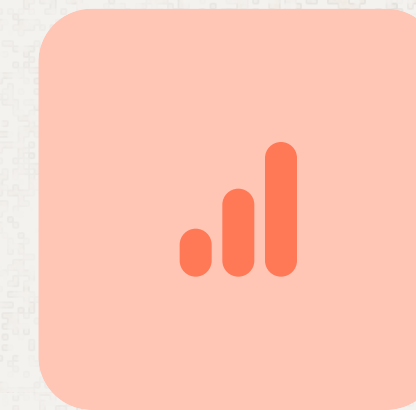
Единая точка правды – знаем, что и где лежит в каждой базе. Поиск, описание и управление метаданными из всех источников в одном интерфейсе.

Возможность роста под



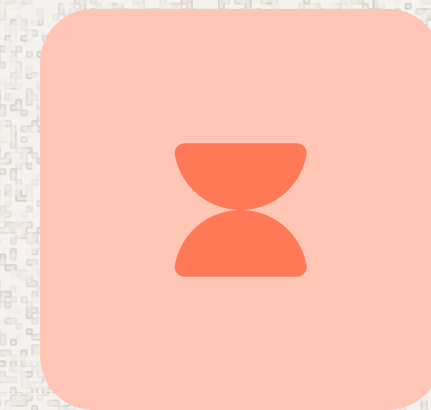
DataFABRIC

DataMesh



## BI и аналитика

Автоматическое построение BI-отчётов, обновляемых в реальном времени. Стратегические решения – без участия аналитика.



## Жизненный цикл данных

Унификация обработки данных. От сырых данных до архивов.

# Application Performance Management

Полное покрытие всей инфраструктуры метриками, логами и трейсами "из коробки" без дополнительной настройки

## Проблема 🙄

Долгий поиск причин сбоя (MTTR – Mean Time To Resolution)



Слепые зоны в микросервисной архитектуре



Высокий порог входа для разработчиков



## Решение 😊

Было: Инцидент → Logs → Гипотезы → Расследование = 4 часа

Стало: Инцидент → Трейс → Log → Fix = 15 мин

Нет участков инфраструктуры без метрик.  
Автоматический граф зависимостей и сквозные трейсы показывают, где на самом деле корень проблемы.

Не надо настраивать Jaeger + Prometheus + Loki отдельно.  
Разработчик получает всё сразу на любой среде (dev/staging/prod).



# Ситуационный центр

## Цели

- ✦ Раннее и проактивное выявление Аварий и Проблем, предотвращение крупных Инцидентов;
- ✦ Снижение простоев и потерь;
- ✦ Быстрая координация команд при Авариях и кризисах;
- ✦ Прозрачность процесса для участников, руководства и пользователей;

## Составляющие

### Люди



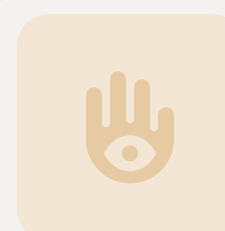
Team Lead



Команды и дежурные

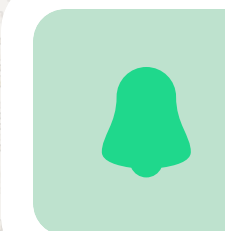


Problem Manager

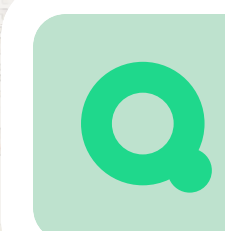


Incident Manager

### Процессы



Incident (Accident) management



Problem management

### Инструменты

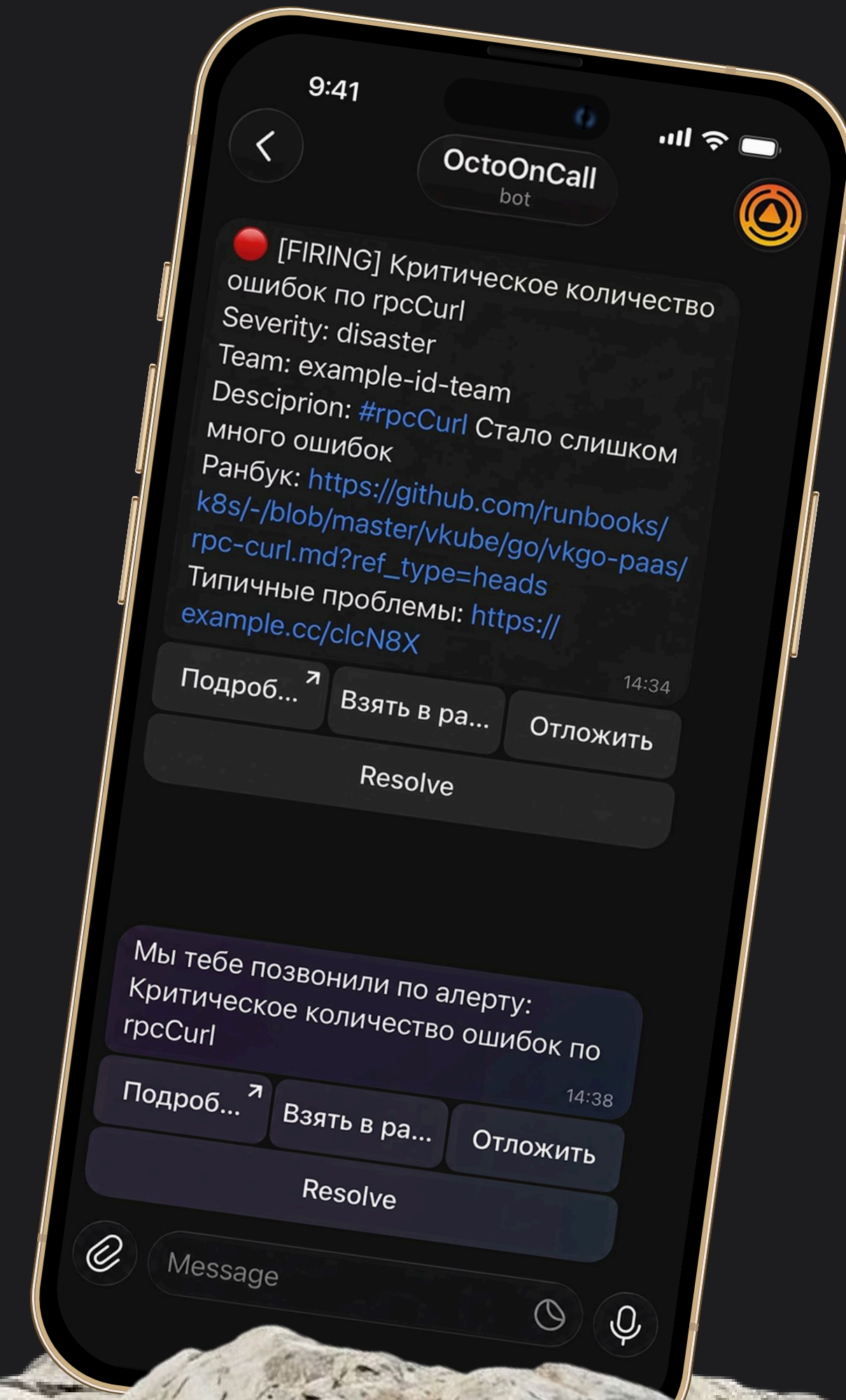


- ✦ Jira and Confluence Процессы, статистика и управляемость, SLA и сроки, автоматизации процессов, База знаний;
- ✦ Grafana Dashboards Визуализация текущего состояния критических систем для команд поддержки;
- ✦ Grafana OnCall Мониторинг и дальнейшая эскалация алертов по заданным цепочкам и расписаниям, с интеграциями с tg, Телефонией, Jira, Teams, график дежурств;

# Grafana OnCall

Автоматизация цепочек эскалаций, интеграции с Telegram, телефонией, Jira и Teams, управление графиком дежурств.

Единая точка маршрутизации алертов: от срабатывания до ответственного инженера — без ручных переключений и потерянных уведомлений, ускоряя реагирование на инциденты.



# Integration Platform

## Единая шина данных

- \* Единый стандарт взаимодействия всех компонентов платформы;
- \* Безопасная передача данных между системами;
- \* Работа в реальном времени (real -time) с возможностью встраивания в поток;

## API Gateway

- \* Управляемые API (Managed API) из коробки;
- \* Контроль доступа – аутентификация, авторизация, rate limiting;
- \* Мониторинг обращений;
- \* API as a Service – тарификация по каждому обращению;



# Machine Learning

Инфраструктура для полного цикла машинного обучения — от экспериментов до эксплуатации моделей в продакшне.



## Обучение моделей

Управляемая инфраструктура для запуска экспериментов, трекинга метрик и версионирования моделей — с поддержкой GPU-кластеров.



## Стандартные модели

Готовые к использованию модели для типовых задач: классификация, предсказание, аномалии — доступны продуктовым командам по запросу.



## Безопасность

Модели работают локально внутри инфраструктуры компании. Данные не передаются внешним провайдерам — полный контроль над информацией.



# AI-агенты

Встроены в структуру – не надстройка, а часть платформы.

## Автоаналитика

Автоматический анализ логов и метрик

Находит аномалии и проблемы до их появления

Сокращает время реакции инцидентов

## Помощь пользователям

Помощь в настройке контуров

Снижает порог входа для новых пользователей

Мгновенные ответы на вопросы

## AI для саппорта

Поиск по документации

Наблюдает за состоянием окружения и уведомляет о проблемах

Снижает TCO платформы

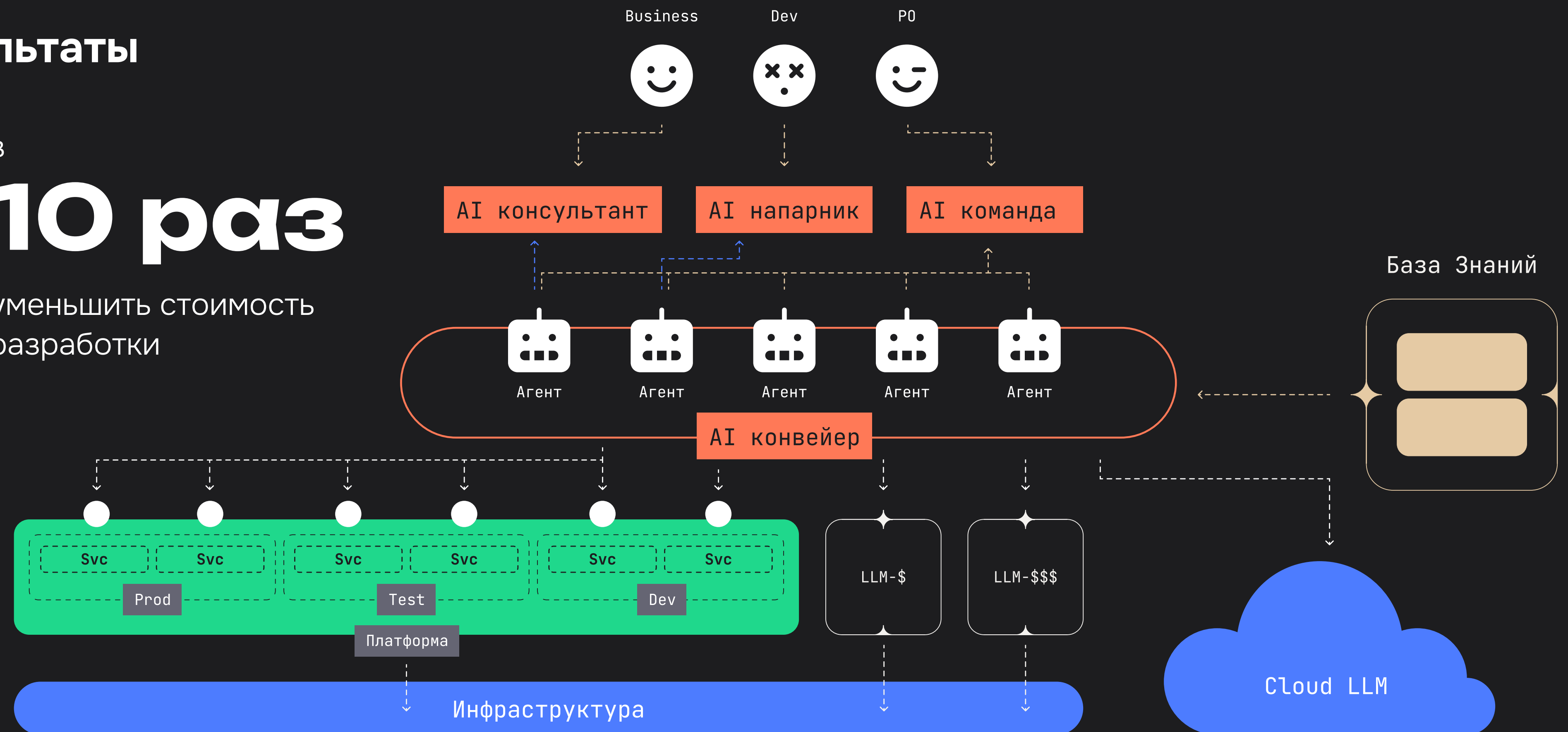
# AI Development Pipeline

## Ключевые результаты

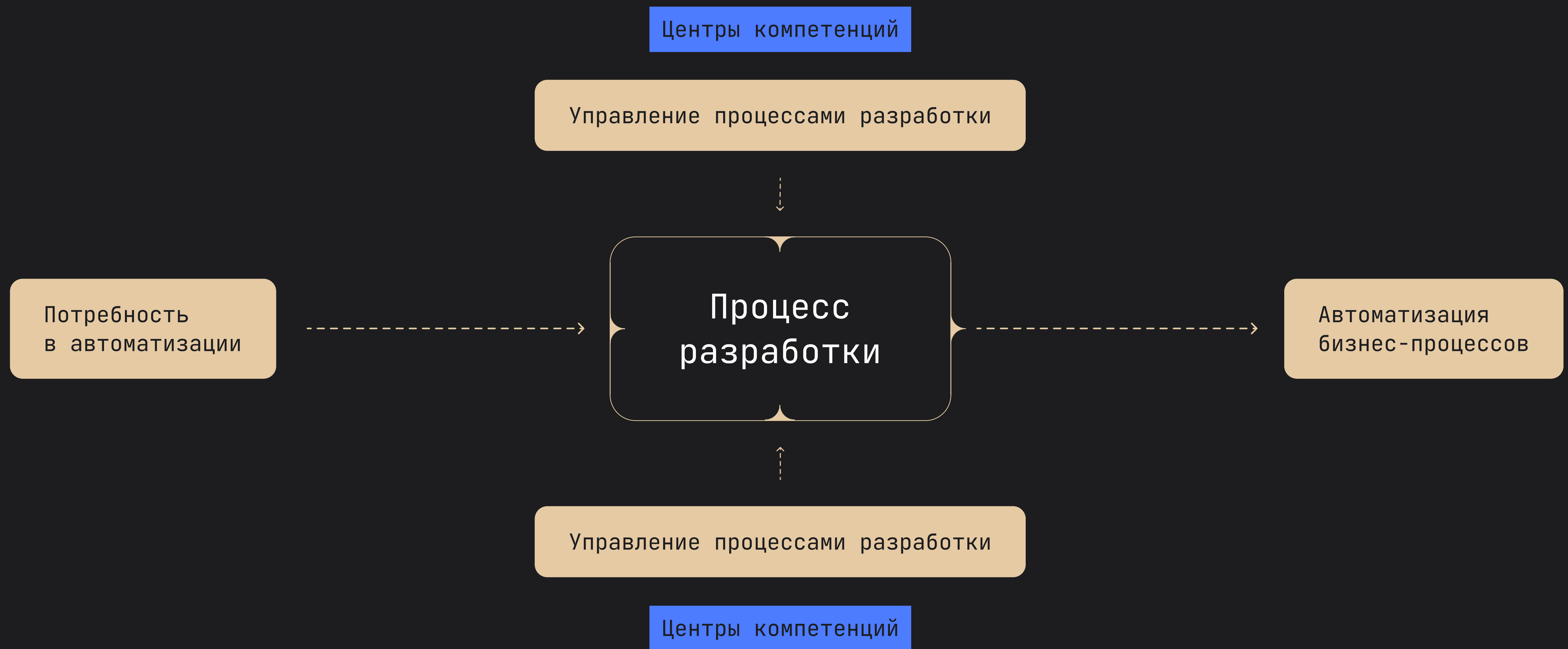
на **90%** сократить Cycle Time

в **10 раз** уменьшить стоимость разработки

на **80%** сократить время вывода продуктов



# Центр компетенций



# Security Software Development Life Cycle

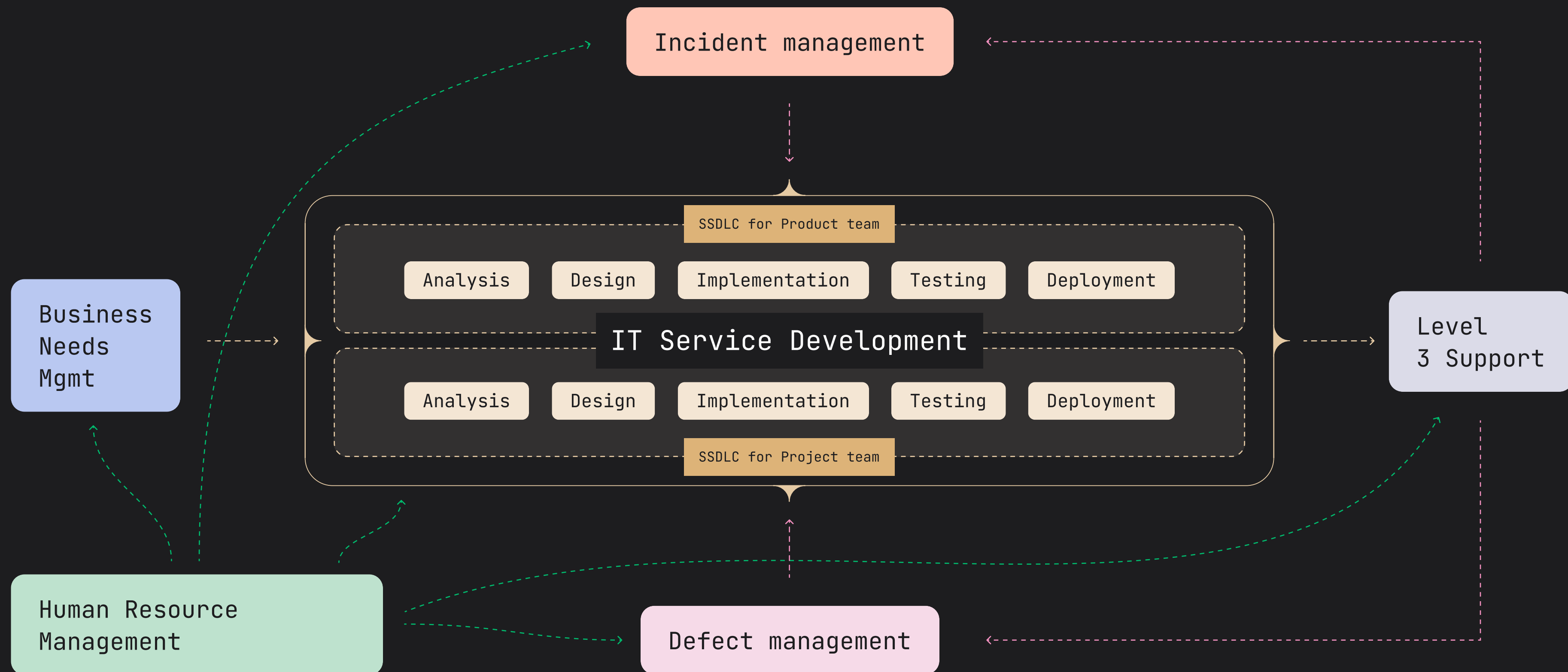
## Инженерная зрелость

- \* Единые архитектурные подходы;
- \* Непрерывная интеграция, как основа «встроенного качества»;
- \* Обязательные quality gates, как элемент интеграции;
- \* Многоуровневый контроль качества;
- \* Ориентация на контракты при разработке;
- \* Полная реализация DevSecOps принципов в разработке;

## Цели развития ИТ-процессов

- \* Снижение количества инцидентов, связанных с обновлением ИТ-инфраструктуры;
- \* Снижение количества дефектов и уязвимостей, обнаруживаемых в среде промышленной эксплуатации;
- \* Минимизация рисков информационной безопасности;
- \* Обеспечение регулярности поставок изменений и снижение среднего срока поставки;

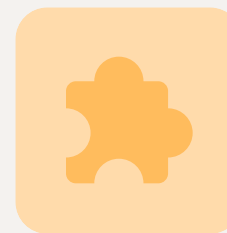
# Целевая схема процессов IT-развития



# Design System

Единый стандарт цифровых продуктов компании:  
интерфейсы любой команды выглядят и работают последовательно,  
а запуск новых продуктов не начинается с нуля.

- \* Подходит для любого типа продукта;
- \* Актуальный и современный технологический стек;
- \* 100% покрытие тестами;
- \* Подробная документация;
- \* Модульная структура;
- \* CI/CD конвейер;



## Единая библиотека компонентов

Готовые UI-блоки для всех команд – интерфейсы собираются быстрее и выглядят единообразно



## Стандарты бренда и UX

Цвета, типографика, отступы и паттерны взаимодействия зафиксированы – пользователи не замечают смену продукта



## Скорость запуска продуктов

Новые интерфейсы строятся из готовых блоков – время на разработку фронтенда сокращается кратно



## Централизованные обновления

Изменение в одном месте распространяется на все продукты – технический долг и рассинхронизация интерфейсов исчезают

# Результаты

## НЕЗАВИСИМОСТЬ

### Полный контроль над технологическим выбором

Нет лицензионного риска, нет зависимости от решений внешних вендоров. Платформа строится на open-source — компания управляет своей инфраструктурой сама.

## СКОРОСТЬ

### Продуктовые команды работают быстрее

Developer self-service устраняет ожидание ресурсов. Команды сами создают окружения, деплоят сервисы и анализируют данные — без блокирующих зависимостей.

## ЭКОНОМИКА

### Предсказуемая стоимость владения инфраструктурой

Прозрачная тарификация на ресурсы ЦОД. Никаких скрытых платежей за лицензии — вы платите только за вычислительные мощности, которые реально используете.

## КАЧЕСТВО

### Единые стандарты во всех продуктах

Шаблоны, паттерны и политики применяются ко всем сервисам автоматически. Меньше разрозненных решений, ниже технический долг, выше предсказуемость систем.

