

avito.tech

2022

Платформенный продукт

Как это делать?

Лукьянченко Александр

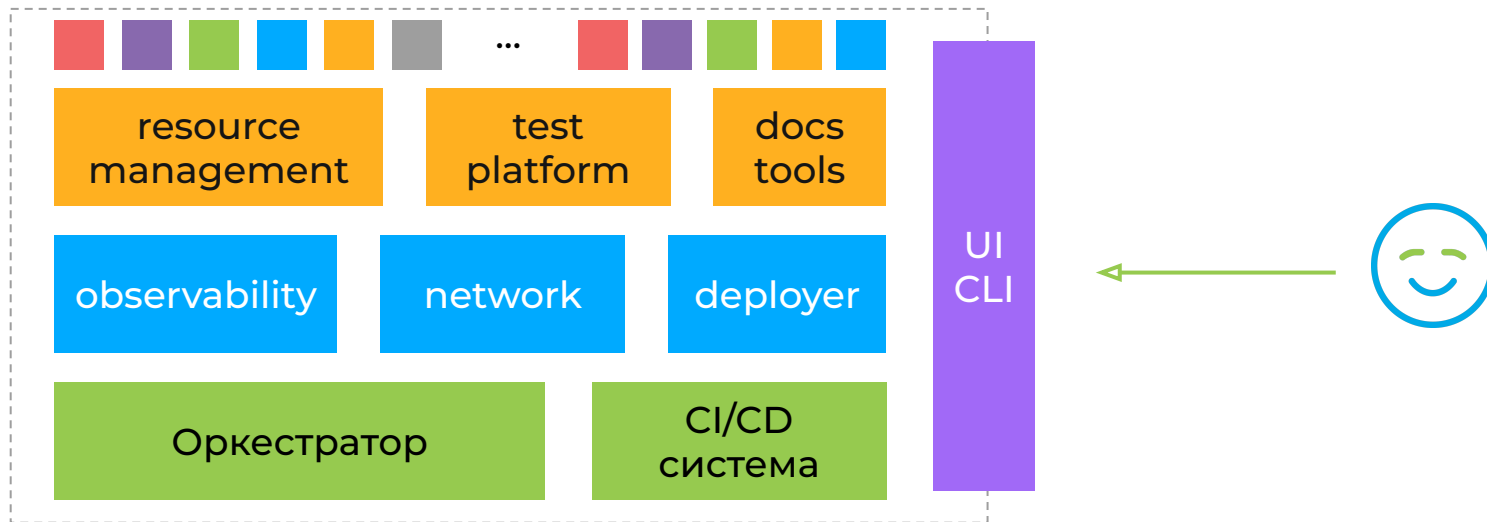
Руководитель PaaS, Avito

План

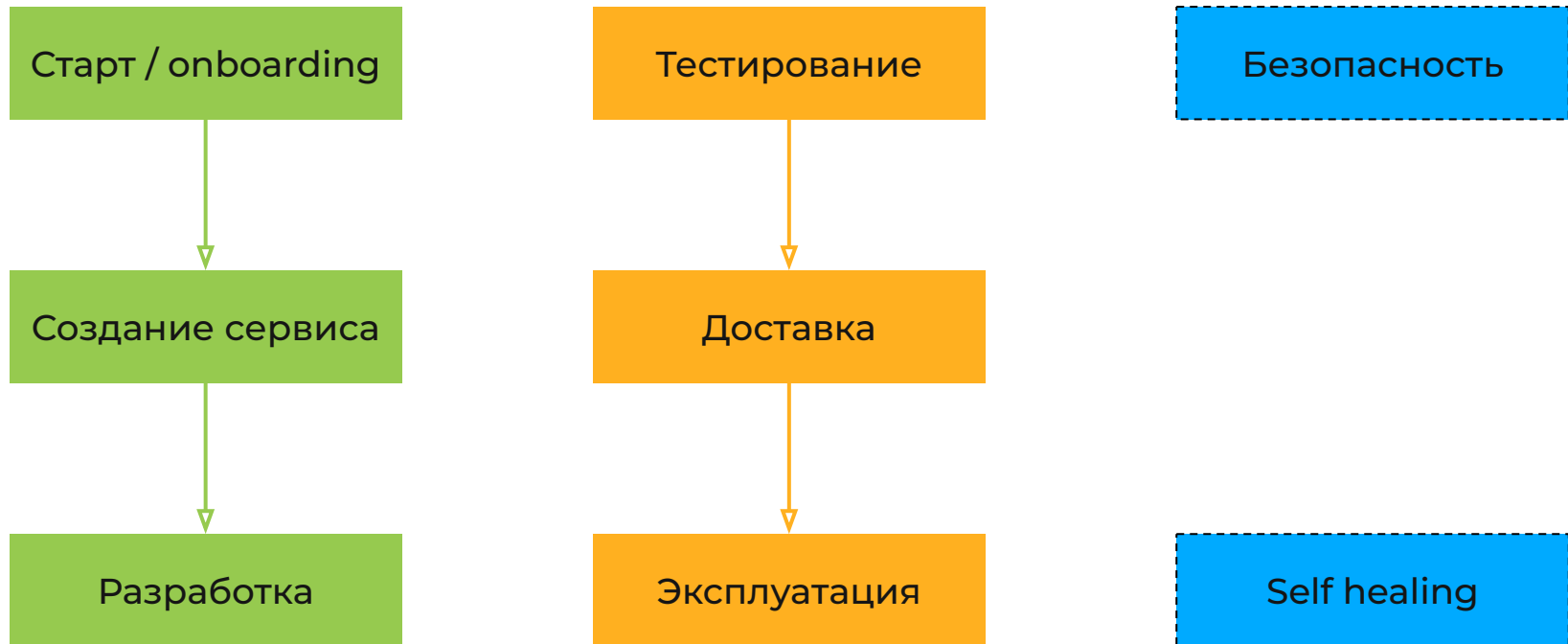
1. Что такое платформенный продукт? (PaaS/IDP)
2. Уровень автоматизации
3. Фазы зрелости и внедрение
4. Продуктовый подход
5. Check list полноты платформы

Что такое платформенный продукт?

PaaS / Internal Developer Platform



Что покрывает платформа?



Продуктовые процессы



Support



Единое окно
описания
проблем и
идей



Сбор метрик
удовлетворенности
и custdev



Discovery +
проработка
CJM

Уровень автоматизации

Поиск золотой середины

Гибкость



Простота

Поиск золотой середины

Догоняющие метрики



Объём
саппорта



Количество
запросов на
новые
ВОЗМОЖНОСТИ

Поиск золотой середины

Догоняющие метрики

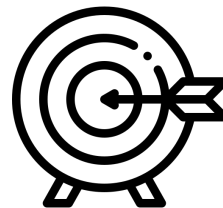


Объём
саппорта



Количество
запросов на
новые
ВОЗМОЖНОСТИ

Метрики этапа внедрения



Эффективность
автоматизаций

Ценность платформы для бизнеса

01. **Time to market:** время на onboarding, старт новых сервисов / продуктов, ускорение релизов в production



Ценность платформы для бизнеса

01. **Time to market:** время на



onboarding, старт новых сервисов / продуктов, ускорение релизов в production

02. **Фокус на ценности:** self service без необходимости



наличия экспертизы в продуктовых командах

Ценность платформы для бизнеса

01. **Time to market:** время на onboarding, старт новых сервисов / продуктов, ускорение релизов в production



02. **Фокус на ценности:** self service без необходимости наличия экспертизы в продуктовых командах



03. **Стабильность системы и качество:** унифицированные средства observability и общие quality gates



Ценность платформы для бизнеса

01. **Time to market:** время на onboarding, старт новых сервисов / продуктов, ускорение релизов в production



02. **Фокус на ценности:** self service без необходимости наличия экспертизы в продуктовых командах



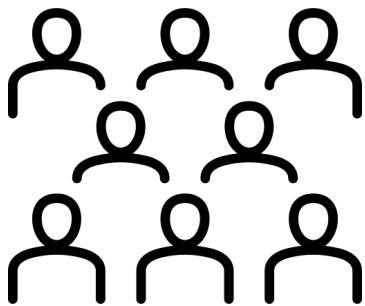
03. **Стабильность системы и качество:** унифицированные средства observability и общие quality gates



04. **Уверенность в будущем:** централизованный контроль зоопарка технологий, подходов разработки



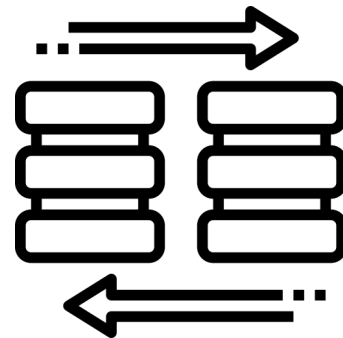
Чего это стоит?



Разработка



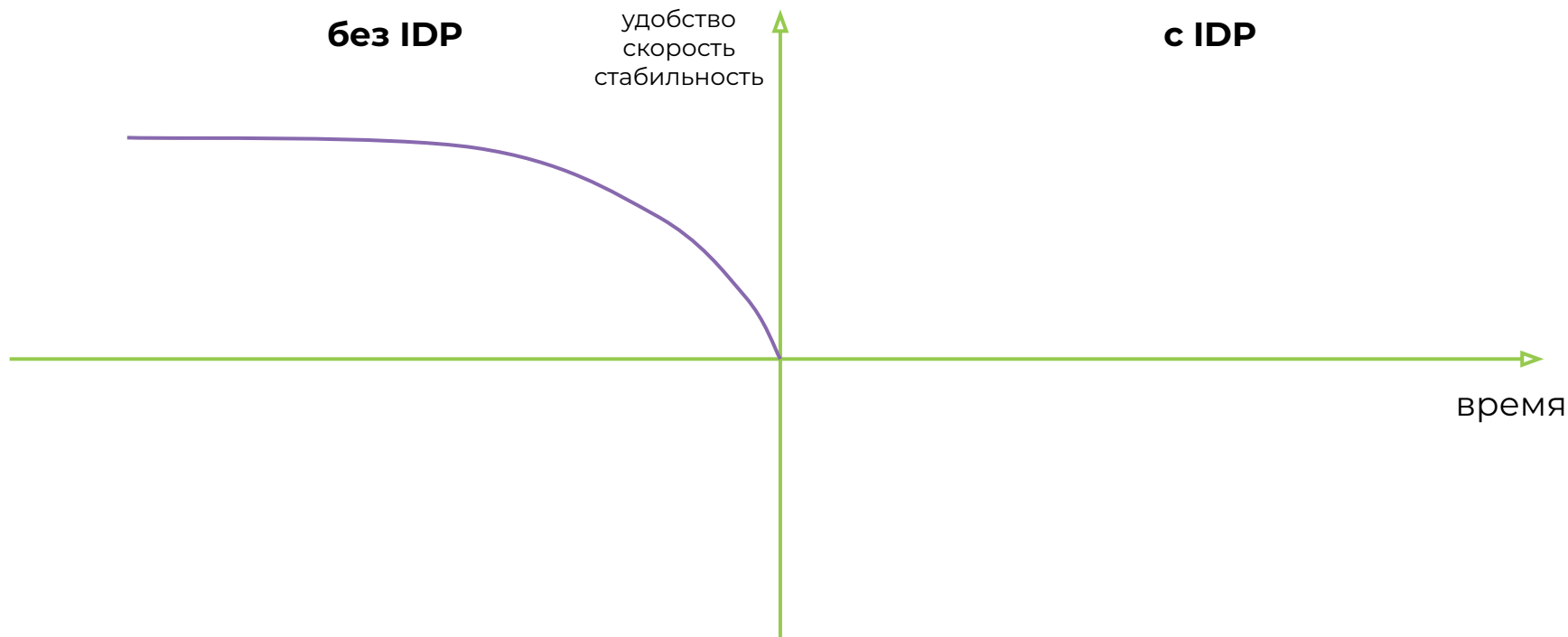
Возможный vendor
lock



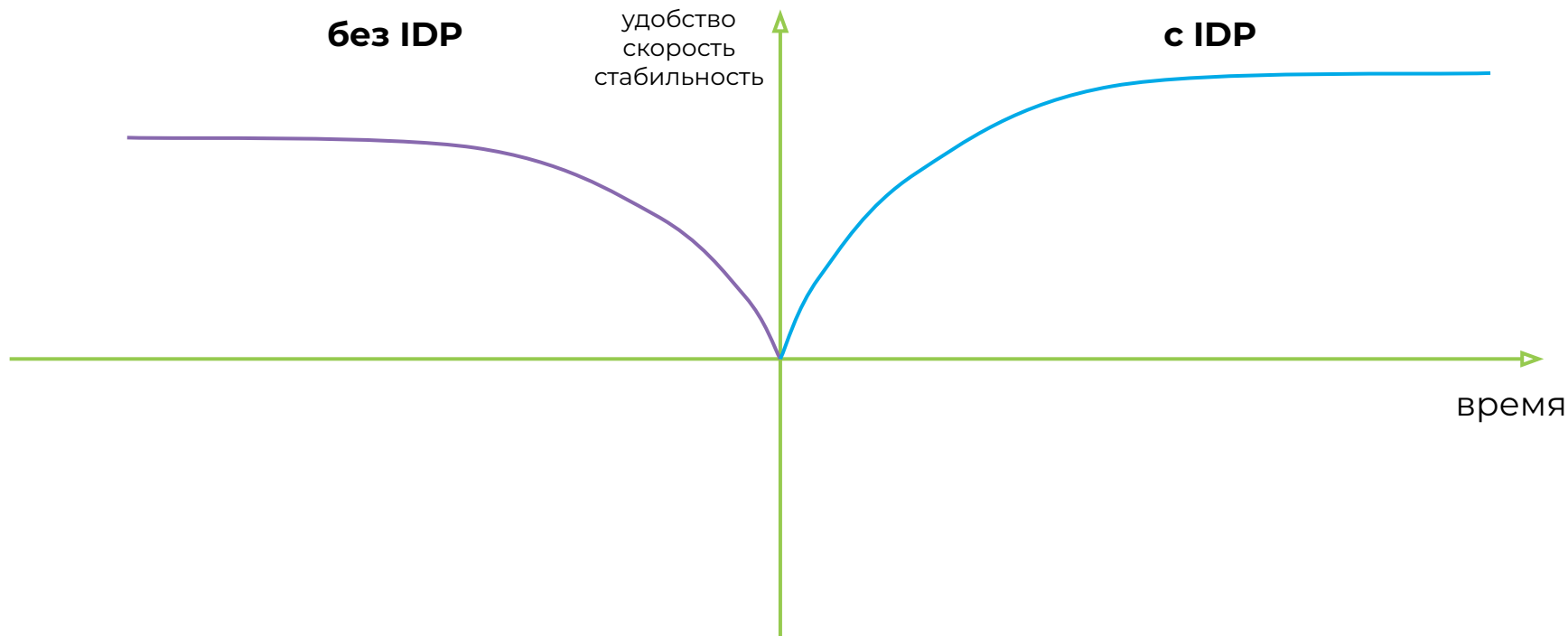
Миграция и
внедрение подходов

Фазы зрелости и внедрение

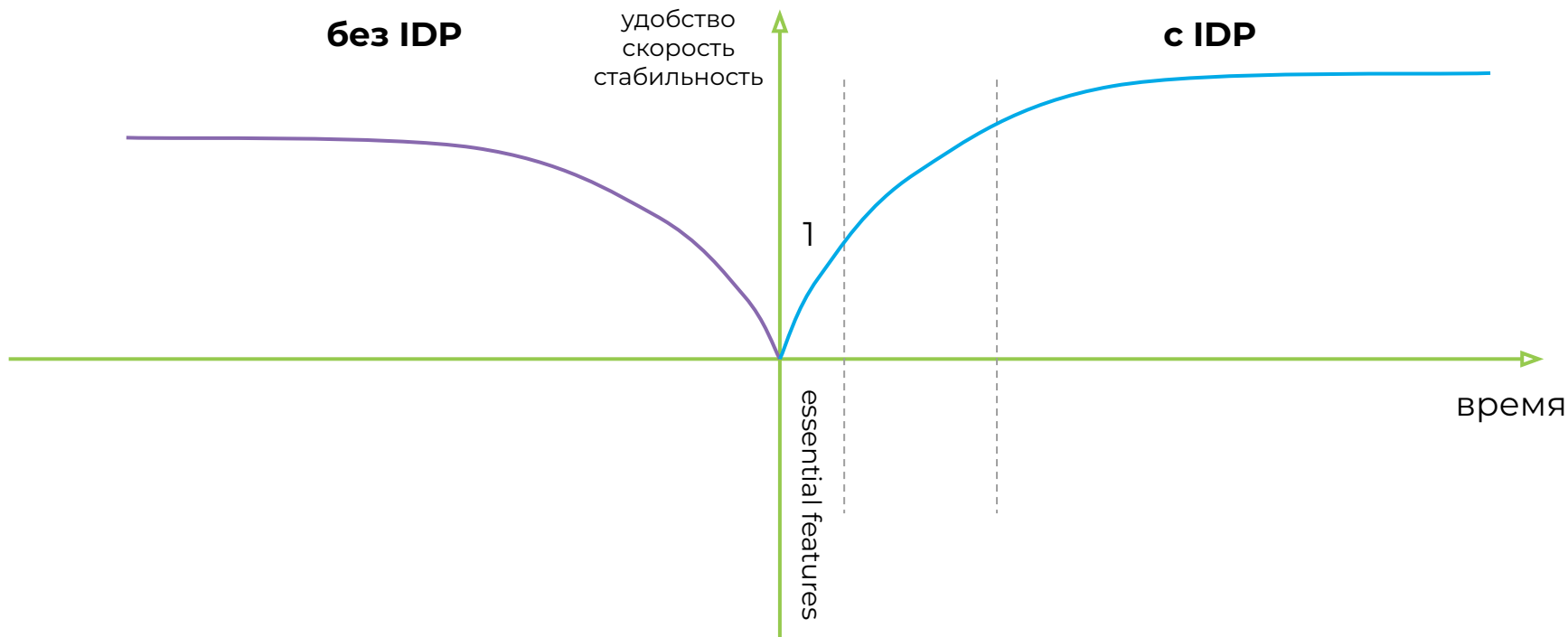
Фазы зрелости



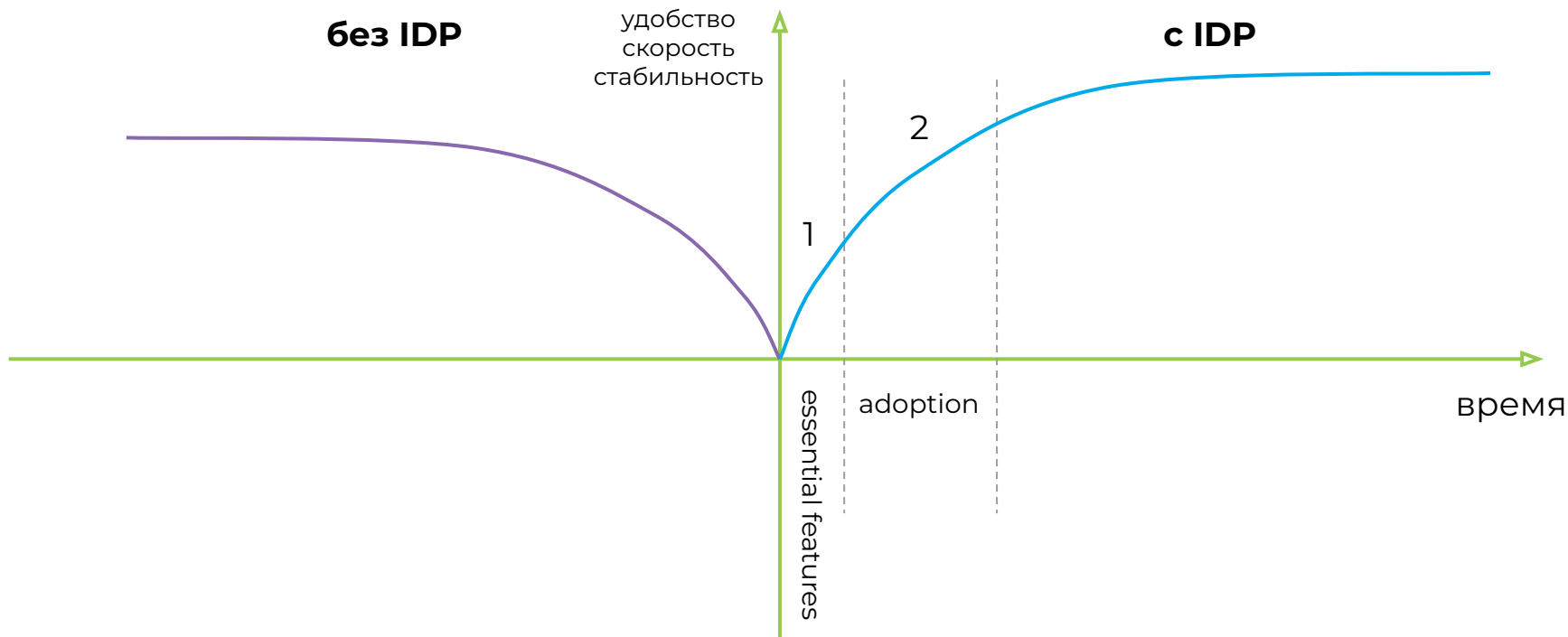
Фазы зрелости



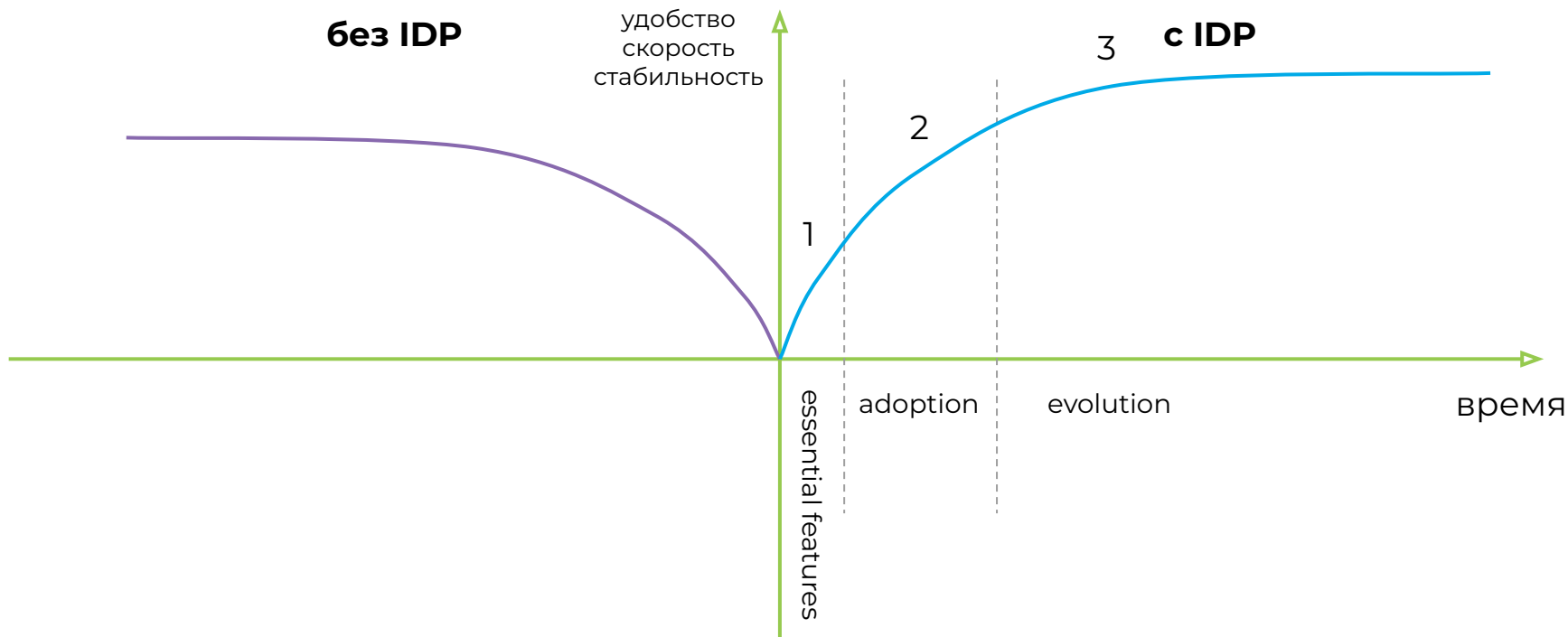
Фазы зрелости



Фазы зрелости



Фазы зрелости



Продуктовый ПОДХОД

Метрики и их сбор

Скорость



Onboarding time

Локальное окружение (развертывание, разработка)

Время на интеграцию с инфраструктурой

Время решения локальных проблем

Метрики и их сбор

Фокус на ценности



CJM coverage

Отношение ручных действий / автоматизации по направлениям

Необходимость обращения в platform/ops команду

Метрики и их сбор

Стабильность



SLI ключевых компонентов

Quality gates эффективность (red -> green после изменения)

MTTU / MTTR

Self healed cases / manual

Метрики и их сбор

Уверенность в будущем



Актуальность технического стека (версии языков, библиотек)
Adoption общих практик

Что важно пользователю продукта?

1. Удобство работы
2. Полнота функциональности
3. Минимум усилий для достижения целей
4. Оперативное решение проблем

Продуктовые процессы



Support



Единое окно
описания
проблем и
идей



Сбор метрик
удовлетворенности
и custdev



Discovery +
проработка
CJM

Support

1. Выделенная единая линия (дежурства по направлениям)
2. Сбор аналитики по зонам
3. Время ответа, удовлетворенность
4. База знаний

Requests механика



1. Сбор информации: описание проблемы, окружения, потенциального импакта, контактных лиц.
2. Обработка высокоуровневая.
3. Проработка потенциального решения и ответ.
4. Планирование и компоновка в Roadmap.

Discovery



1. Ценность (+ press release).
2. Stakeholders.
3. Как будем измерять impact.
4. Сложность реализации.
5. Артефакты на выходе (CJM, метрики, выводы из custdev).

Effort score / satisfaction



1. Выделяем ключевые зоны CJM.
2. На финальных шагах задаём контекстные вопросы:
 - a. Сложность операции?
 - b. Удовлетворенность?

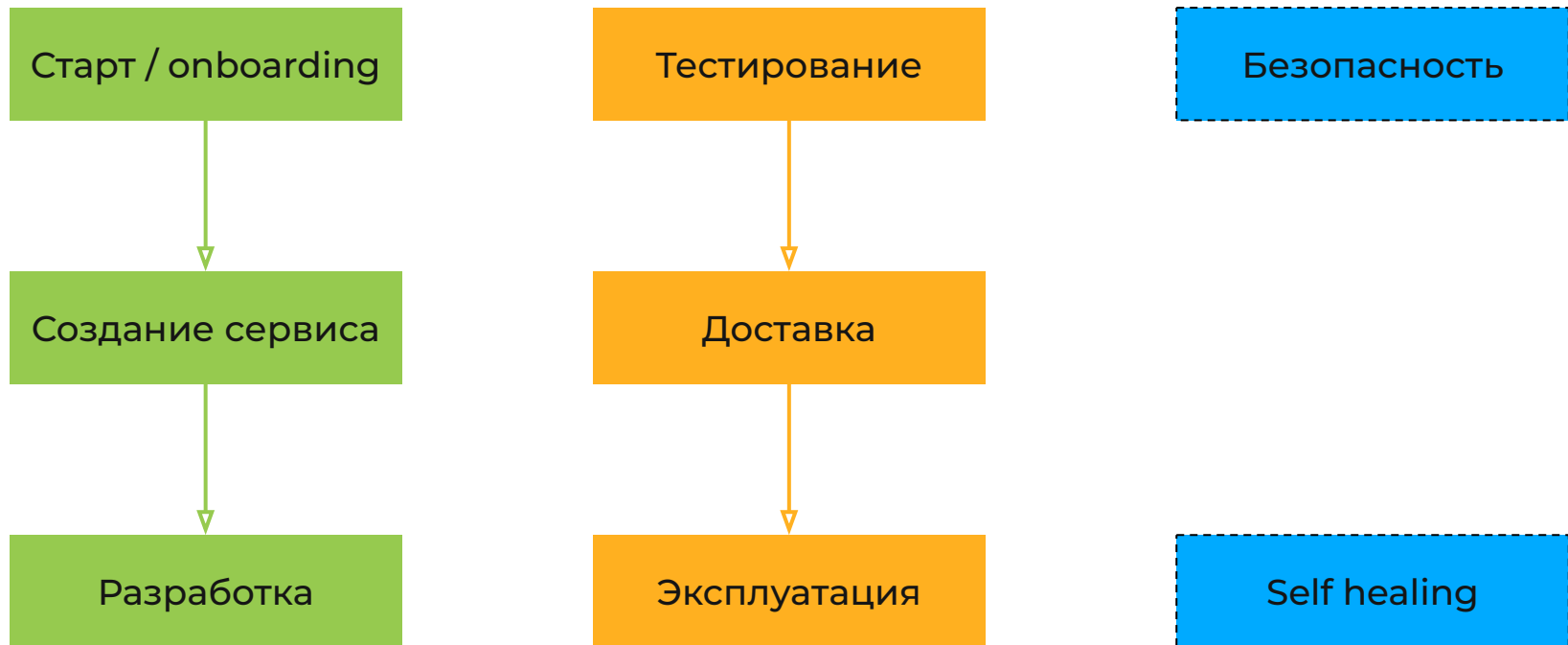
CJM



Платформа и её ВОЗМОЖНОСТИ

+ checklist!

Жизненный цикл разработки



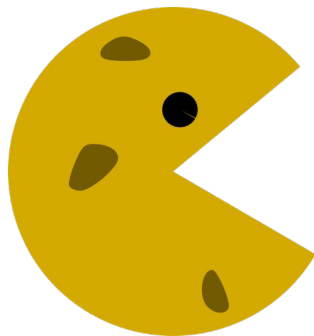
Старт и onboarding

Локальная среда

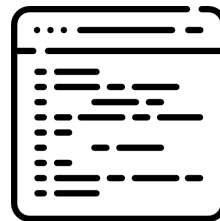
One button start



Готовые компоненты и boilerplate



Cookiecutter



Boilerplate

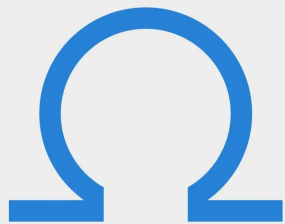
...



Monitoring

Разработка

Конфигурирование

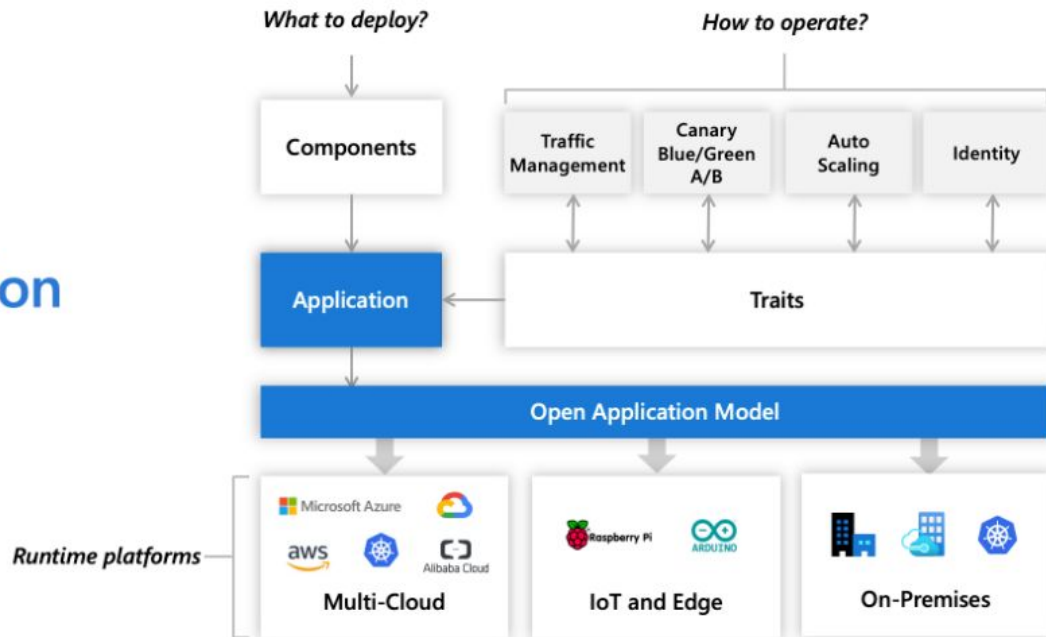


Open
Application
Model



Конфигурирование

Open Application Model

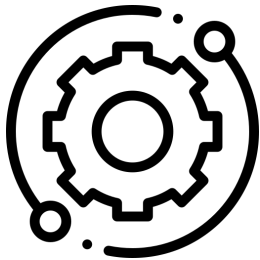


Конфигурирование

```
apiVersion: core.oam.dev/v1beta1
kind: Application
metadata:
  name: first-app
spec:
  components:
  - name: helloworld
    type: webservice
    properties:
    image: oamdev/helloworld-python:v1
    env:
      - name: "TARGET"
        value: "KubeVela"
    port: 8080
  traits:
  - type: ingress
    properties:
      domain: localhost
      http:
        /: 8080
```



Zero configuration init



```
sessionRedis := redis.New()
```



Env variables

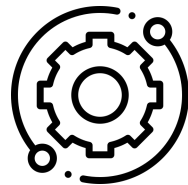
Кодогенерация и сервисное взаимодействие



Server



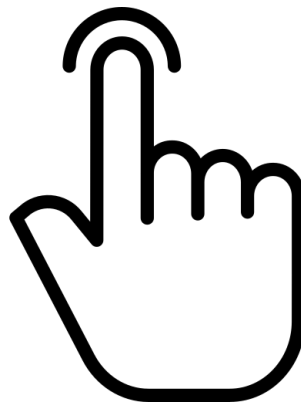
Client



Сгенерированный код
(circuit breakers, retries,
timeout propagation,
observability, zero conf)

ХaaS зависимости

1. Cron
2. Workers
3. Databases
4. Object storage
5. Queues, data-bus

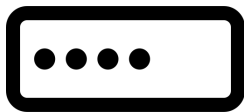


Поддержка актуальности тех стека



Безопасность

Менеджмент секретов



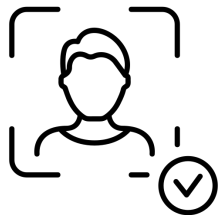
UI / CLI



Env/file
Injection



Межсервисное взаимодействие и безопасность



mTLS



Авторизация
сервисов

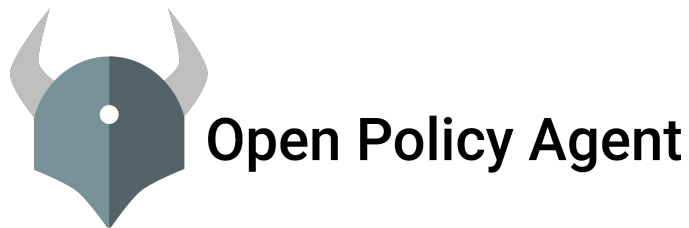


envoy



Istio

IDP авторизация



Role based access control

Security crawlers

1. OS / docker images
2. Libraries
3. Secrets leaks
4. Security linters

Тестирование

Фазы тестирования



Auto test platform



Линтинг, style guides

actions



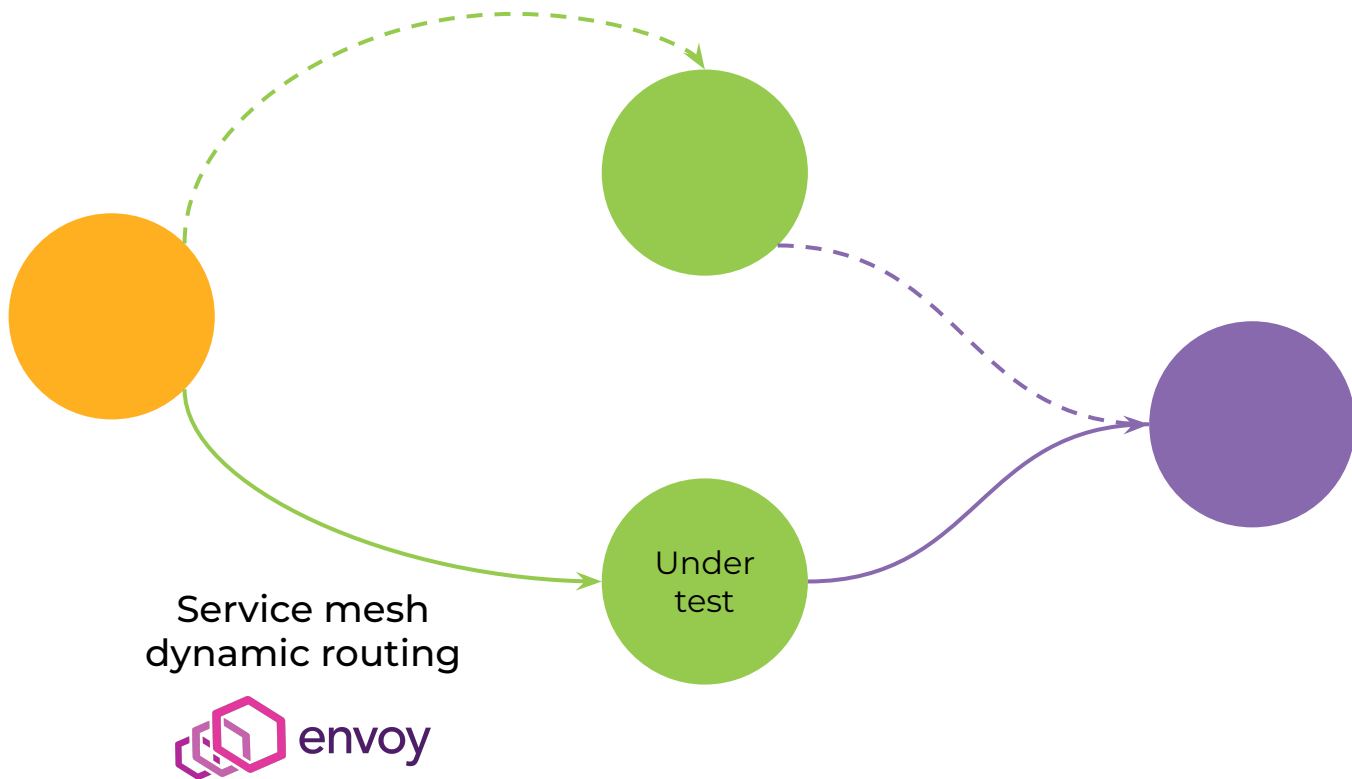
Unit / Integration tests
mutation



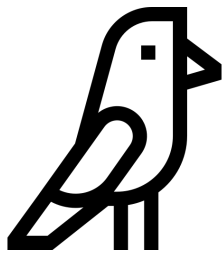
Контрактное
тестирование

РАСТ

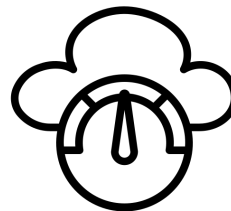
E2E



Shift right



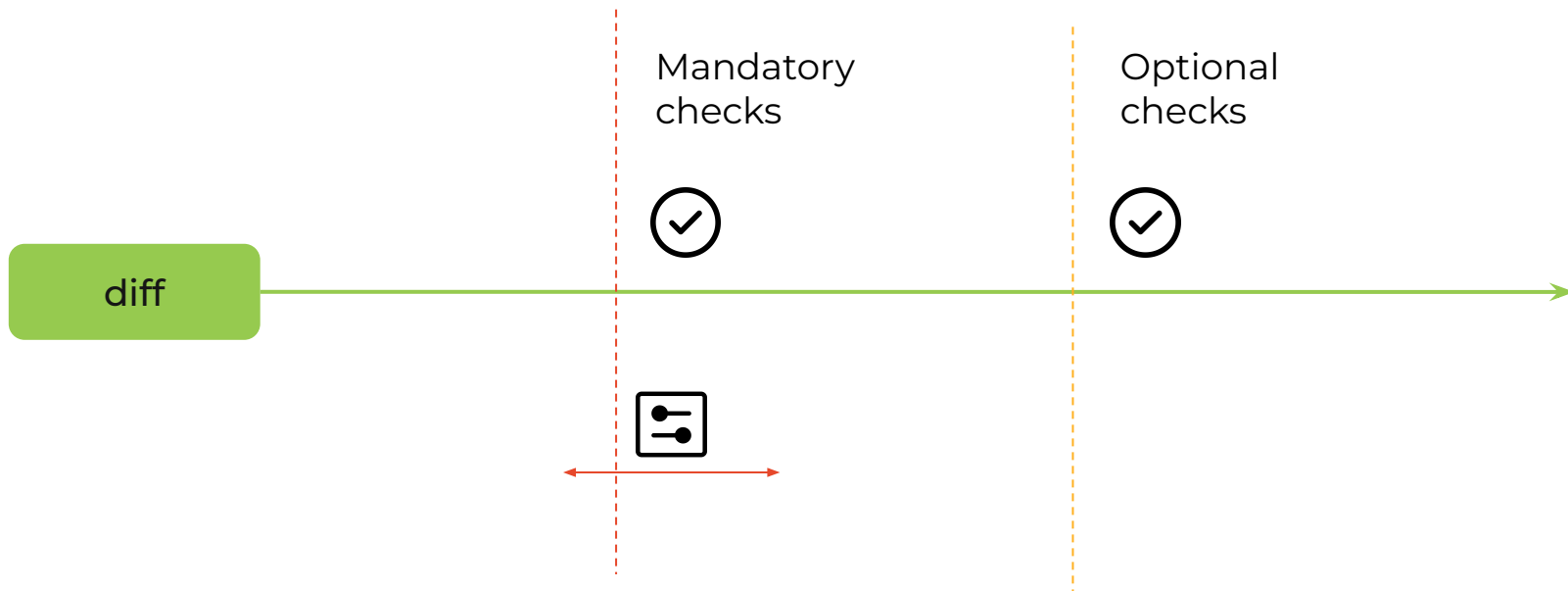
Canary



Chaos

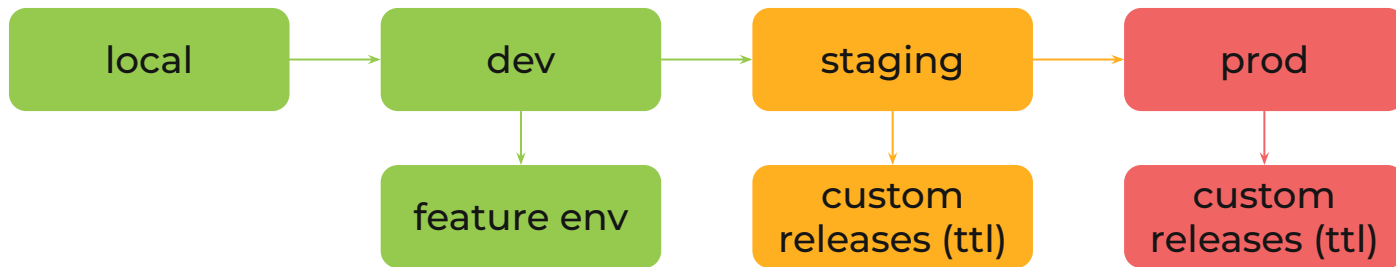


Контролируемые quality gates



Доставка

Environment management



Pipelines

1. Готовые унифицированные сборки
2. One button deploy
3. Поддержка различных стратегий: canary, blue/green, shadow.
4. Continuous deployment
5. Возможность расширения без ручного внедрения в базовые пайплайны
6. Переиспользование артефактов

Problem detector

1. Root cause analyzer
2. Детектор типичных проблем: OOM, throttling, падения контейнеров, проблемы шедулинга, деградации воркер нод и прочие.



Эксплуатация

Реестр сервисов и инцидент менеджмент

Home | Services | Backstage x +

demo.backstage.io/catalog

Good afternoon, Guest!
Backstage Service Catalog

SERVICES WEBSITES LIBRARIES DOCUMENTATION OTHER

Services

PERSONAL

- Owned 7
- Starred 0

BACKSTAGE

- All 10

Refine Results CLEAR

TAGS

- ☐ data
- ☐ go
- ☐ java
- ☐ playback
- ☐ website

Owned (7)

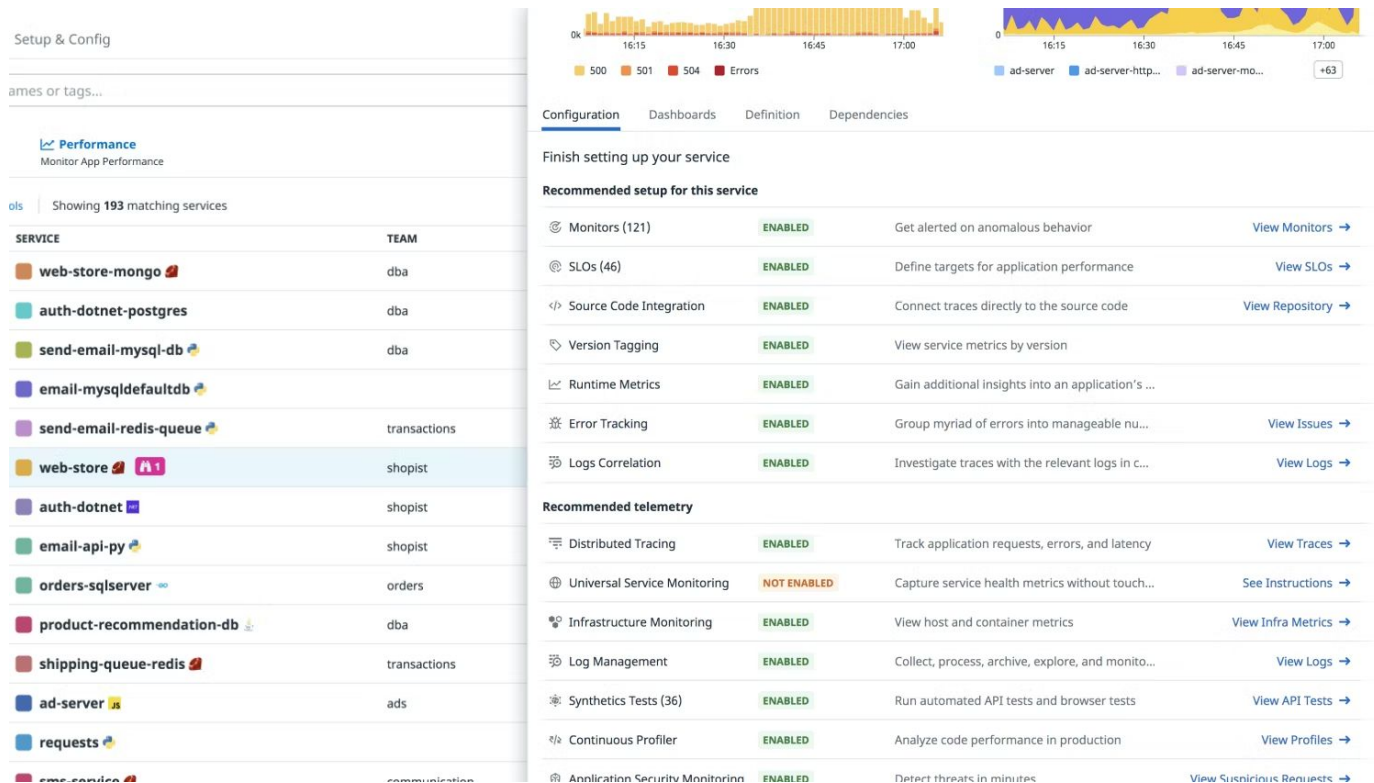
Search

NAME	SYSTEM	OWNER	LIFECYCLE	DESCRIPTION	TAGS	ACTIONS
artist-lookup	artist-engagement-portal	team-a	experimental	Artist Lookup	java, data	🔗 ✎ ★
playback-order	audio-playback	user:guest	production	Playback Order	java, playback	🔗 ✎ ★
searcher		user:guest	production	Searcher	go	🔗 ✎ ★
shuffle-api	audio-playback	user:guest	production	Shuffle API	go	🔗 ✎ ★
wayback-archive		team-a	production	Archive of the...		🔗 ✎ ★
wayback-archive-storage		team-a	production	Storage...		🔗 ✎ ★
wayback-search		team-a	production	Search of the...		🔗 ✎ ★



Backstage

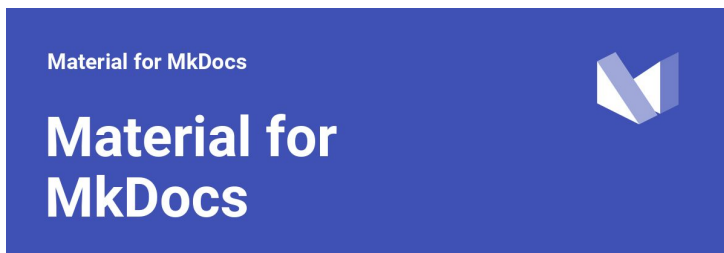
Реестр сервисов и инцидент менеджмент



Datadog
service catalog

Documentation as a service

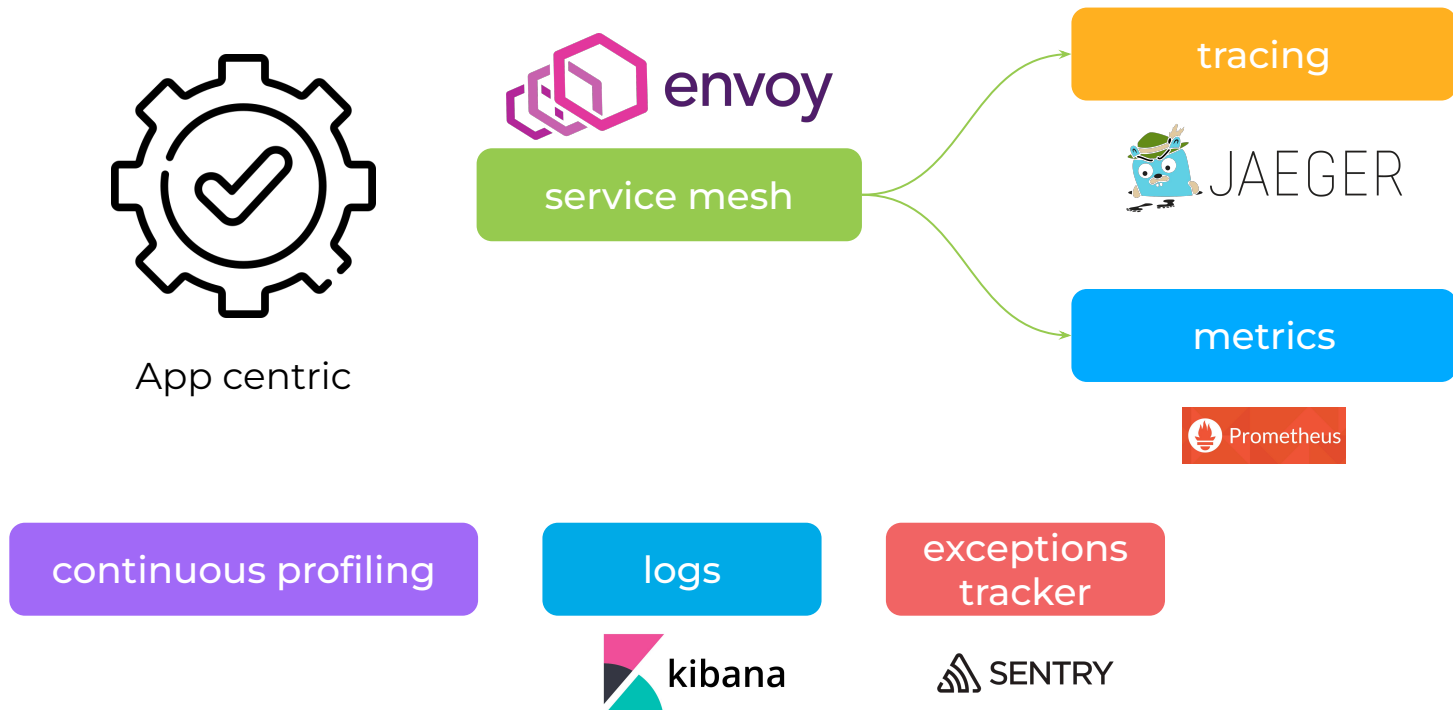
1. Документация в кодовой базе
2. Простая инициализация
3. Markdown
4. Code review для документации



Capacity management

1. Vertical / Horizontal Pod Autoscaler
2. Квотирование по орг структуре
3. Автоматический поиск ненужных сервисов, неэффективного использования железа

Унифицированный observability



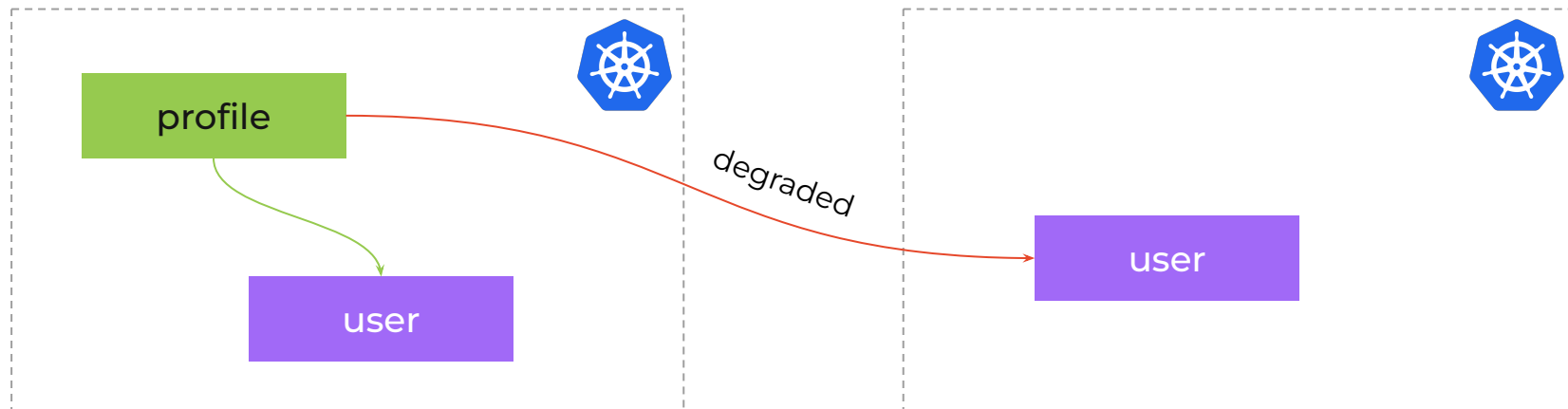
Non Functional Requirements

1. Описание требований по API: latency, нагрузка.
2. Автоматическое вычисление.
3. Отображение статистики, SLI.
4. Alerting на $SLI < SLO$.

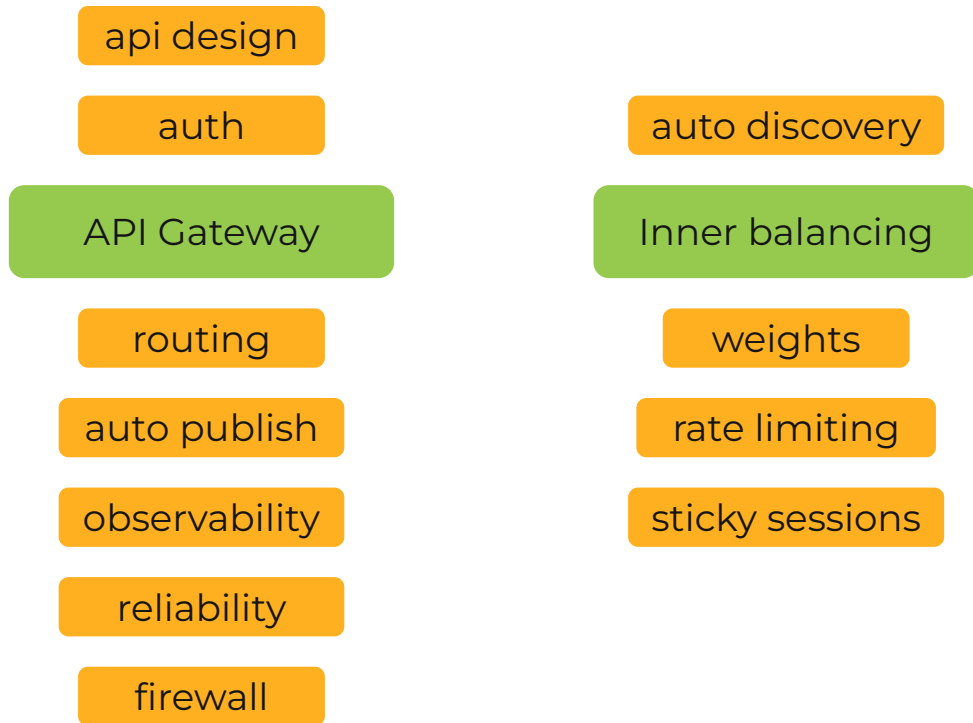
Self healing infrastructure (runtime)

1. Descheduling.
2. Network retries.
3. Outlier detection.
4. Unhealthy nodes detection.

MultiDC multi region поддержка



Балансировка и API gateway



PaaS / IDP Checklist



<https://forms.gle/JcKMENfgCuYYqRGg6>

Ваши вопросы!

avito.tech

Москва — 2022

Александр Лукьянченко

Руководитель PaaS



[@lookyan](https://www.instagram.com/lookyan)