



Как создать 1000 тестовых стендов и не выгореть

Константин Борисенко. ООО "Блекхаб геймс"

tg: @kost2191

О себе

Руководитель инфраструктуры в Blackhub games – компания разрабатывает мобильную онлайн-игру Black Russia с аудиторией более 8 млн игроков)

Feature Owner внедрения динамических стендов в 2-х компаниях (gamedev)

Опыт в IT: 16 лет

Руководство инфрой: 5 лет

Кандидат технических наук



BLACK
RUSSIA

О чем этот доклад

Не бросайтесь пилить автоматизацию поднятия тестовых стендов. Сгорите.

Ваша задача – решить проблему бизнеса:

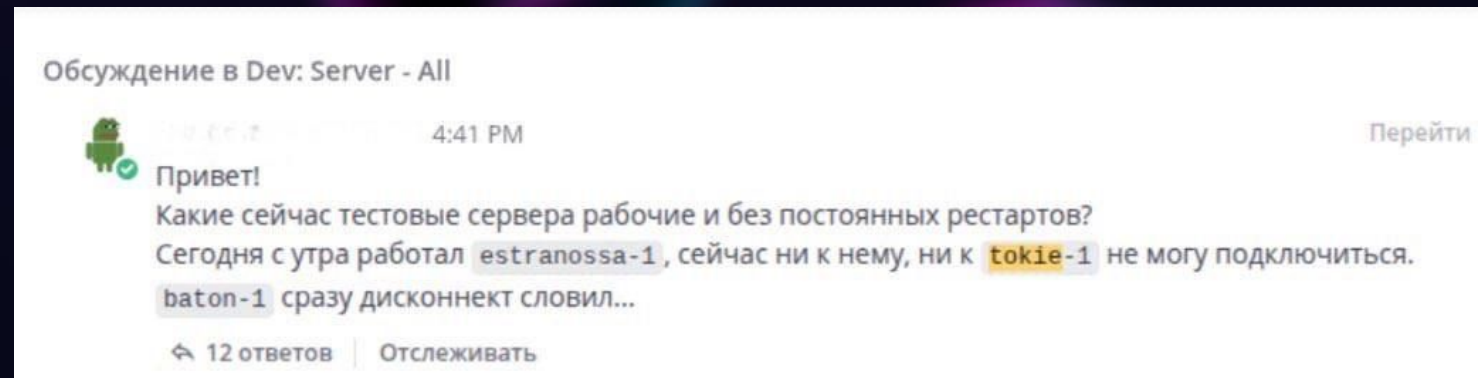
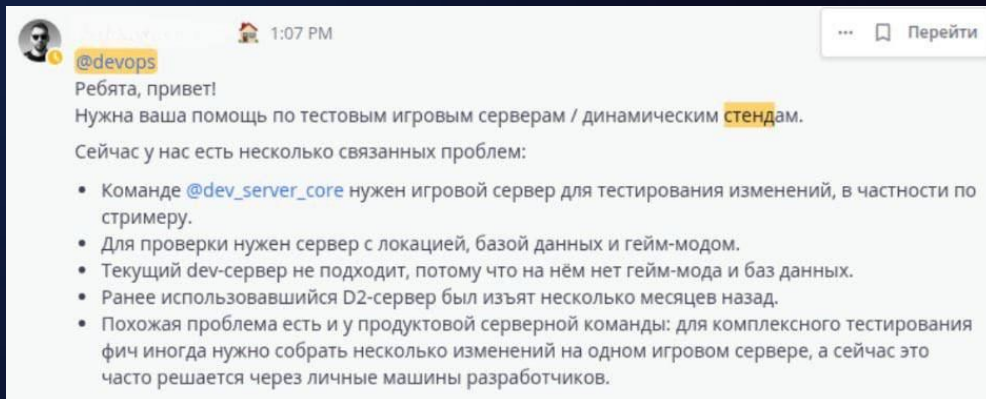
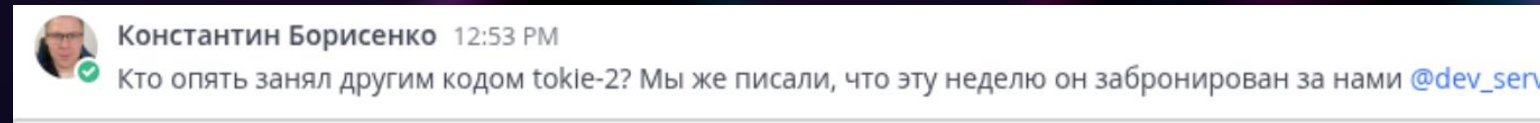
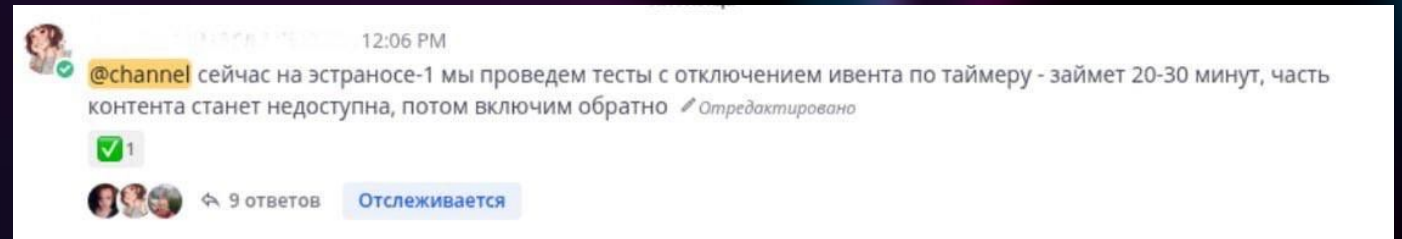
Долгий процесс от разработки до production

Правильно выстроив процесс работы со стендами, вы получите:

- Автоматизацию, не требующую вашего внимания
- Успех у коллег
- Поднимете скорость time to market!

Проблемы тестовых стендов

- Нехватка стендов
- Проблемы “бронирования”
- Сложности связи стенда и тестируемой фичи



Глобальные этапы внедрения динамических стендов:

1. Сбор кейсов от различных отделов по проблемам
2. Определение компонент, которые необходимо деплоить per окружение
3. Как изолированные компоненты одного стенда согласовать между собой
4. Инфраструктурные “изыскания” – разработка
5. Внедрение (несколько этапов)

Найдите команды, испытывающие проблемы

Кому может быть нужна работа с множеством стендов?

Роли:

- Разработчики
- QA
- Геймдизайнеры, переводчики
- Отдел продаж (демонстрации клиентам + Тестовые периоды)

Как понять, чем помочь коллегам?

Выявите сценарии, когда другие отделы тратят лишнее время из-за инфраструктуры.

Собирайте не просто проблемы, которые есть сейчас. Спрашивайте “как ты видишь идеальный процесс для себя”?

Кейсы необходимы для максимально широкого понимания требования к стендам

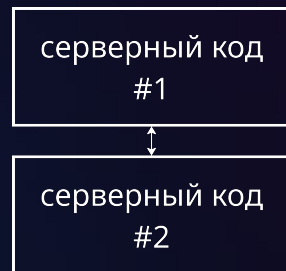
Это позволит учесть все нюансы и очертить границы изоляции стендов.

Определение компонентов для стенда

серверный код
#1

- Серверный код #1

Определение компонентов для стенда



- Серверный код #1
- Серверный код #2

Определение компонентов для стенда



- Серверный код #1
- Серверный код #2
- БД1, БД2

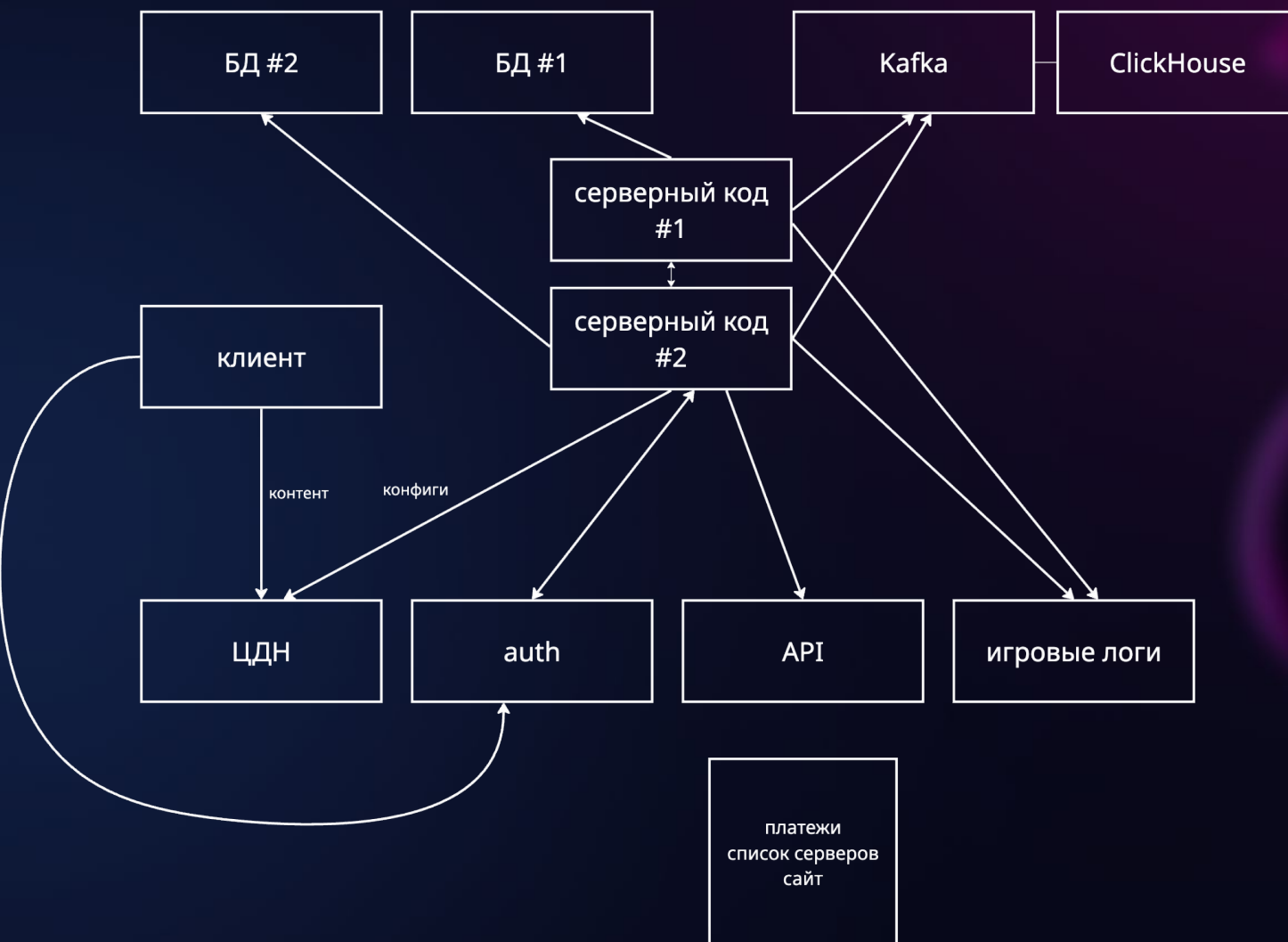
Определение компонентов для стенда



- Серверный код #1
- Серверный код #2
- БД1, БД2
- Клиентский код:
 - Сайт
 - Android
 - IOS
- Контент (3D. кнопки и тп) (Perforce)
- Конфиги (Баланс, цены и тп)
- ЦДН

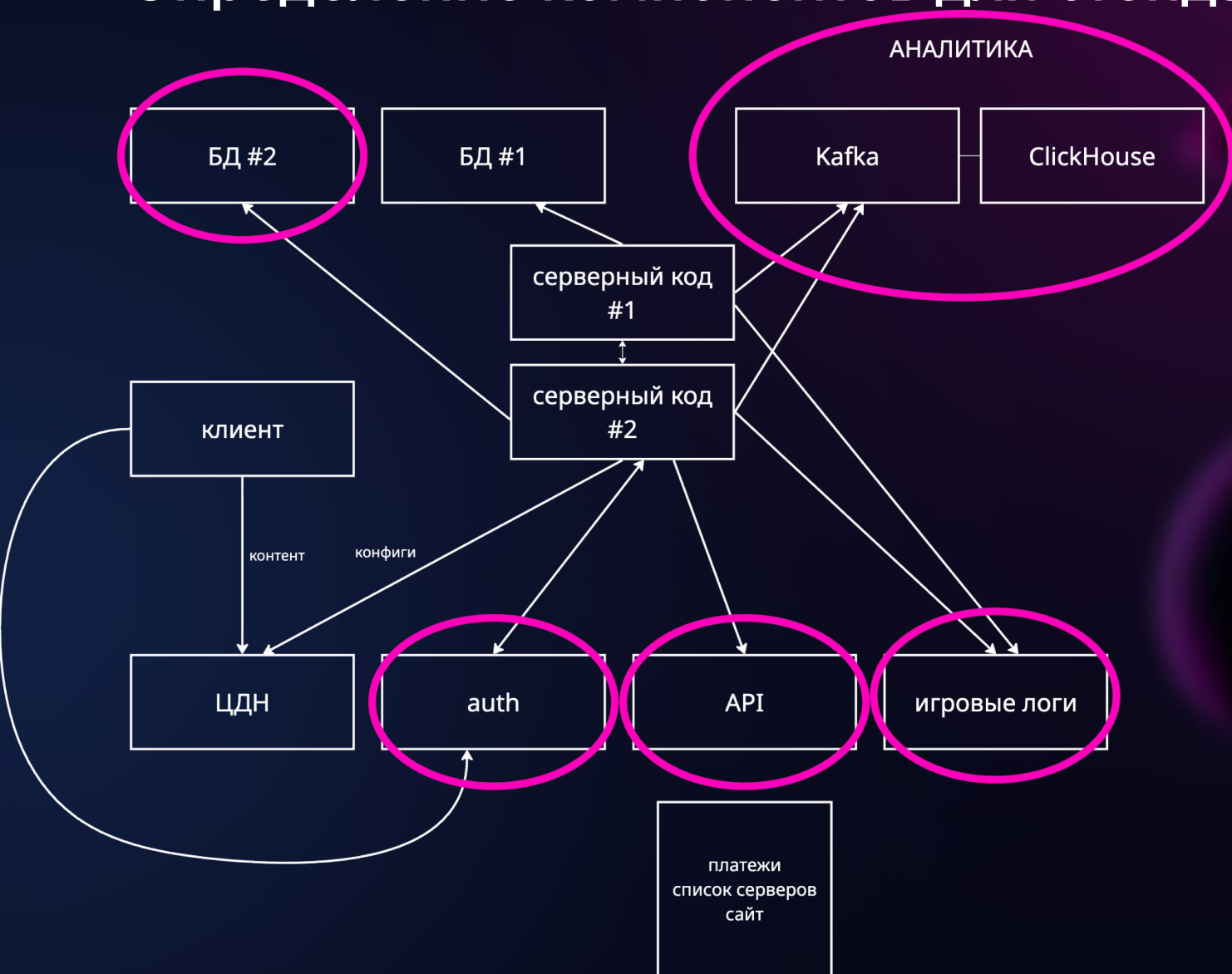
Определение компонентов для стенда

АНАЛИТИКА



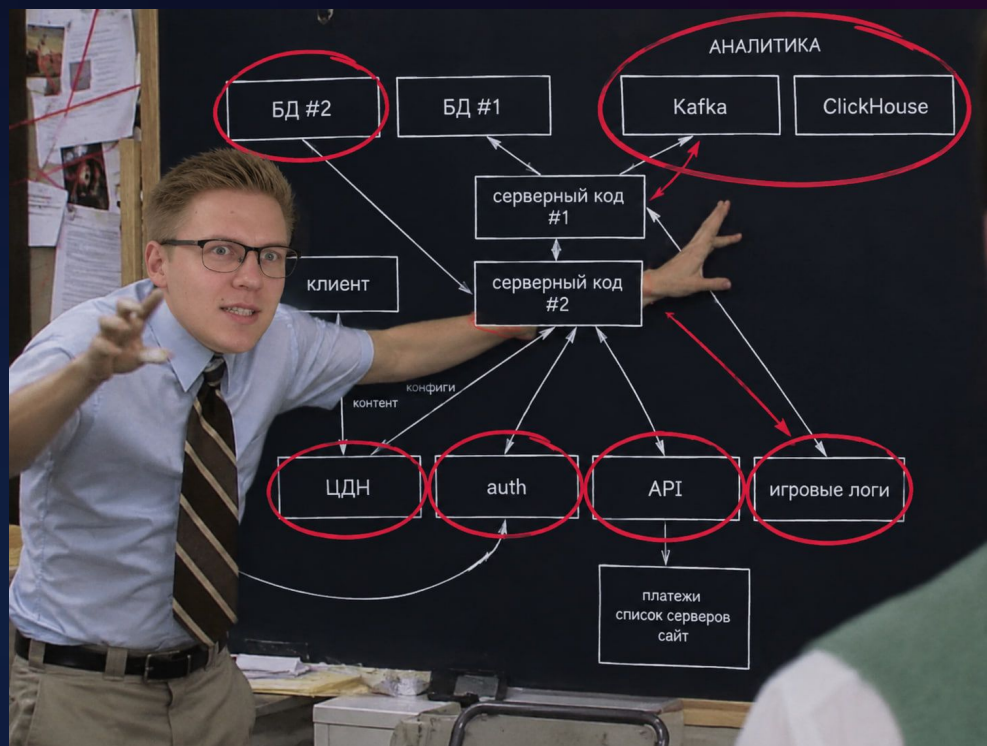
- Серверный код #1
- Серверный код #2
- БД1, БД2
- Клиентский код:
 - Сайт
 - Android
 - IOS
- Контент (3D. кнопки и тп) (Perforce)
- Конфиги (Баланс, цены и тп)
- ЦДН
- Аналитика/логи
- Внутренние сервисы

Определение компонентов для стенда



- Инстанс БД общий на все стенды
- Кафка общая
- Научите внутренние сервисы работать с множеством стендов

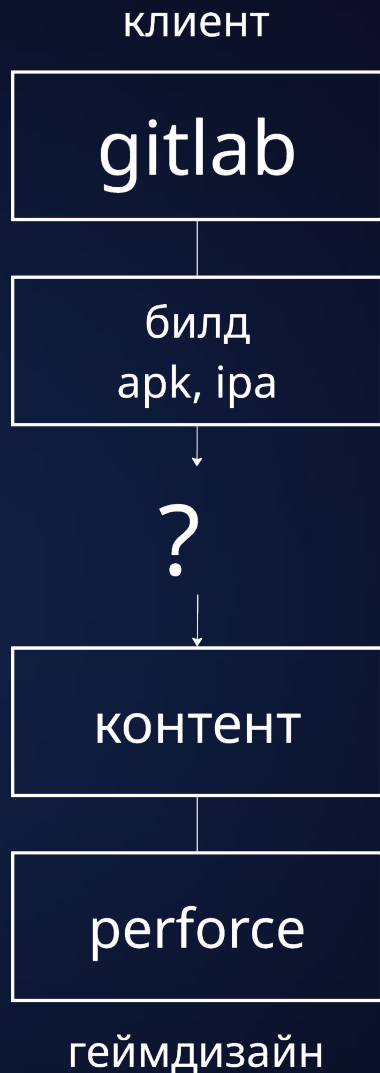
Как согласовать между собой компоненты одного стенда



Постарайтесь упростить выбор версий компонент стенда для конкретной задачи

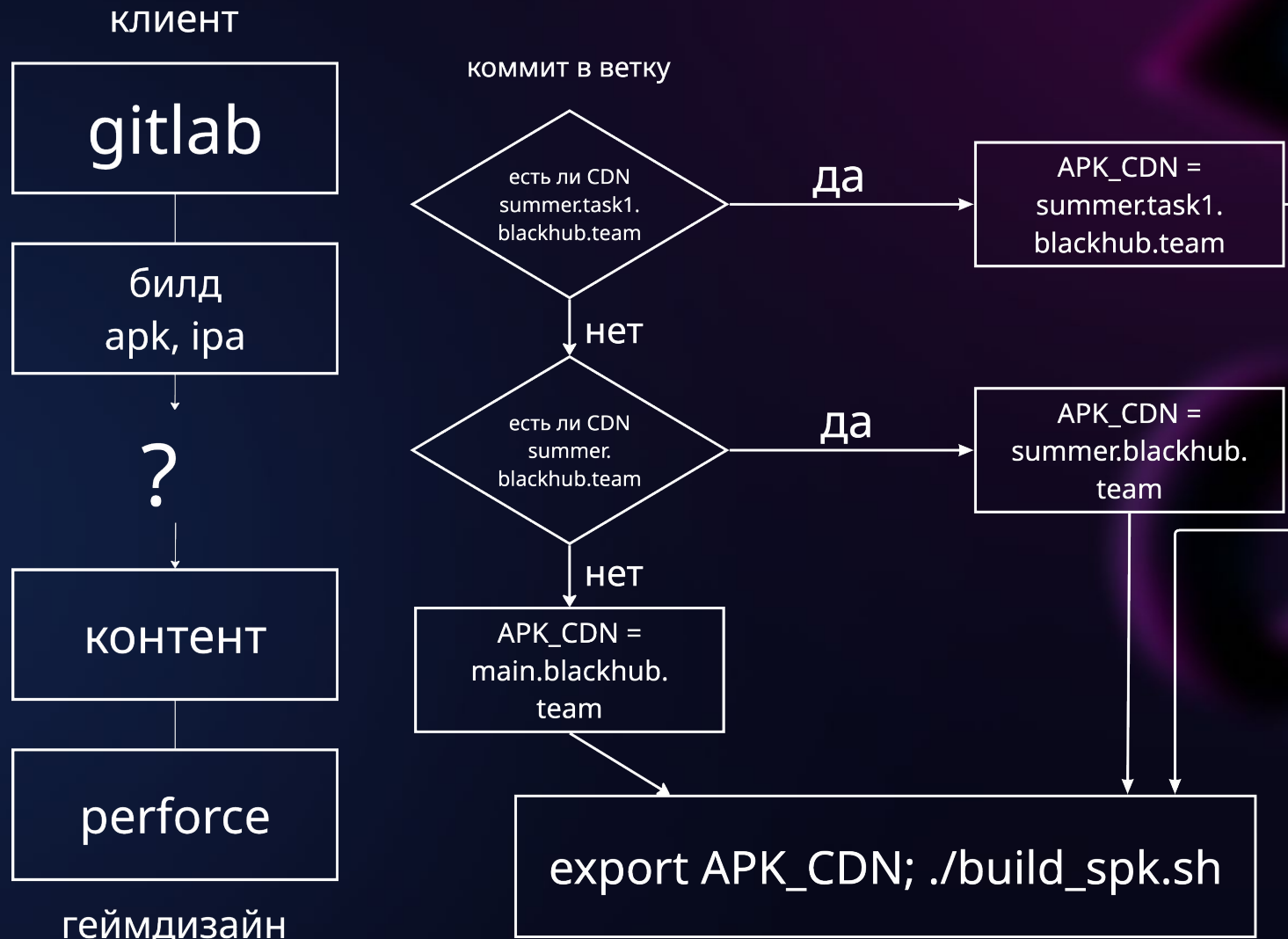


Связь между собой разных компонент



Представим, что фича
разрабатывается 2
командами:
GameDesign и Client

Связь между собой разных компонент



Сложности связи стенда и тестируемой фичи -

RESOLVED

Промежуточный итог

- Определили ключевые сложности -> в реализации учтем их
- Знаем границы изоляции будущих стендов
- Сняли нагрузку с создателей стендов по выбору версий компонент
- Смежные команды делают задачи для доработки компонент на работу с динамическими стендами

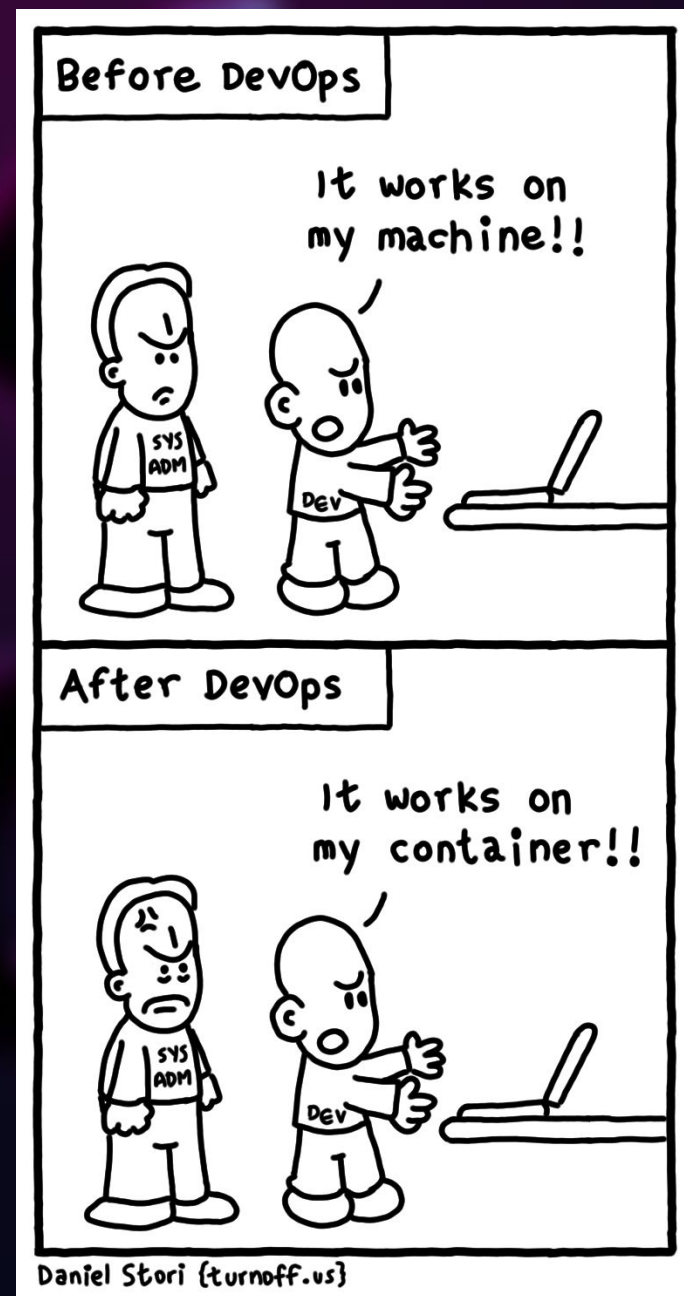
Го деплоить?



Без чего шансы на успех минимальны

- У вас нет CI/CD инфраструктуры
- Компоненты стенда не в докере
- Kubernetes
- Нет миграций БД у компонентов

Это блокеры к началу вашей
технической части работы над стендами



Даешь каждому желающему по стенду! Попытка 1

Реализация через terraform и ansible

Добавили возможность восстанавливаться из бэкапов базу данных с production

Минусы: очень дорого, долгое поднятие стенда

Плюсы: такие стенды должны быть - stage + perf

Даешь каждому желающему по стенду! Попытка 2

Реализация через **docker compose**

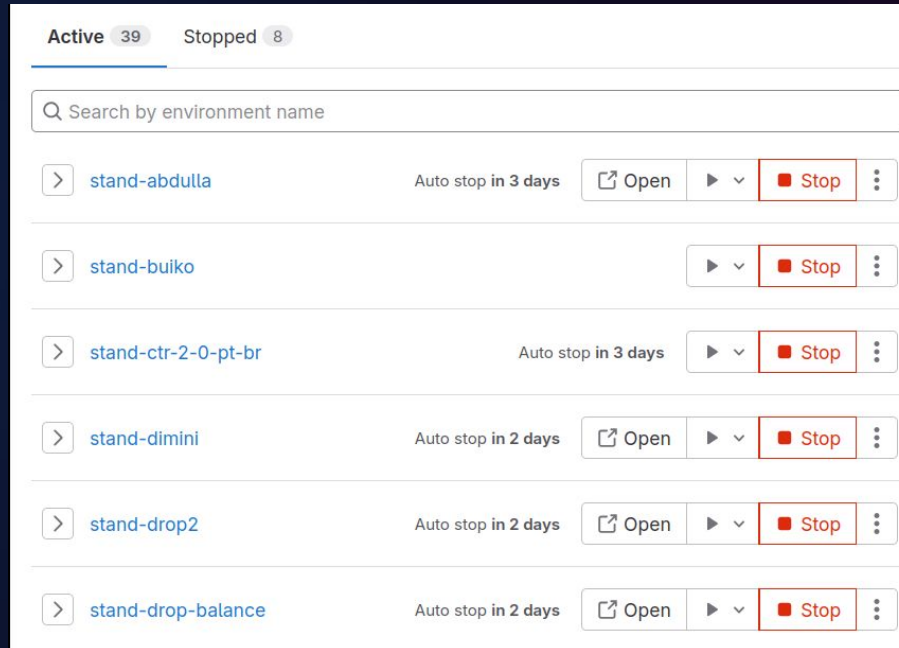
Через vhosts и CD генерацию docker-compose с дальнейшим запуском

Минусы:

1. Залили стенд, который съедал весь ЦПУ- все стенды начали тормозить
2. 250+ контейнеров начинали ломать весь сетеп - система не справлялась
3. Пользоваться адекватно этим могли только серверные разработчики

Даешь каждому желающему по стенду! Попытка 3

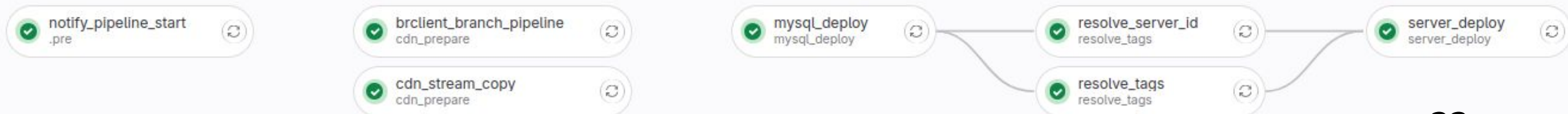
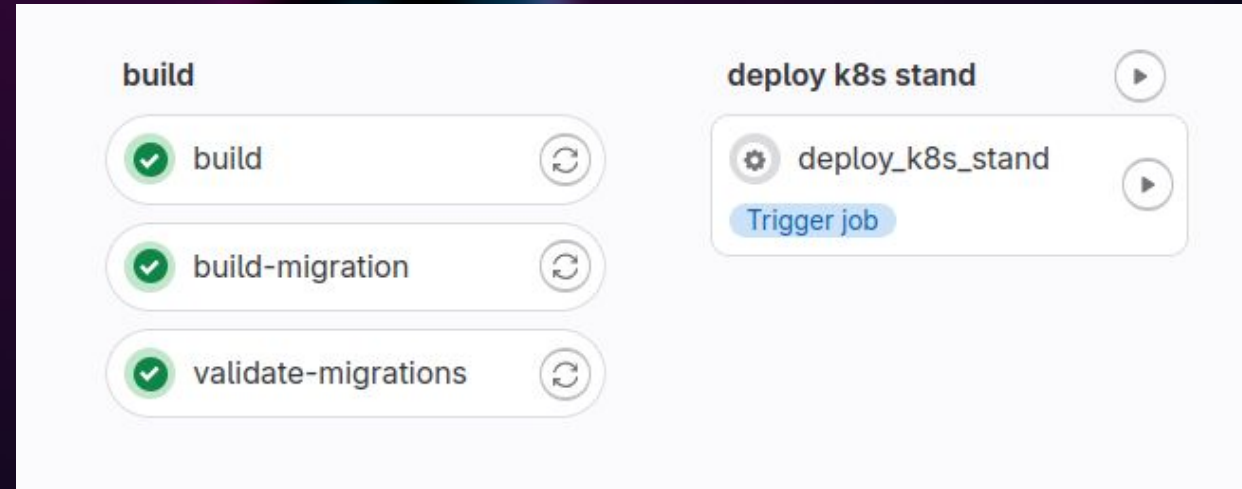
- K8S - инфра для стендов
- TTL стенда через Gitlab Environments
- Связь стенда с веткой через CI/CD



postgres-operator	Active	2d19h
stand-abdulla	Active	58d
stand-ctr-2-0-pt-br	Active	43h
stand-dimini	Active	2d17h
stand-drop-balance	Active	2d20h
stand-drop2	Active	2d19h
stand-epic-ctr-2-0-br-27529	Active	2d19h
stand-epic-ctr-2-0-main	Active	6d
stand-epic-ctr-2-0-main-pt-br	Active	46h
stand-epic-donate2-br-31096	Active	2d19h
stand-epic-halloween-base	Active	4d19h
stand-epic-halloween-crane	Active	10d
stand-epic-halloween-main	Active	4d23h
stand-epic-halloween-racemode	Active	3d19h
stand-epic-master-br-31694	Active	3d
stand-epic-new-actor-actions-main	Active	3d23h
stand-epic-newshub2-br-30622	Active	9d
stand-epic-sdl-main	Active	3d18h
stand-epic-story5-main	Active	12d
stand-epic-support-update-main	Active	4d17h
stand-feature-professions-main	Active	42h

Процесс деплоя стенда

1. Уведомление о старте
2. Сборка клиента при необходимости
3. Копирование cdn-стрима для стенда
4. Деплой mysql
5. Решение зависимостей других компонент, Выбор свободного ID сервера
6. Деплой сервера
7. Уведомление об успешном деплое/фейле



Уведомление в корп чат о готовности стенда



gitlab informer БОТ 3:44 PM

Stand deployment *success*

[pipeline: 108034](#)

Stand deployment *success*

Deployment information

server name: regress-rabot-v2

server core: main

brgamemode: feature-professions-main

server jsons: <https://professions-k8s-regress-rabot-v2.dev.blackhub.team/server-jsons.zip>

Jenkins build: <https://ci.blackhub.team/job/gameserver-stand-deploy/895/>

by @Александр Кипятков

Связь между собой разных компонент

Важно: иметь место, где отображаются все версии используемых компонент, чтобы сотрудники могли сверить часы

Type to search					Фильтры
☐ Сервер	Публикация	Состояние	Версия	Активность	
☐ stand-epic-sdl-main #300 Динстенд	Неопубликован	● Онлайн 300 игроков	br-svr · Build 99524 epic/sdl/main · 2d6f1d4a ↗ brgamemode · Build 107963 epic/sdl/main · d4274ed9 ↗	2026-09-20 11:51:25 6 ч. 49 мин. 43 сек.	

☐ stand-epic-halloween-base #315 Динстенд	Неопубликован	● Онлайн 0 игроков	br-svr · Build 108065 main · 651751ef ↗ brgamemode · Build 107348 epic/halloween/base · 61a418cc ↗	2026-09 6 ч. 58 ми	
---	---------------	-----------------------	---	-----------------------	--

Что еще упростит работу со стендами?

Пресеты конфигураций БД. Например:

- a. Заранее созданные аккаунты персональные с уровнями
- b. Включенная админка на аккаунты
- c. Любые настройки конкретного аккаунта, которые упростят жизнь QA и другим (запросите у команд и отдайте на реализацию соответствующим отделам)

Code owners Assign users and groups as approvers for specific file changes. [Learn more.](#)

master > brgamemode / database / main / sql / accounts.sql Find file Blame

accounts.sql 78.00 KiB Edit Lock Replace

```
1 INSERT INTO `accounts` (`id`, `name`, `password`, `rehashed`, `next_change_pass`, `email`, `confirm_email`, `le
2 (1, 'Max_Kalash', '7e2a009ada60dbfc7303ab71be1527dc', 0, 1681884911, 'None', 0, 10, 17, 0, 0, 161, 1200200, 0,
3 (2, 'Black_Russia', 'b77d18d1fd0dafe2bbe6e28a136839c2', 0, 1680041949, 'None', 0, 10, 1, 0, 0, 161, 7806000, 0,
4 (3, 'Egor_Sharov', 'c702e3531e0046ed6cac10ae01b30230', 1, 1678968878, 'None', 0, 20, 8, 0, 0, 6894, 7592555, 36
5 (4, 'Grand_Persona', '6272d353f1e990c19bfe25caad9c31e7', 1, 1691409264, 'None', 0, 10, 17, 0, 0, 2, 800793, 0,
6 (5, 'Persona_Persona', '0bd370f322379355a19516ce9813ee4e', 1, 1680188915, 'None', 0, 10, 17, 0, 0, 161, 7700000
7 (6, 'Sander_Kligan', 'eca3d48f51f8fe04f44f953b2d2312cc', 1, 1680691849, 'None', 0, 13, 30, 3, 0, 78, 28126263,
8 (7, 'Luisa_Ossi', '71b69e1930d9eca456df3e27732a3972', 1, 1680179628, 'None', 0, 10, 8, 0, 1, 5365, 50158500, 17
9 (8, 'Lev_Kalashnikov', 'ff03ba458a11fd51a9ca9cbecaa2ef72', 1, 1679936195, 'None', 0, 10, 9, 0, 0, 161, 65599381
10 (9, 'Youtube_Manager', '3b48711ad604e5be80ae1d6205a5155b', 0, 0, 'None', 0, 10, 3, 0, 0, 71, 100000, 0, 1, 0, 0,
```

Глобальные этапы внедрения динамических стендов в команды

На первый взгляд поверить трудно - Вы можете столкнуться с противниками стендов + с теми, кто не будет начинать ими пользоваться.

- Постоянный сбор обратной связи по упрощению процессов работы и освещение улучшений в общих каналах
- Реклама работы со стендами
 - Первые преемники - проповедники новых возможностей внутри своих команд и между команд
- Реклама не стендов, но решаемых кейсов с их помощью. Демонстрация успешных кейсов коллег

Погружение команд в работу с динамическими стендами

Разным командам необходима разная степень детализации работы со стендами:

- Серверный отдел
- QA
- Геймдизайнеры
- Клиентские разработчики

Разные степени погружения в работу динамических стендов

Стоимость

1. Анализ объемов RAM и CPU для компонент
 - Спойлер: выставяйте CPU request не больше 100m
2. Прикинуть размеры ноды, чтобы на нее помещалось эффективное количество подов
3. Не забываем учитывать поды мониторинга, логов и тп.

8 тысяч рублей ~ 5 стендов + prometheus, логгирование, nginx, k8s managed

Сейчас 40 стендов = 120к // - ~3к в месяц цена стенда.

Чем больше стендов, тем цена за стенд ниже

Итоги:

- Не создавайте абстрактные тестовые стенды, поймите своего потребителя и его потребности
- Нехватка стендов -> нельзя просто так взять и создать 1000 тестовых стендах не модифицируя работу ваших систем
- Для того чтобы минимизировать конфликты, сделайте надежный и простой механизм бронирования
- Сложности связи стенда и тестируемой фичи -> Предложенный ранее процесс автоматизации выбора версий компонент упрощает для сотрудника создание стенда

Как создать 1000 тестовых стендов и не выгореть

Константин Борисенко. ООО "Блекхаб геймс"
tg: @kost2191