

Полный цикл тестирования микросервисов в Docker контейнерах

Что будет?

- Запустим микросервисы с использованием Docker
- Протестируем с помощью Gradle и WireMock
- Настроим Jenkins Pipeline на Docker'ах
- Соберем отчеты в Report Portal и Allure Report

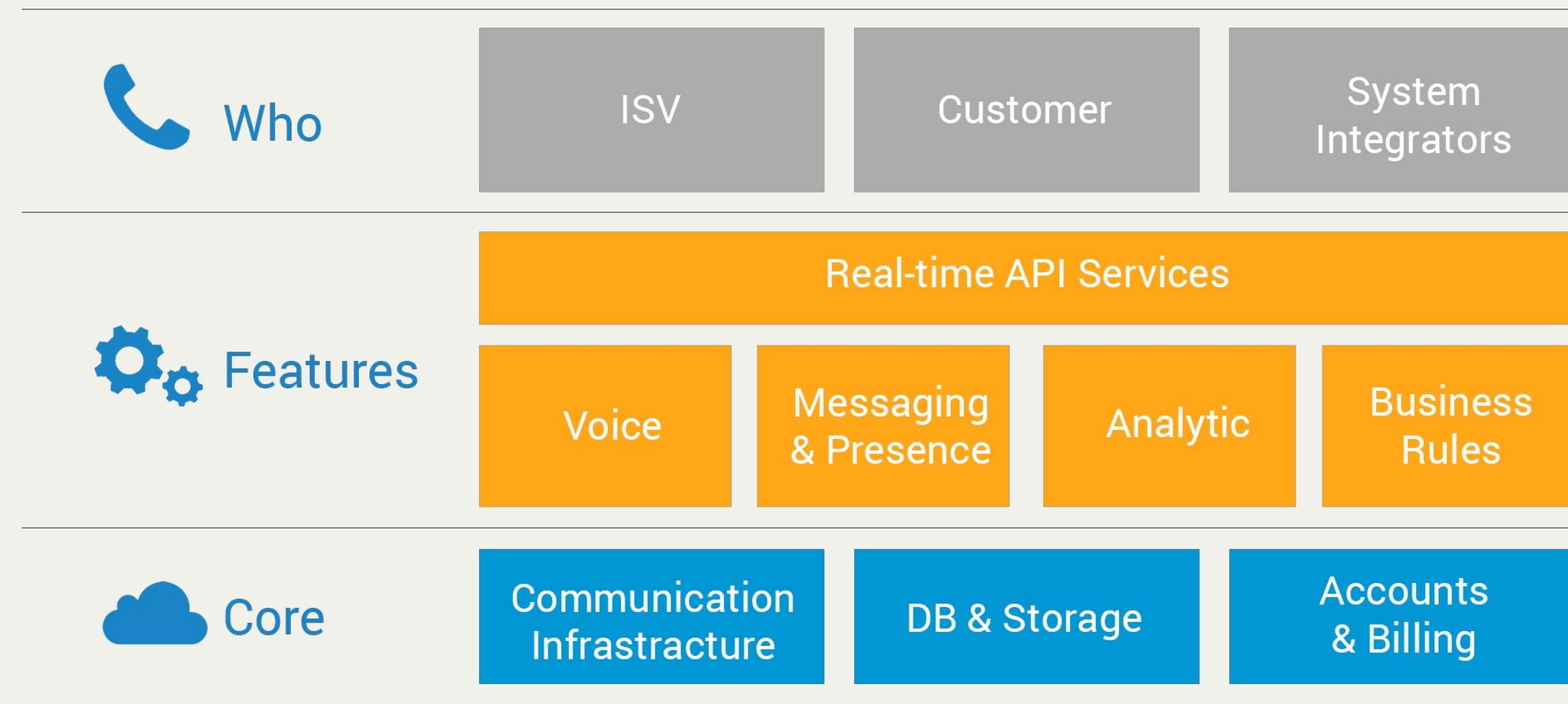
github.com/dins-heisenbug-2016

Сервис

обслуживает большую часть API



API Platform



Статистика



сервис
237,000 SLOC

тесты
229,000 SLOC

Статистика

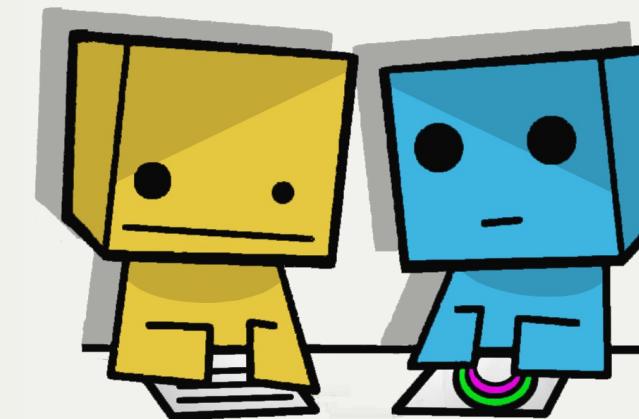
1 500 страниц



4 800 страниц



6 200 страниц

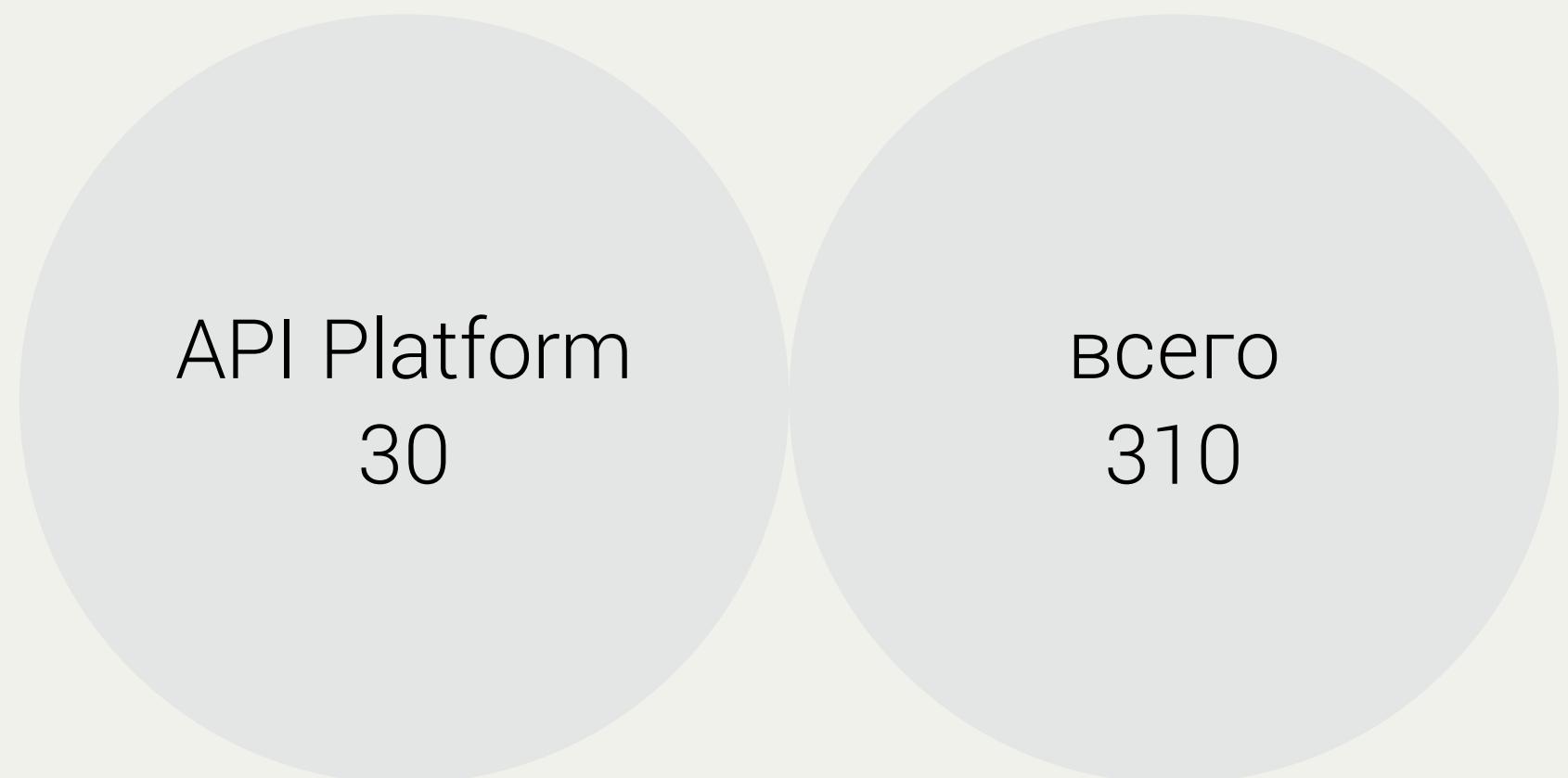


наши тест-кейсы

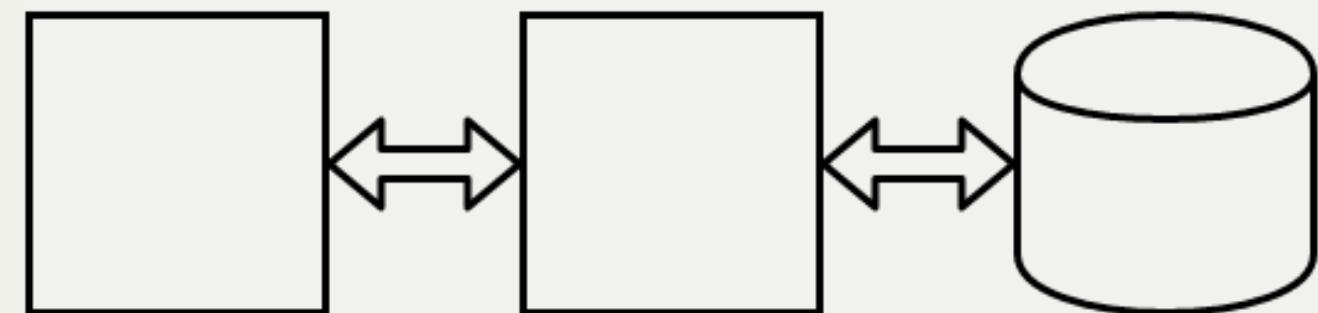
Система облачной коммуникации



Статистика компонентов



Окружение нормального человека



Тестовые окружения

11 окружений

70+ VM

120 vCPU

160GB RAM

Процесс релиза

7 команд и 1 репозиторий



Процесс релиза

> merge
полная
регрессия

1 день
тратит
команда

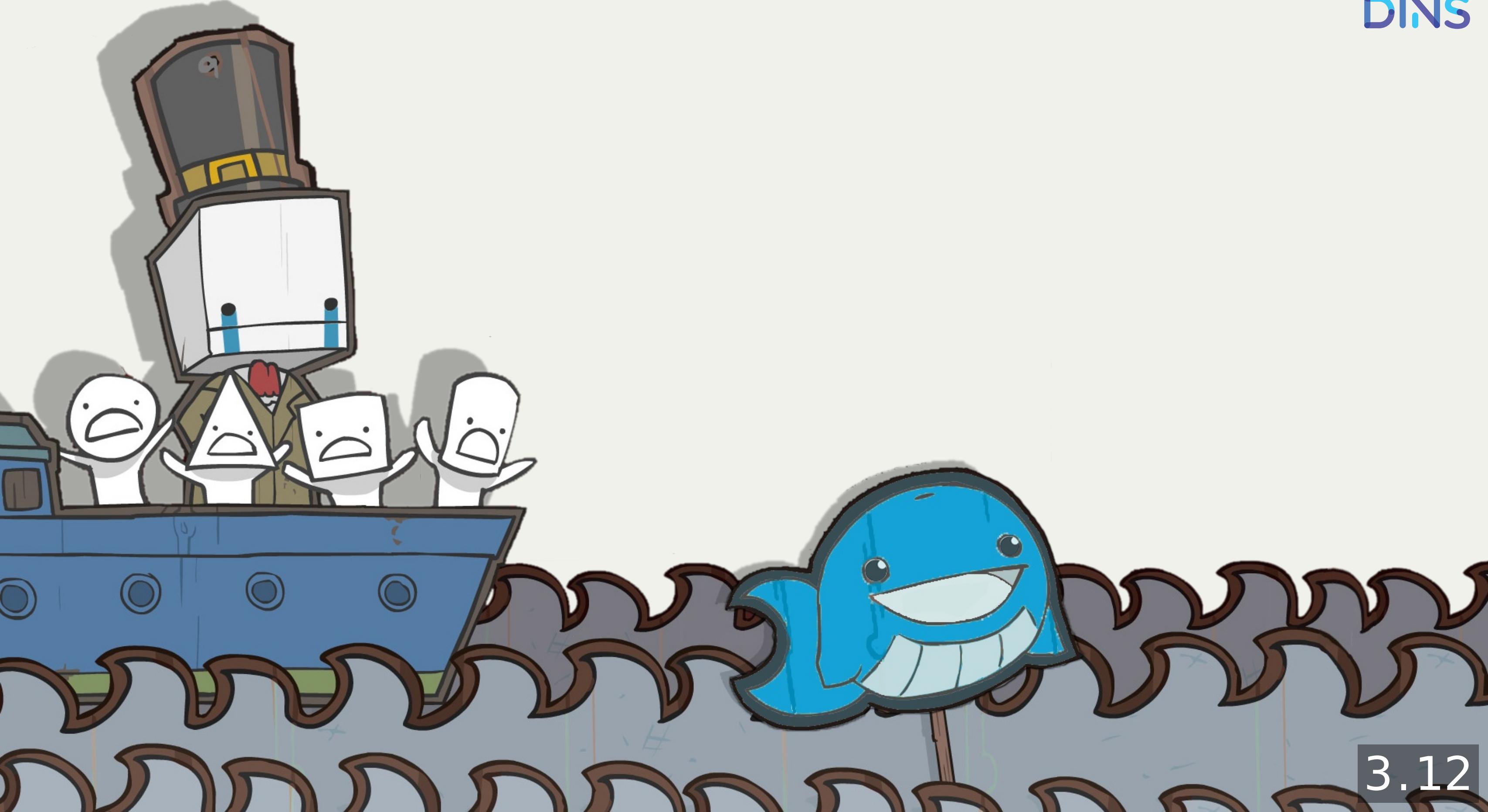
1 месяц
для всего
проекта

Регрессионное тестирование



15,000
тестов

1,5 + 3
часа



А что если

сервис сам декомпозировался

Интеграций станет **больше**

Общая регрессия будет **длиннее**

Тестовый фреймворк потребует **новых костылей**

Поддержка CI-сервера **усложнится**

Печаль. **Беда.**

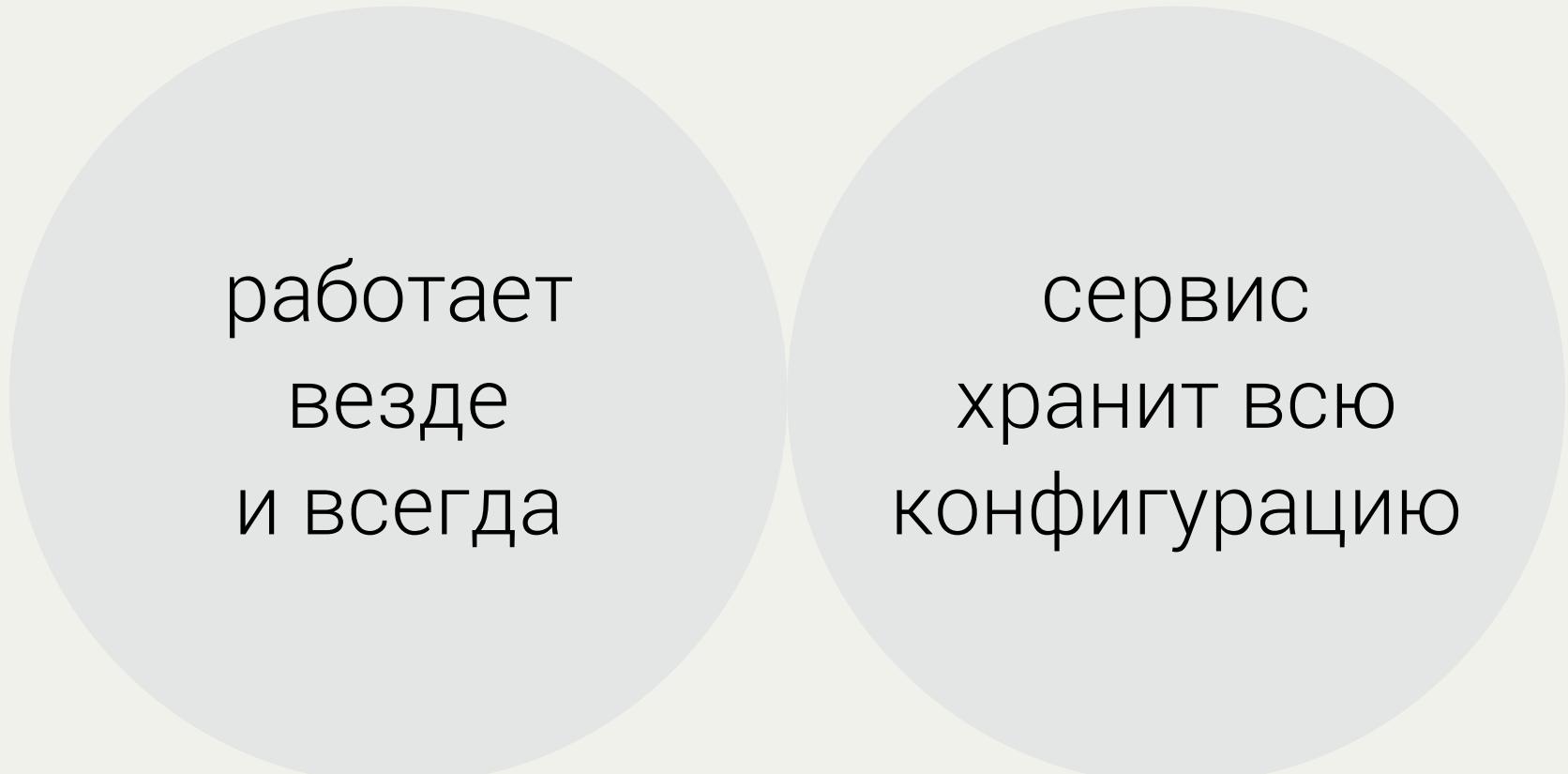


Дистрибуция

при использовании VM окружения будут различаться



Используем Docker



работает
везде
и всегда

сервис
хранит всю
конфигурацию

Меняем подход к тестированию

используем
заглушки
и моки

уменьшаем
общую
ретрессию

делаем
окружение
меньше

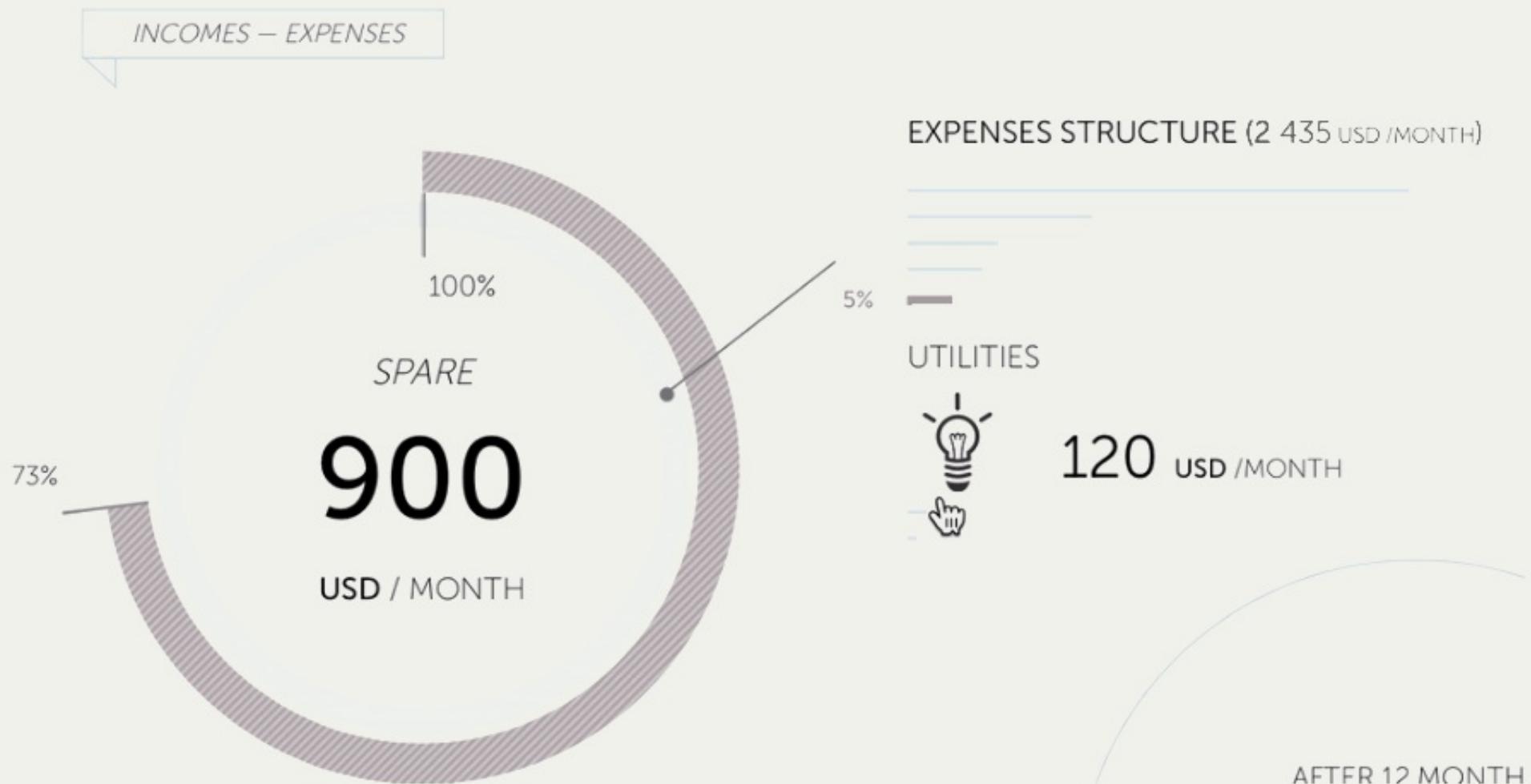
нужен пример

[build](#) [passing](#) [codecov](#) [80%](#) [license](#) [MIT License](#) [chat](#) [on gitter](#)

Piggy Metrics

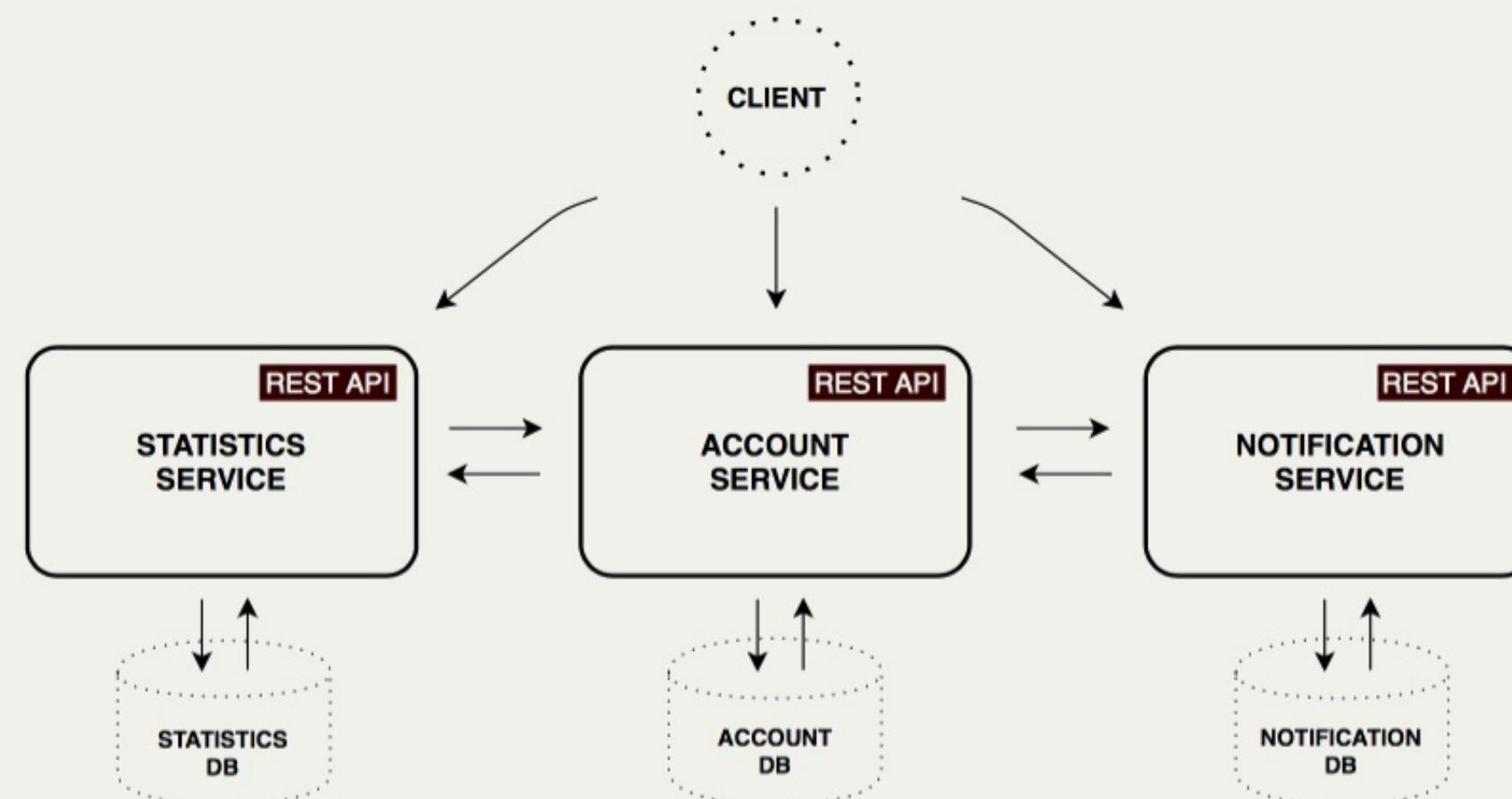
A simple way to deal with personal finances

This is a [proof-of-concept application](#), which demonstrates [Microservice Architecture Pattern](#) using Spring Boot, Spring Cloud and Docker. With a pretty neat user interface, by the way.



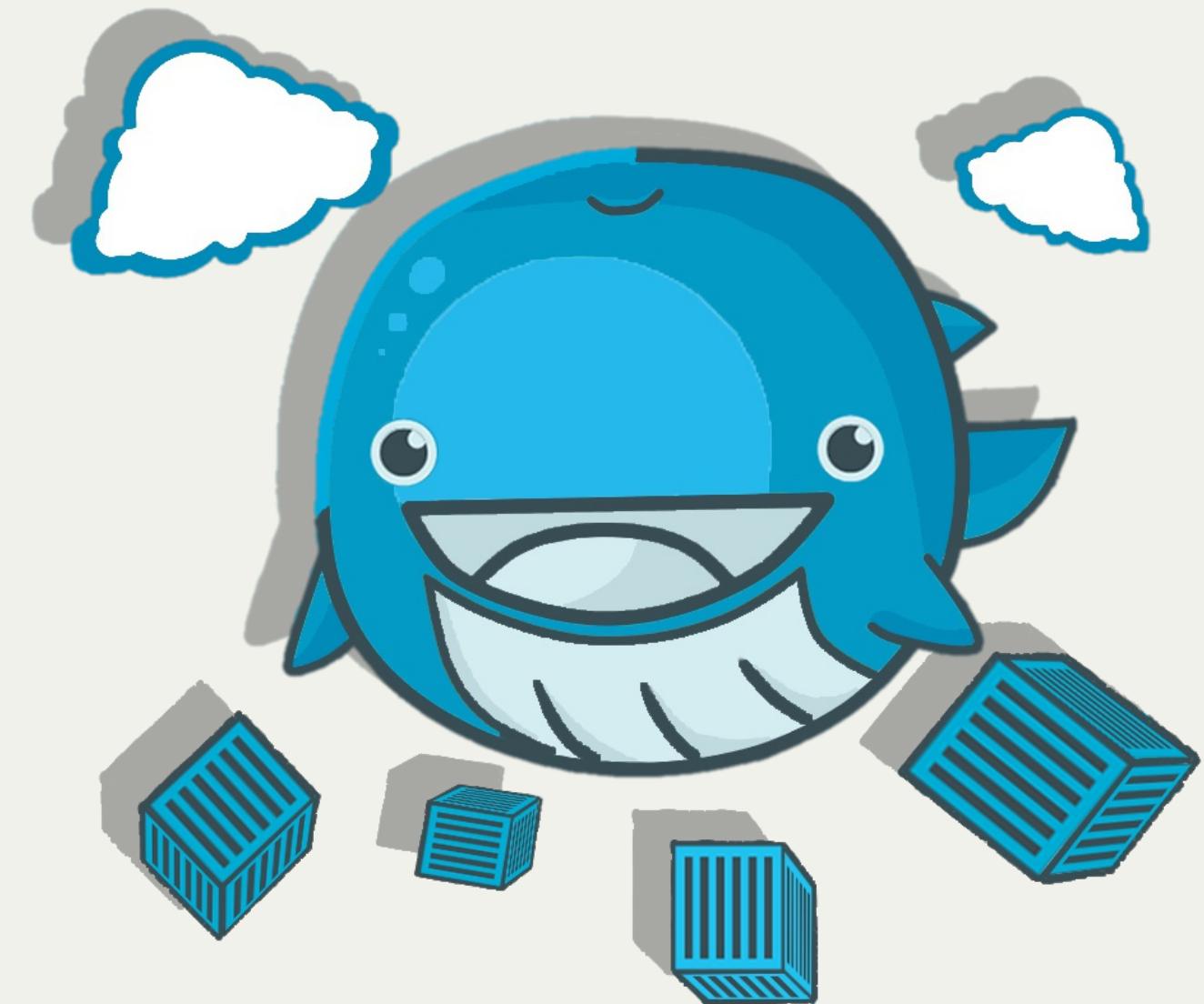
Functional services

PiggyMetrics was decomposed into three core microservices. All of them are independently deployable applications, organized around certain business capability.



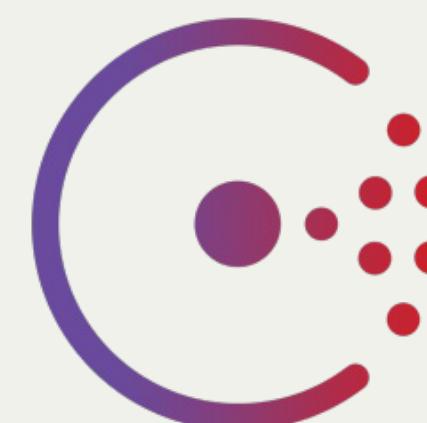
Тестовое окружение

можно сделать из подручных средств и палок





H A
P R O X Y



нужен пример

```
docker-compose
version: '2'
services:
  auth-service:
    build: auth-service

  account-service:
    build: mock-service
```

```
test task triggers compose in gradle
dockerCompose {
    useComposeFiles = [new File(getProjectDir(), 'docker-compose.yml')]
    stopContainers = true
    removeContainers = false
    removeVolumes = false
    waitForHealthyStateTimeout = Duration.ofMinutes(1)
    waitForTcpPortsTimeout = Duration.ofMinutes(3)
}

dockerCompose.isRequiredBy(test)
```

Тестовый фреймворк

— легендарное мифическое морское чудовище гигантских размеров, головоногий моллюск, известный по описаниям исландских моряков, из языка которых и происходит его название.

то есть ...



когда текущий тестовый фреймворк уже не получился ...

Что делать?

... а новый ещё не получился

Wonderful Testing Framework

независимые
модули

делаем
вклад в
open-source

требования
к коду
prod-like

Wonderful Testing Framework

DSL Технический, бизнес

Выходные данные Отчеты, логирование, дашборды

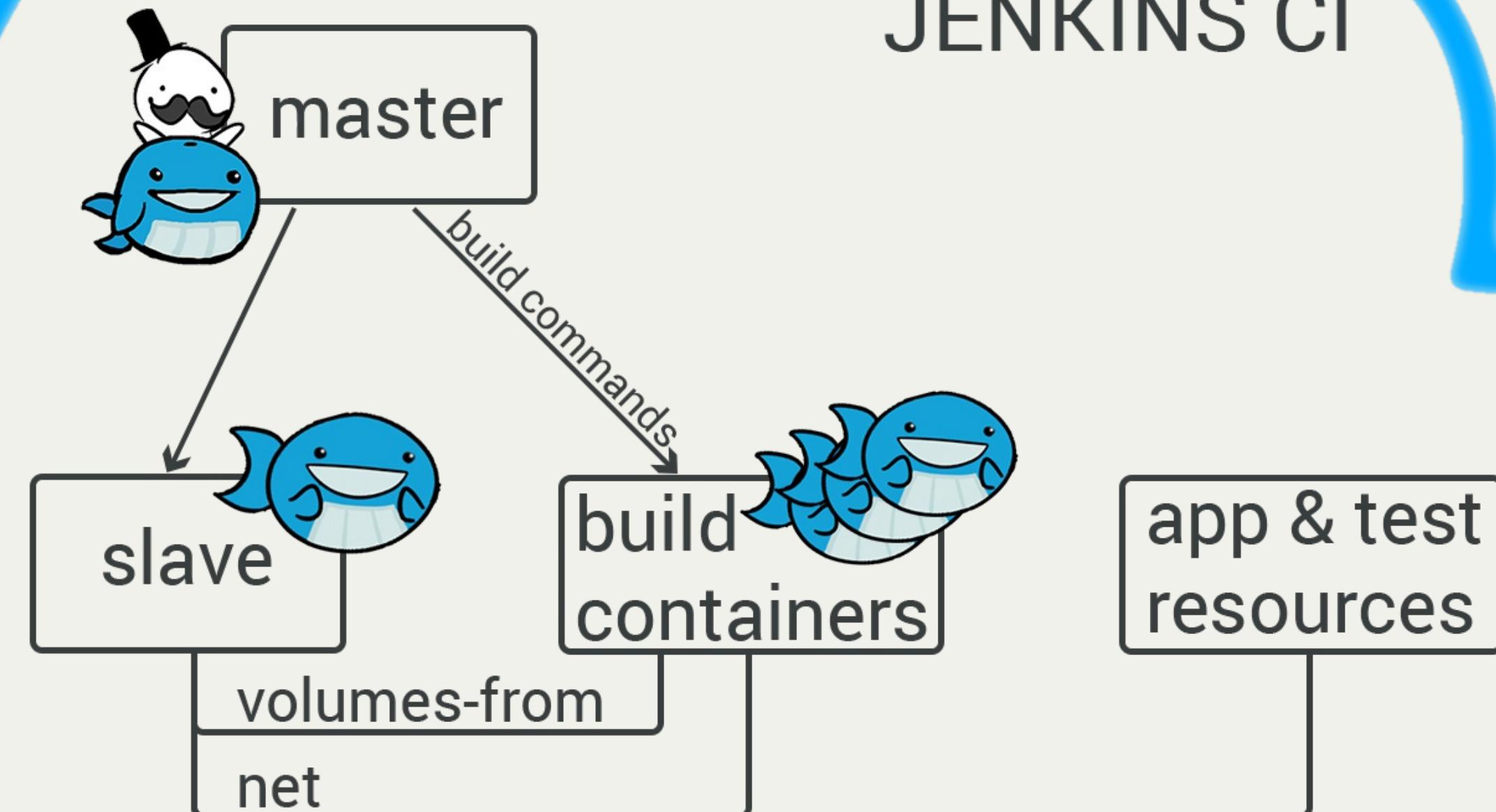
Окружение Конфигурация, моки

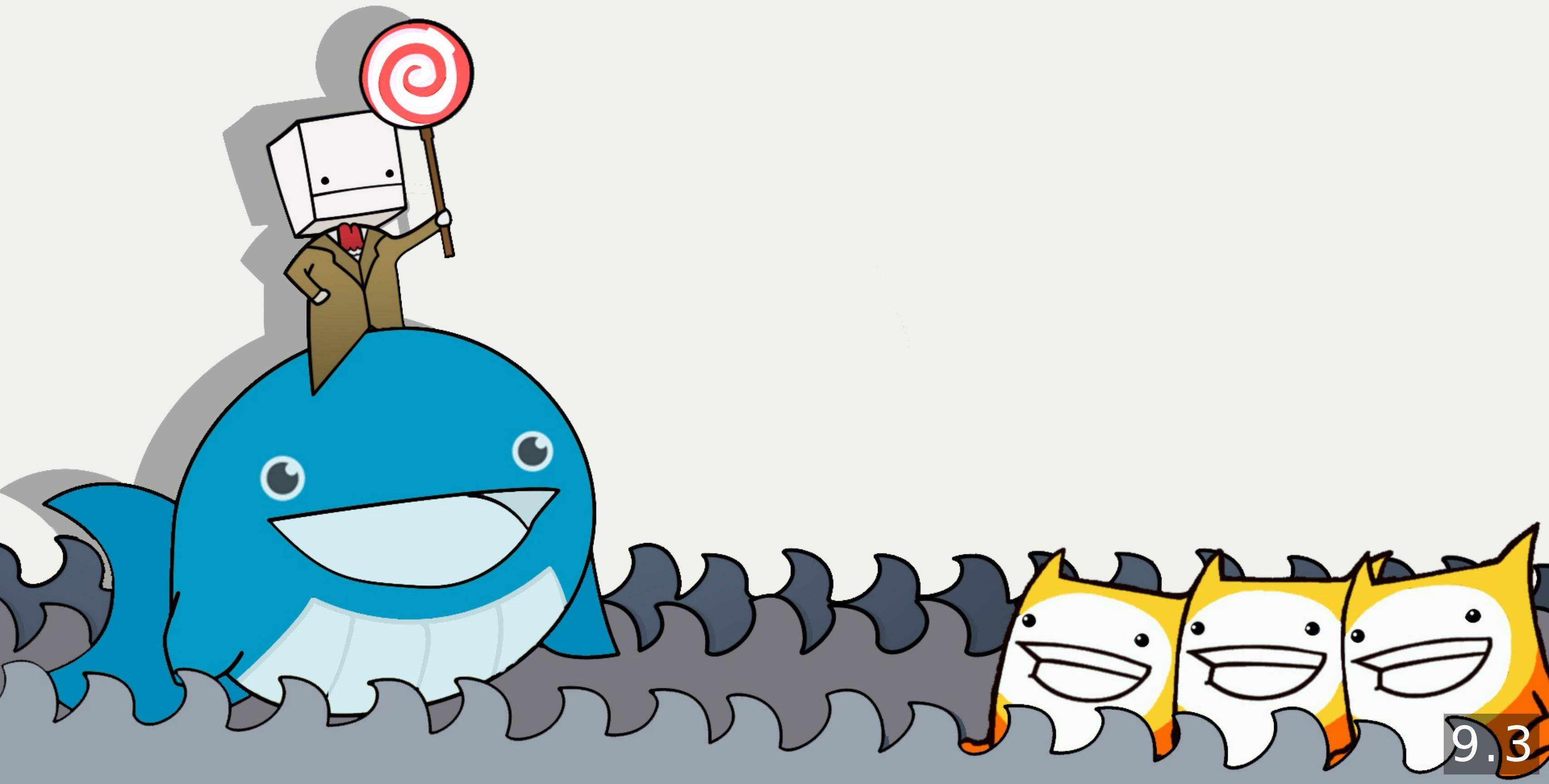
Аналитика Статистика прогона, ошибки

Интеграции Трекер задач, менеджер тест-кейсов, источники данных, CI сервер, хранилище логов

Continuous Integration & Delivery

JENKINS CI





нужен пример

```
generate allure report
def allureBuilder = docker.build('jenkins/allure-builder')

allureBuilder.inside() {
    sh "allure generate results -o reports"
}
```

```
parallel project build
for (int i = 0; i < projects.size(); i++) {
    def project = projects[i]
    builds["Build ${project}"] = {
        stage("Build ${project}") {
            gradleBuilder.inside() {
                sh "gradle ${project}:build"
            }
        }
    }
}
stage('Build') {
    parallel builds
}
```



Отчет по тестированию

разные
уровни
детализации

API
для
интеграции

результаты
в реальном
времени

ReportNG



ReportNG Sample Report

[Overview](#) · [Log Output](#) · [Coverage Report](#)

▼ Another Suite

[Only Successful Tests](#) ✓

▼ ReportNG Sample

[All Sample Tests](#) ✘

[Only Unsuccessful Tests](#) ✘

All Sample Tests

Test duration: 0.017s

Failed Configuration

org.uncommons.reportng.sample.FailedConfiguration

configure	0.000s	java.lang.RuntimeException: Configuration failed.
-----------	--------	---

Failed Tests

org.uncommons.reportng.sample.FailedTests

assertionFailure	0.000s	java.lang.AssertionError: This test failed.
assertionFailureWithMultilineMessage	0.000s	java.lang.AssertionError: This test failed. Its message is on multiple lines. The last one has leading whitespace.
assertionFailureWithOutput	0.000s	Here is some output from an unsuccessful test. java.lang.AssertionError: This test failed.
exceptionThrown	0.000s	java.lang.IllegalStateException: Test failed.

Skipped Tests

org.uncommons.reportng.sample.FailedConfiguration

thisShouldBeSkipped	0.001s	
---------------------	--------	--

org.uncommons.reportng.sample.SkippedTests

skippedDueToDependentGroup	0.001s	<i>Depends on group(s): should-fail</i>
----------------------------	--------	--

skippedDueToDependentMethod	0.000s	<i>Depends on method(s):</i>
-----------------------------	--------	------------------------------

org.uncommons.reportng.sample.SkippedTests.skippedDueToDependentGroup

Passed Tests

org.uncommons.reportng.sample.DataProviderTest

testProvider	0.007s	<i>Method arguments: "One", 1.0</i>
testProvider	0.000s	<i>Method arguments: "Three", 3.0</i>
testProvider	0.000s	<i>Method arguments: "Two", 2.0</i>
testProvider2	0.000s	<i>Method arguments: "Three", 3.0</i>
testProvider2	0.000s	<i>Method arguments: "Two", 2.0</i>

Allure Report



[Back to allure-core_master-deploy](#) [index](#) [Zip](#)

Allure

xU Many info test

Failed

3 test cases

Total suite	#	Title	Duration	Status
Allure migration test	1	Test with long assertion text	661ms	FAILED
Allure migration test	2	<script>3443</script>	758ms	FAILED
Allways pass test	3	Lot of steps test	61ms	BROKEN

! <script>3443</script>

AssertionError: This test should be failed [Show trace](#)

Description

Testsuite has testcases with many steps and many lines in description Single-line description

Links

TMS-1, JIRA-1, JIRA-2

Steps

[16:58:59] Test started

> [16:59:00] Make this test failed ✘

[16:58:59] Test finished with status: FAILED

JSON Attachment 3 KB

XML Attachment 65 B

JPG Attachment 32.9 KB

CSV Attachment 134 B

WFRM Attachment 224.1 KB

Overview

Defects

xUnit

Behaviors

Graph

Timeline

Many test

Non test

En

Collapse

Extent Report



Extent	Report Name		Oct 14, 2016 10:47:05 AM	v3.0.0									
	Status Category Dashboard Search												
	Tests	 4 test(s) passed 3 test(s) failed, 2 others	 8 step(s) passed 3 step(s) failed, 6 others	 8 step(s) passed 3 step(s) failed, 6 others									
	Tests	<h2>HTMLTagsTest</h2> <p>Oct 14, 2016 10:47:05 AM Oct 14, 2016 10:47:05 AM 0h 0m 0s+47ms</p> <p> </p> <p>HTML can be embedded anywhere in the report to create meaningful logs and messages.</p> <table border="1"> <thead> <tr> <th>Status</th> <th>Timestamp</th> <th>Details</th> </tr> </thead> <tbody> <tr> <td> Pass</td> <td>10:47:05 AM</td> <td>Labels can be created using: Success Fail Warning Info Skip</td> </tr> <tr> <td> Fail</td> <td>10:47:05 AM</td> <td> <pre>Success Fail Warning Info Skip</pre> </td> </tr> </tbody> </table>			Status	Timestamp	Details	Pass	10:47:05 AM	Labels can be created using: Success Fail Warning Info Skip	Fail	10:47:05 AM	<pre>Success Fail Warning Info Skip</pre>
Status	Timestamp	Details											
Pass	10:47:05 AM	Labels can be created using: Success Fail Warning Info Skip											
Fail	10:47:05 AM	<pre>Success Fail Warning Info Skip</pre>											
	CategoryTest	Pass											
	AuthorTest	Pass											
	ExceptionTest-1	Fail											
	ExceptionTest-2	Fail											

Report Portal



DEFAULT_PROJECT ▾ |

admin SUPERADMIN

+ All > superadmin_TEST_E... > BlackBox Stable > Endpoint Path > testDelete

#1 X #2 X #3 X #4 X #5 X #6 X #7

Log level Trace Debug Info Warn Error Attachment

Message Time ▾

[2016-10-21 10:24:28,645] [INFO] [pool-1-thread-1-87] Before method section. begin	2016-10-21 13:24:28
[2016-10-21 10:24:28,667] [INFO] [pool-1-thread-1-87] Request... 10:24:25:820	2016-10-21 13:24:28
[2016-10-21 10:24:28,691] [INFO] [pool-1-thread-1-87] Request method: POST Request path: http://pla01-t04-pwr01:80/restapi/oauth/token Proxy: <none> Request params: <none> Query params: <none> Form params: grant_type=password username=14182040020 extension=101 password=Test!123 pin endpoint_id Path params: <none>	2016-10-21 13:24:28

нужен пример

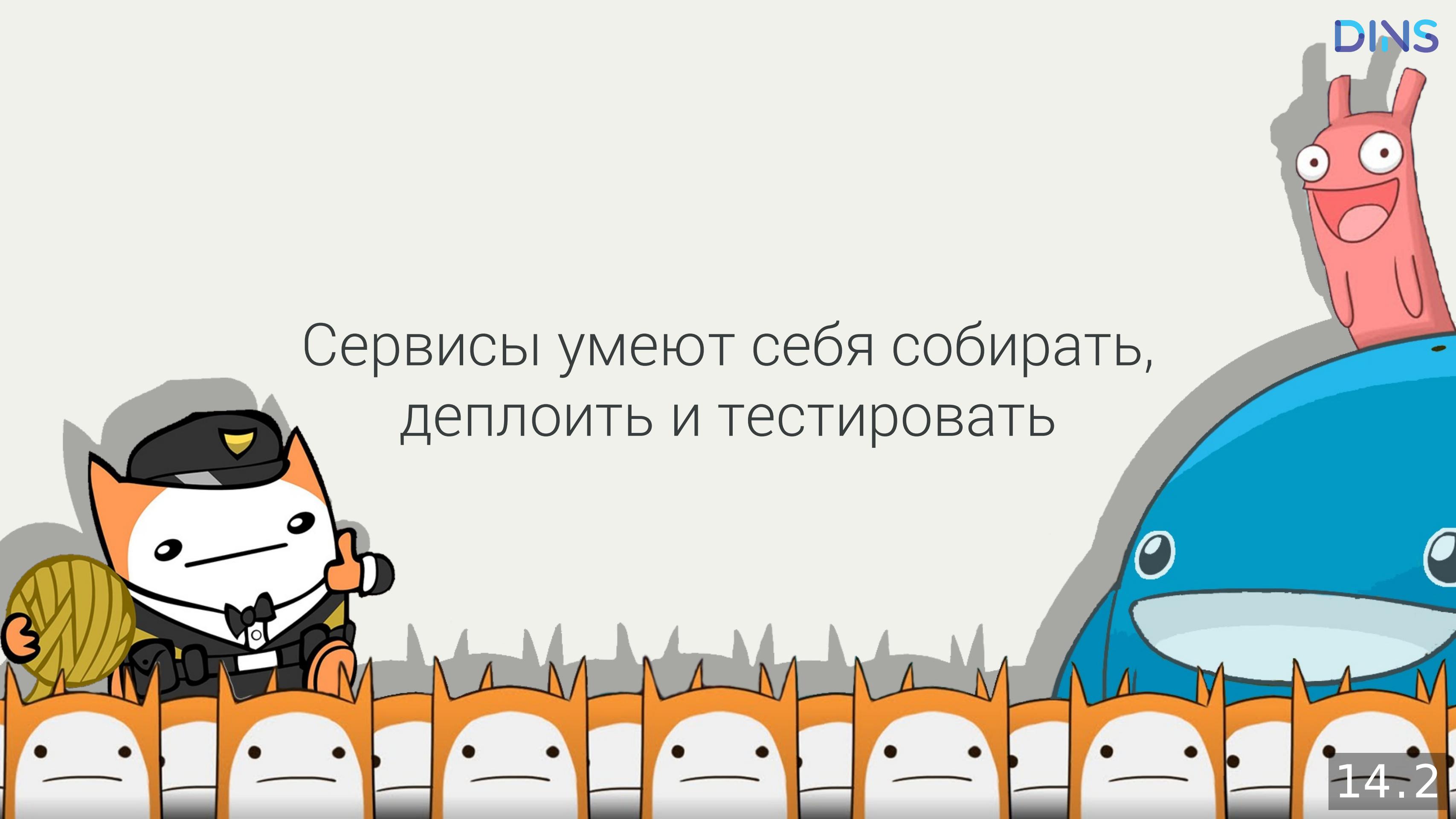
```
integrate allure
@Step("{0}")
private void logInfo(String s) {
    log.info(s);
}

logInfo("Prepare account");
```

Что получили

- Артефакты, которые можно развернуть где угодно
- Вся конфигурация хранится с кодом сервиса
- Удобные инструменты для отчетов

и самое главное...



Сервисы умеют себя собирать,
деплоить и тестировать

github.com/dins-heisenbug-2016
engineering.riotgames.com
battleblocktheater.com

Почта
info@dins.ru
andrey.smirnov@dins.ru
dmitry.krasilnikov@dins.ru

Твиттер
[@andreich](https://twitter.com/andreich)
[@kr_dmitry](https://twitter.com/kr_dmitry)

Вопросы