



Практический кейс внедрения SDLC. Сертифицированная Java: через тернии к звездам

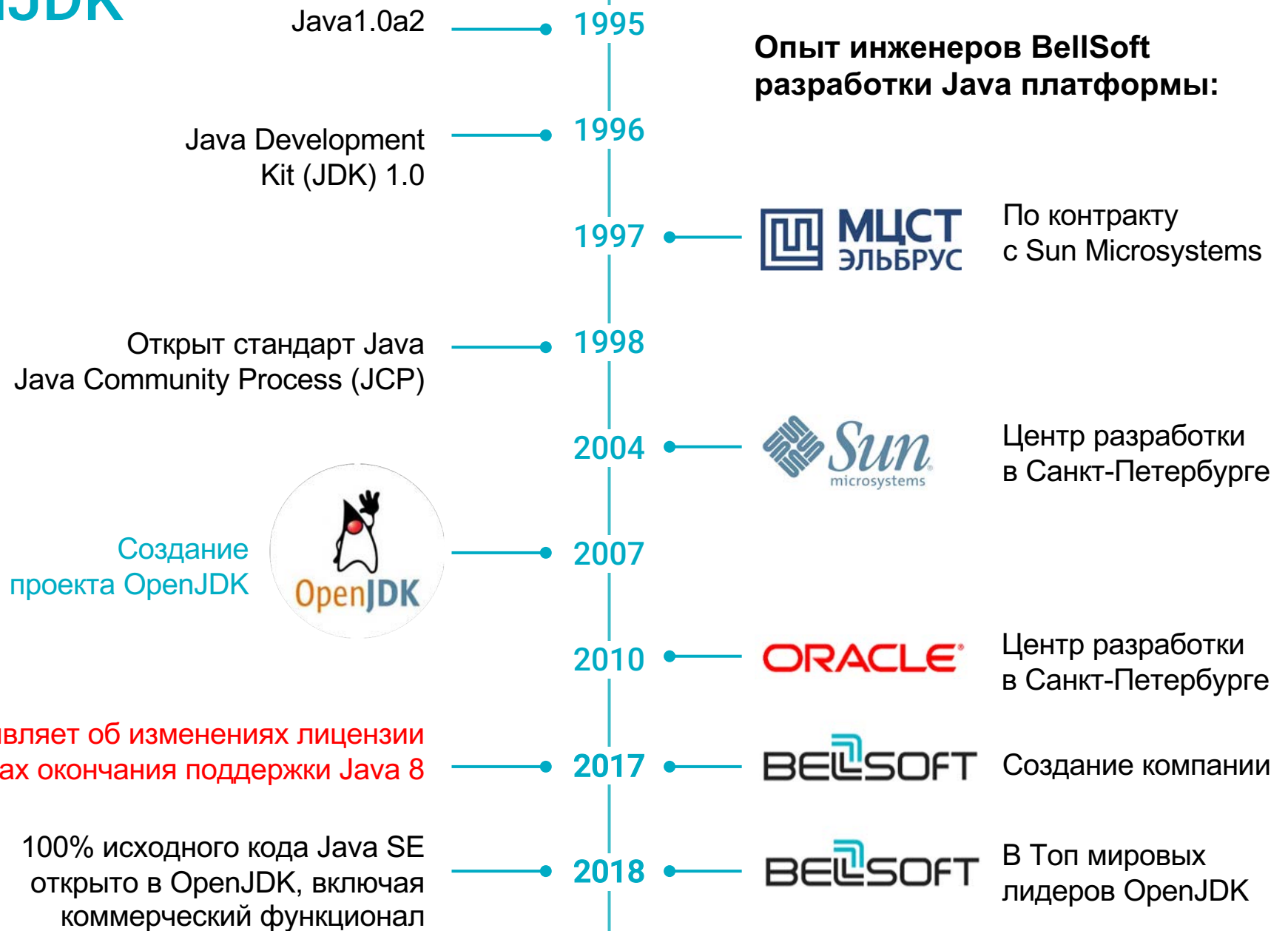
BellSoft 2022

www.bell-sw.com

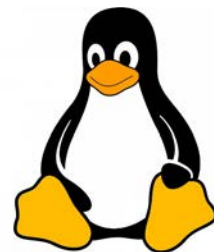
От Java до OpenJDK и BellSoft

20 лет опыта разработки Java платформы

В 2020 BellSoft избрана в исполнительный комитет Java Community Process (JCP)



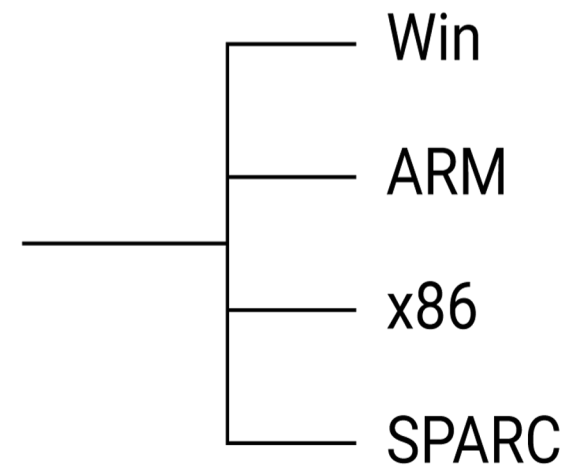
Наш вклад в Open source



GraalVM™



Java™



BellSoft staff

39 Сотрудников

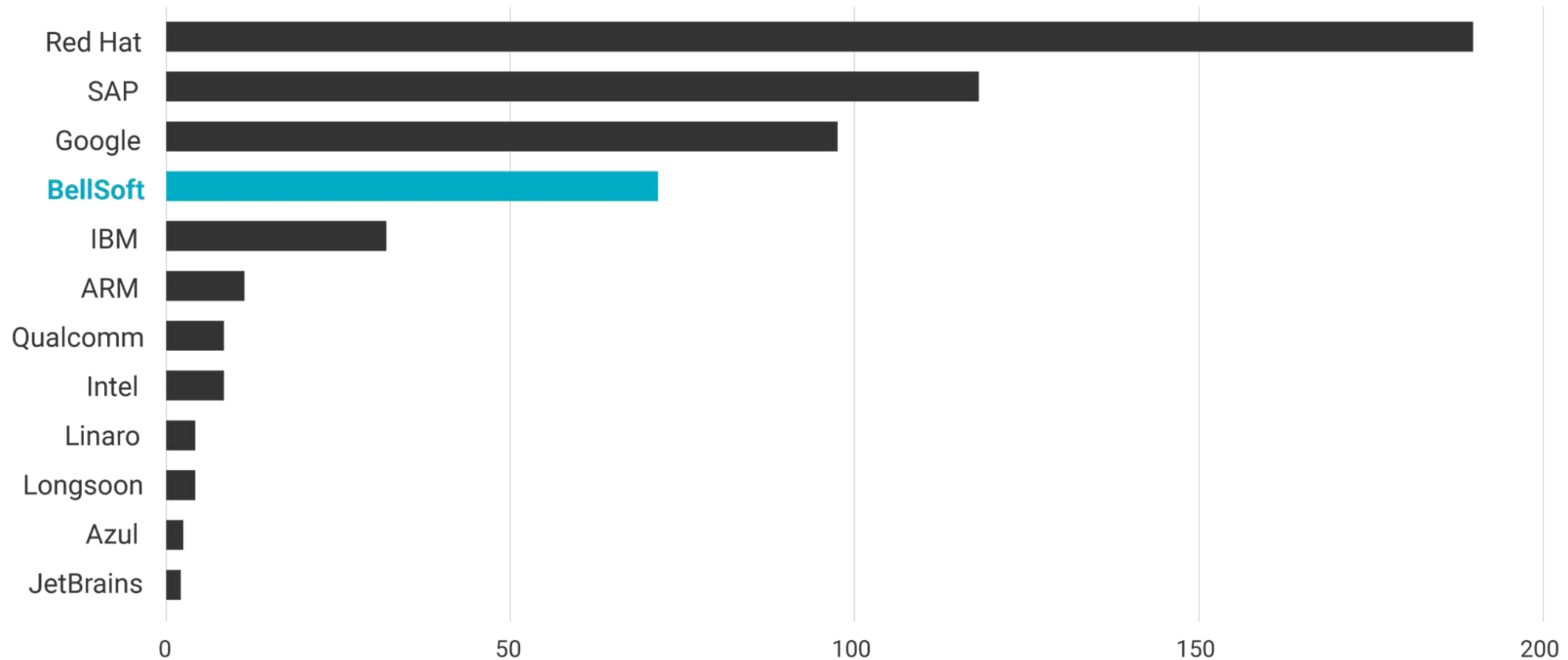
29 Инженеров

7 OpenJDK Committers

3 OpenJDK reviewers



Number of external contributions to OpenJDK upstream (8.2017-8.2018)



Развиваем OpenJDK

JEP 315: Improve Aarch64 Intrinsics

Owner Dmitrij Pochevko
Type Feature
Scope Implementation
Status Closed / Delivered
Release 11
Component hotspot / compiler
Discussion hotspot dash compiler dash dev at openjdk dot java dot net
Effort L
Duration L
Reviewed by Mikael Vidstedt, Vladimir Kozlov
Endorsed by Vladimir Kozlov
Created 2017/10/10 12:40
Updated 2018/09/10 14:45
Issue [8189104](#)

Summary

Improve the existing string and array intrinsics, and implement new intrinsics for the `java.lang.Math` `sin`, `cos` and `log` functions, on AArch64 processors.

JEP 388: Windows/AArch64 Port

Authors Monica Beckwith, Ludovic Henry, Bernhard Urban-Forster
Owner Vladimir Kozlov
Type Feature
Scope Implementation
Status Closed / Delivered
Release 16
Component hotspot
Discussion aarch64 dash port dash dev at openjdk dot java dot net
Effort M
Duration S
Blocks JEP 391: macOS/AArch64 Port
Reviewed by Vladimir Kozlov
Endorsed by Mark Reinhold, Vladimir Kozlov
Created 2020/06/29 19:13
Updated 2021/08/28 00:28
Issue [8248496](#)

Summary

Port the JDK to Windows/AArch64.

JEP 386: Alpine Linux Port

Owner Boris Ulasevich
Type Feature
Scope Implementation
Status Closed / Delivered
Release 16
Component hotspot
Discussion portola dash dev at openjdk dot java dot net
Effort M
Duration M
Reviewed by Alan Bateman, Vladimir Kozlov
Endorsed by Mikael Vidstedt
Created 2019/08/13 10:33
Updated 2021/08/28 00:30
Issue [8229469](#)

Summary

Port the JDK to Alpine Linux, and to other Linux distributions that use musl as their primary C library, on both the x64 and AArch64 architectures,

JEP 391: macOS/AArch64 Port

Authors Anton Kozlov, Vladimir Kempik
Owner Vladimir Kempik
Type Feature
Scope JDK
Status Closed / Delivered
Release 17
Component hotspot
Discussion aarch64 dash port dash dev at openjdk dot java dot net
Effort M
Duration M
Depends JEP 388: Windows/AArch64 Port
Reviewed by Andrew Haley, Vladimir Kozlov
Endorsed by Vladimir Kozlov
Created 2020/08/07 07:08
Updated 2021/09/29 17:30
Issue [8251280](#)

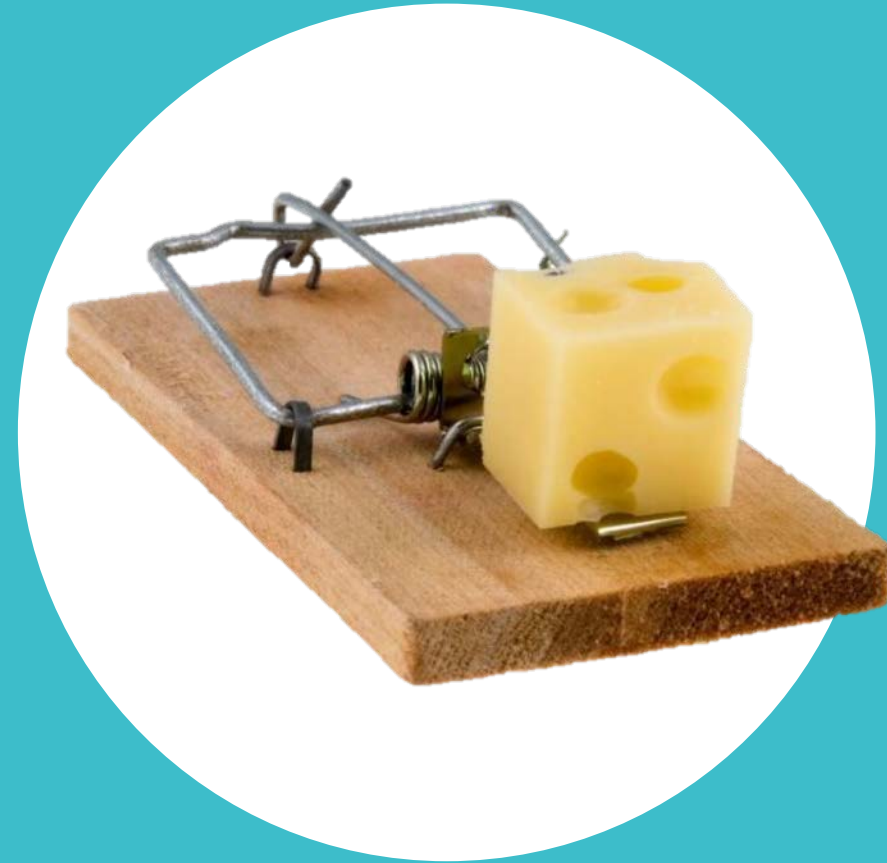
Summary

Port the JDK to macOS/AArch64.

Проблема: лицензионная ловушка Java

- Oracle изменили лицензию для Java 11 и обновлений Java 8
- OTN (Oracle Technology Network) License
- Только разработка, тестирование и демо цели
- Для использования Java SE 11 и обновленных версий Java 8 в продакшене необходима коммерческая лицензия

На протяжении 25 лет Oracle (Sun) предоставлял дистрибутив Java Development Kit и обновления безопасности всем желающим бесплатно, но теперь мир Java изменился



OpenJDK сегодня

OpenJDK Upstream

Версии:

Developing version: JDK 18

Feature versions: JDK 17

LTS versions: JDK 8, JDK 11, JDK 17

Legacy versions: JDK 7

Ports:

Windows X86 64 & 32 bit

Linux x86 64 & 32 bit

Linux ARM 64 & 32 bit

Alpine Linux x86 & ARM

OpenJDK проекты

Project Panama

Project Loom

Project Portola

Windows ARM Port

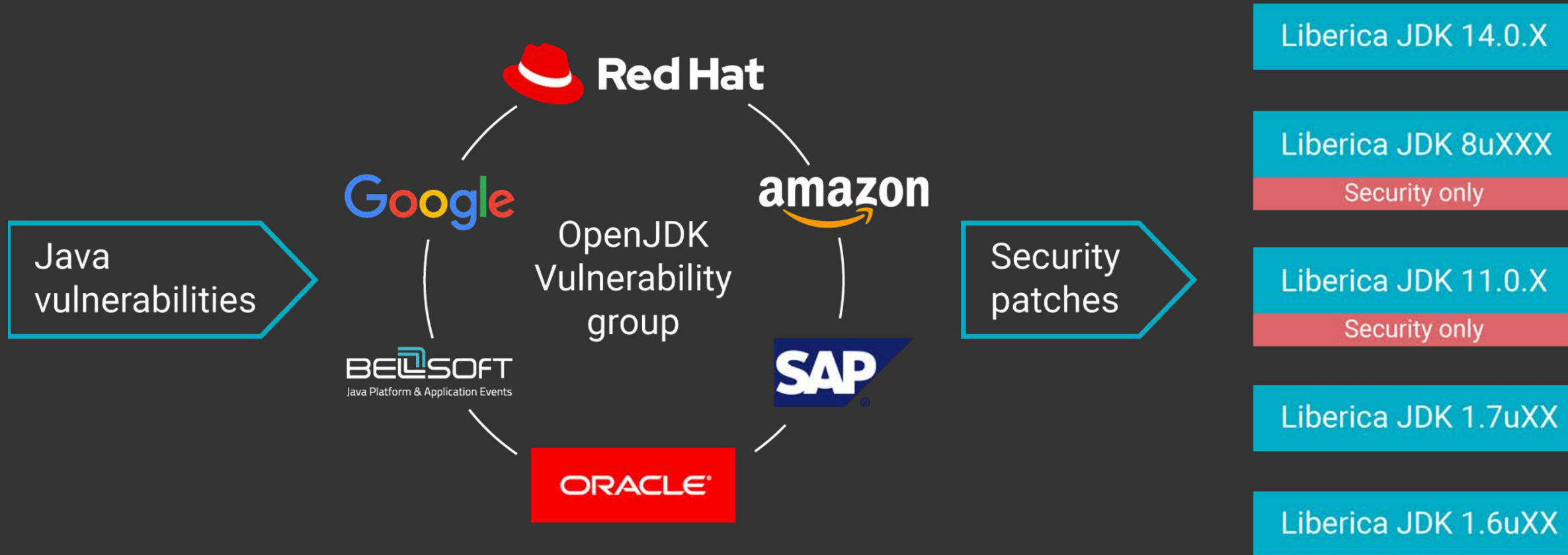
RISC-V Port

Security



Liberica JDK Security

Collaborative approach

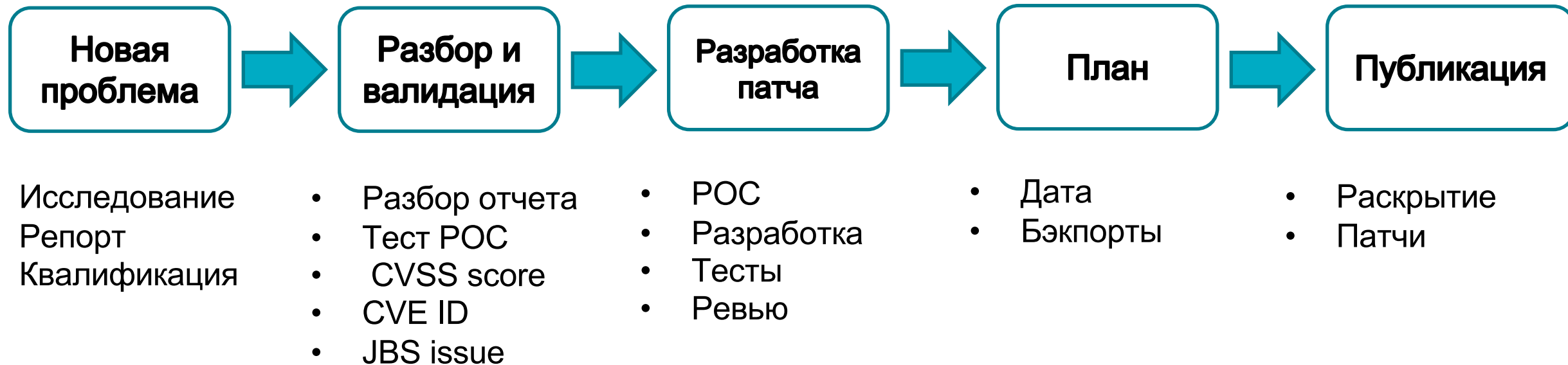


Liberica JDK 8u292 выпущен через 2 часа после Oracle Java SE

OpenJDK Vulnerability group

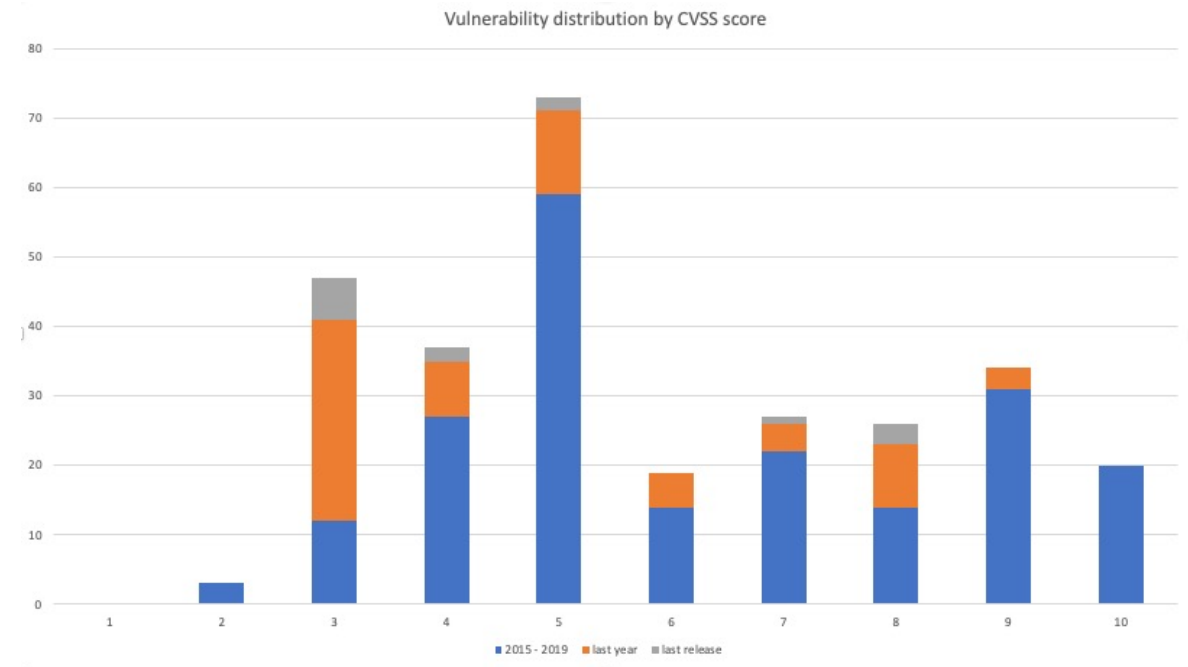
- Замкнутая группа в рамках OpenJDK включает 11 организаций (04.06.21)
 - Участники OpenJDK
 - Согласие с правилами обмена конфиденциальной информацией
 - Подтвержденный опыт работы с дефектами безопасности
 - Быть признанным экспертом в OpenJDK и доступ к ТСК
 - Новые члены группы принимаются путем голосования (3 и более голосов)
- Отчеты о найденных уязвимостях в OpenJDK
- Работает совместно над созданием фиксов
- Разработка проходит в режиме ограниченного доступа

Vulnerability workflow



Данные устраненных проблем с безопасностью в Java

CVSS Score	2015 - 2019	2019	last release
1	0	0	0
2	3	0	0
3	12	29	6
4	27	8	2
5	59	12	2
6	14	5	0
7	22	4	1
8	14	9	3
9	31	3	0
10	20	0	0
Total	166	70	14



Кто любит пиво?

- Рецепт пива известен
- Ингредиенты доступны
- Вы будете варить пиво сами?

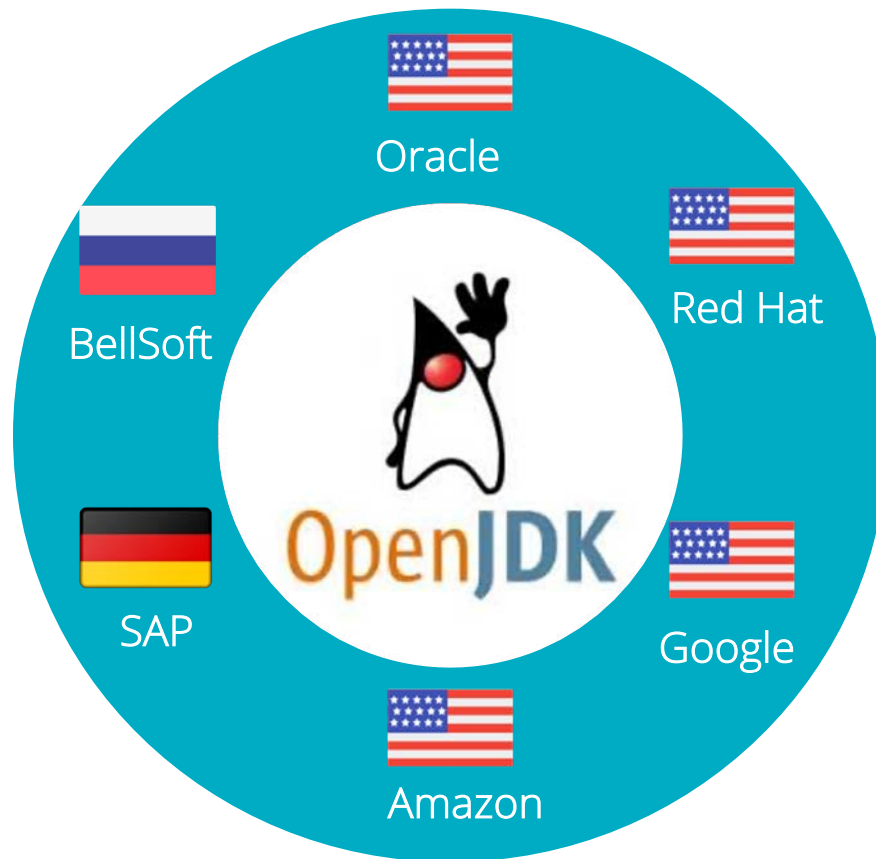


Дистрибутивы OpenJDK*

*топ 5 активных участников проекта OpenJDK



LIBERICA JDK



Amazon Corretto

Java у каждого своя



Legacy



Modern



Desktop

Богатое наследство

- Устаревшие версии JDK 1.4, 1.6, 1.7
- Вендорские сервера приложений WebLogic, WebSphere
- Редкие платформы Power, SPARC
- Скоро мы все перепишем как надо, но пока есть более срочные задачи



Красота и простота

- Толстые клиенты для богатого наследства
- Java WebStart
- Java Applets
- JavaFX



Гламур и практичность

- Микросервисная архитектура
- Контейнеризация
- Облака
- Native image
 - Quarkus, Micronaut, Spring Boot Native
 - GraalVM



Unified Java run-time

Supported

Secured

Standard

Superb

Java SE standard runtime by BellSoft – Liberica JDK

- **Liberica JDK 8 & 11 поддерживаемые платформы:**
 - Linux x86 32 & 64 bit
 - Windows x86 32 & 64 bit
 - Windows ARM 64 bit (April 2021)
 - MacOS x86 & Apple Silicone M1 64 bit
 - Linux ARM 32 & 64 bit
 - Solaris SPARC & x86
 - PowerPC 64-bit
 - Other platform by request
- **Российские платформы:**
 - AltLinux, Astra Linux, RedOS, Rosa и даже больше
 - Baikal, Elbrus SPARC



Java SE standard runtime by BellSoft – Liberica JDK Legacy

- **Liberica JDK 1.6 & 1.7 supported platforms:**
 - Linux x86 32 bit
 - Linux x86 64 bit
 - Windows x86 32 bit
 - Windows x86 64 bit
 - Other platform by request
- **Liberica JDK security updates releases in parallel with Oracle Java SE**



TCK — гарантия стандарта Java SE

- TCK — тестовая сюита более 140 тысяч тестов
- Гарантирует, что при смене рантайма поведение приложений не изменится
- Каждая сборка Liberica JDK прошла TCK тестирование
- BellSoft лицензирует TCK у Oracle
(<http://openjdk.java.net/groups/conformance/JckAccess/jck-access.html>)

Пакеты доставки

- Liberica JDK доступна на bell-sw.com
- Пакеты и инсталляторы:
 - Windows binaries (with installer)
 - MacOS binaries (with installer)
 - Linux Repos: APT, YUM
 - Docker containers at hub.docker.com
 - Full JDK with Debian or CentOS
 - Liberica JDK Lite with Alpine Linux – only 41 MB
 - Самый маленький в мире
 - Приватный Docker репозиторий



Liberica Native Image Kit

- GraalVM based
- Alpine Linux support
- MUSL support



BellSoft releases
Liberica Native Image Kit

Конференция JRush https://www.youtube.com/channel/UCAsJMQdRujw4_r6k-eATMDw

Клиенты BellSoft



Promsvyazbank



А у вас есть сертификат?

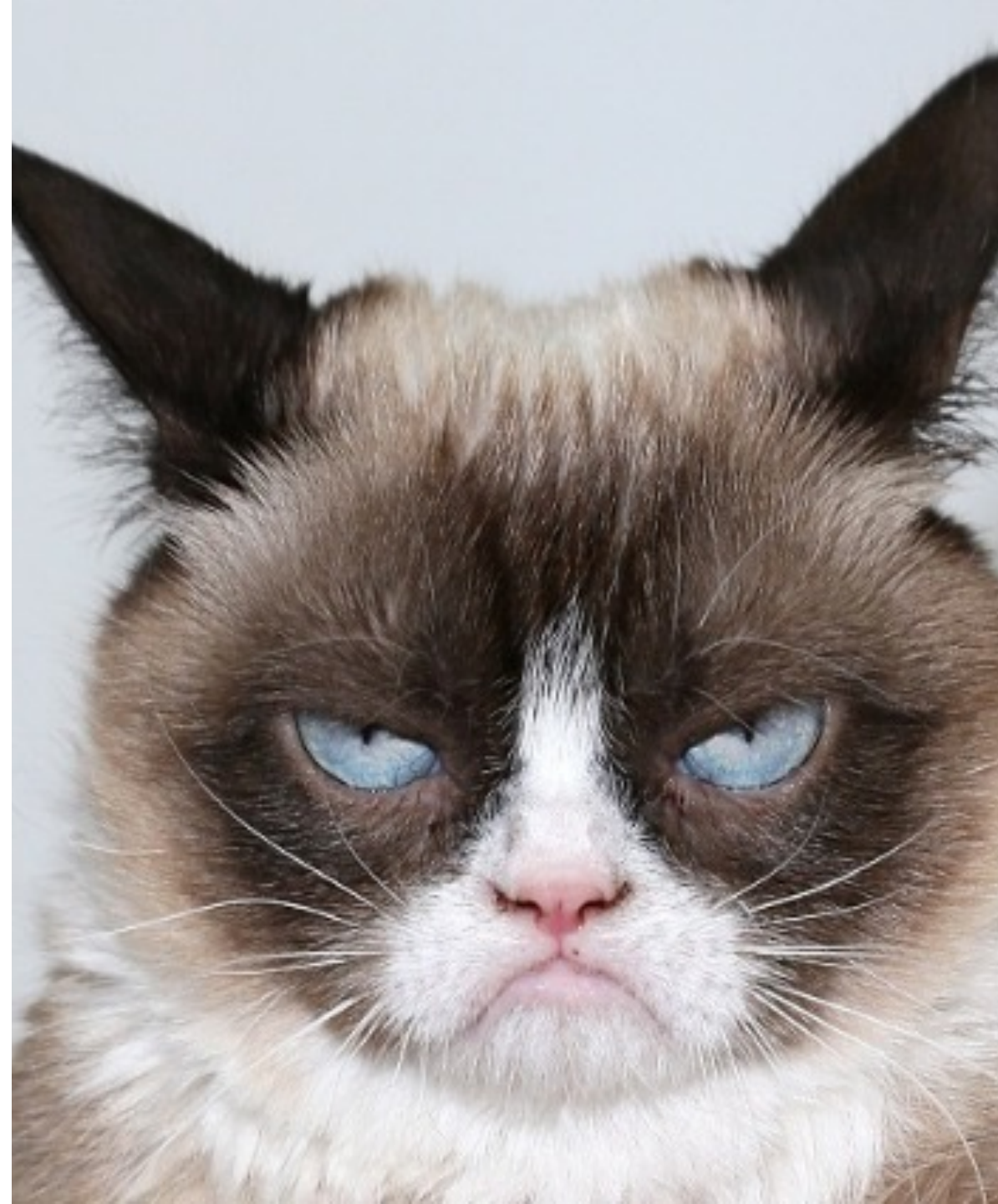


Сертифицированная Java миф?



Перепишите на C/C++

- Код приложения
 - Нет методик проверки кода Java
- Среда исполнения
 - Нет профиля сертификации
 - Исходники больше 1 GB
 - Ответственность за огромную кодовую базу тяготит



Путь к сертификату: Лицензия



Безопасный процесс разработки ПО

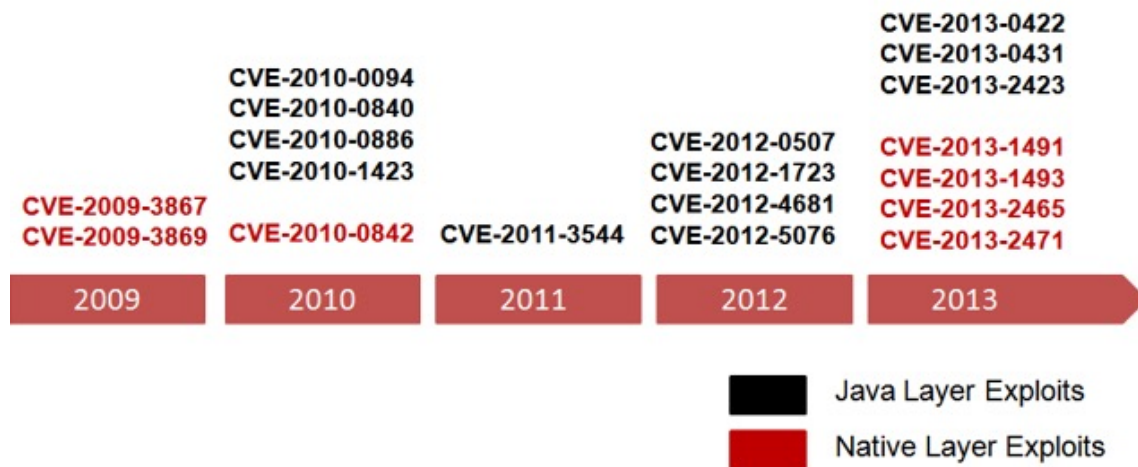


- **Философия:** полная автоматизация процессов, любые новые отказы подвергаются анализу, на ранних этапах работы над релизом для выявления отказов проводится анализ ошибок и рефакторинг
- **Объем работ:**
 - Legacy версии JDK (1.6, 1.7)
 - 14 стандартных платформ
 - 290 бинарных файлов
 - 192 млн результатов тестирования в рамках одного релиза
 - ПО производится в соответствии с промышленным процессом SDL
- **Тестирование:**
 - 100% ТСК
 - для выявления уязвимостей
 - Регрессионное и функциональное тестирование по уровням с растущим числом прогонов
 - Регрессионное тестирование производительности
 - Стресс-тестирование (jcmstress, J1T tester, Big Apps), фаззинг
 - Тесты, встроенные в популярные фреймворки (Lucene, Spring)
 - Тесты, предоставленные клиентами или встроенные в приложения (при наличии)
 - Статический анализатор SVACE от ИСП РАН

Путь к сертификату: практики безопасной разработки

Усиленное внимание к практикам безопасной разработки:

- применение практик безопасного программирования
- проведение автотестов
- динамический анализ кода
- функциональное тестирование



Динамический анализ кода

Фаззинг тестирование:

- Нативной части – AFL++
- Java кода – JQF+Zest
- Сбор информации о покрытии – Icov, Jcov

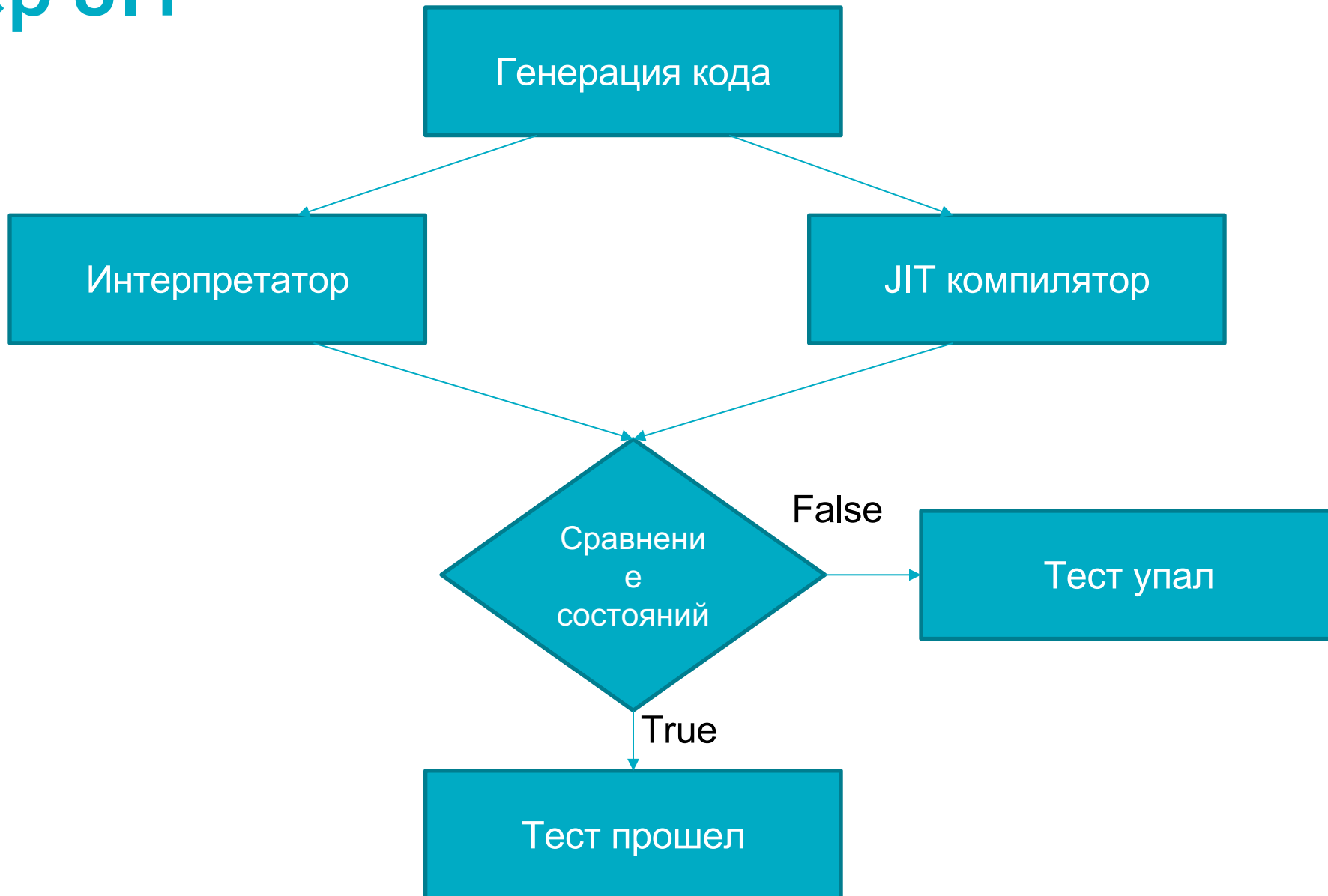


SDL - Svace

- Более 100 000 срабатываний
- Стратегия обработки Critical with Very High and High Availability
 - 98 срабатываний
- 26 дефектов исправлены в JDK 8
- 22 дефекта исправлены в JDK 11



Фаззер JIT



Мечта становится реальностью

- Функции безопасности
 - Контроль допустимости исполняемого бинарника
 - Контроль целостности исполняемого бинарника
 - Очистка памяти после сборки мусора
 - Регистрация событий безопасности
 - Ограничение доступности проблемных зон

Проблемные зоны

- Динамическая компиляция
- Генерация байткода в процессе исполнения
- Динамическая загрузка классов
- Интерфейс плагинов виртуальной машины
- Потенциально эксплуатируемый API
 - Reflection
 - `sun.misc.unsafe`

Цель

Сохранение работоспособности
Java программ и удовлетворение
при этом требованиям
безопасности с приемлемой
потерей производительности





Спасибо!

www.libericajdk.ru