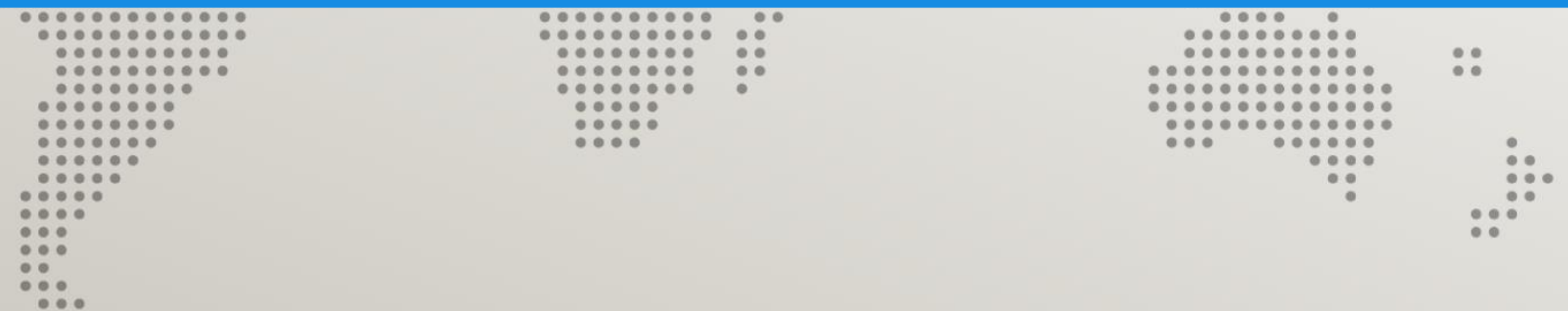




# Open Source



Игорь Ермаков, Dep. CTO Mail.Ru Group



## Примечание

- Все сказанное в этом докладе является рефлексией работы с OpenSource в разных компаниях на протяжении последних 20 лет.

Не на все вопросы есть ответы в этой презентации, и цель этой презентации: поделиться опытом, а также узнать об вашем опыте.

Также в этой презентации не затронуты юридические вопросы, вопросы импортозамещения — их можно обсудить за кофе, как и иные не затронутые вопросы.

Спасибо!

## Содержание

- OpenSource и Mail.Ru Group. История одного пользователя
- Вивисекция OpenSource
- Проблемы OpenSource проектов в разрезе жизненного цикла компании
- Как начать пользоваться OpenSource (с минимум рисков)

## Open Source и Mail.Ru Group.

- Несколько фактов:
  - Mail.Ru уже более 20 лет использует OpenSource в Core бизнесе
  - Активная аудитория Mail.Ru Group перевалила за 140 млн пользователей
  - Отказоустойчивость удерживается 99.95%
  - Постоянно возрастающая нагрузка, в пиках достигающая 5 млн транзакций в минут на определенные системы.

## Mail.Ru Group

- Mail.Ru – open source компания по своему ДНК
- 20 лет назад LAMP +
- ~~Oracle~~, Mysql, PostgreSQL
- Mdadm vs hardware
- Redis, Mongo, Cassandra, Rabbit, Kafka, Zookeeper
- Php, python, go, c, c++, perl

## Почему Mail.Ru Group использует OpenSource

- OpenSource и не OpenSource ломаются. OpenSource мы можем починить сами и/или быстро найти экспертизу на рынке
- Time2Market. Расстояние между проблемой и командой минимально.
- OpenSource очень активно развивается, что дает возможность использовать современные достижения без каких либо формальностей — а это влияет на Time2Market.
- Высокое доверие к исходному коду
- Just for fun

## Типичный путь внедрения opensource решения

- Постановка задачи
- Early adopter
- Интеграция в решения и тесты
- Вывод в production
- Успешная и безуспешная эксплуатация
- Патчи, патчи, патчи
- Contribute to master

## Tarantool тогда

commit 9b8dd7032d05e53ffcbde78d68ed3bd47f1d8081

Author: Yuriy Vostrikov <vostrikov@corp.mail.ru>

Date: **Thu Aug 12 11:39:14 2010 +0400**

Initial public import



## Tarantool сейчас

- 131 репозиторий различных утилит
- Более 80 committers
- Новое B2B направление в компании
- Tarantool-core, Tarantool Enterprise, MDM, DataGrid,
- Клиенты и пользователи: Мегафон, МТС, Теле2, Yota, Beeline, Сбербанк, Газпром-Банк, Альфа-банк, Citymobil

## А что же mail.ru group

- Mail.ru – 50 репозиториев открытого кода - <https://github.com/mailru>
- Одноклассники построили одну из самых больших инсталляций Cassandra
- ОК - <https://github.com/odnoklassniki>
- VK – 49 репозиториев <https://github.com/VKCOM>

## Зачем Mail.Ru Group контрибьютит

- OpenSource очень волатильный. Контрибьюция - это способ следить за изменениями в проектах, наращивать экспертизу
- Hr PR. Не секрет, что на рынке IT специалистов есть проблемы, и как раз такой PR позволяет охватывать большую аудиторию с целью ее привлечения на работу в компанию
- Улучшения количества и качества собственных OpenSource продуктов за счет их массового применения
- Just For Fun

## Практический результат

- Приобретена огромная экспертиза практически по всем направлениям OpenSource проектов (от web до геораспределенных СУБД)
- Созданы свои собственные OpenSource продукты, которые пользуются популярностью
- Нас приглашают на конференции по всему миру. А это важный технический PR
- Высокий Time2Market (мы знаем все об: NGINX, PostgreSQL, Tarantool, ...).

## Проблемы OpenSource

- Как же без них!
- OpenSource-проекты ломаются, и порой неожиданно для всех. При этом поддержка — это твоя компания
- То что нужно твоей компании, как правило, не нужно OpenSource проекту
- Экспертизы на рынке много, но ее надо **хантить**
- OpenSource стоит денег (этот тезис раскрою ниже).

## Самые болезненные

- И это “Вторая проблема: то что нужно твоей компании как правило не нужно OpenSource проекту”
- Самые острые случаи (о том, как их решить будет ниже):
  - Бизнес требования могут противоречить функциональности OpenSource проекта
  - Фиксы каких-то проблем могут не принять в OpenSource-проект (причины могут быть совершенно разные)
  - \*Циклы релизов OpenSource-проекта могут отличаться от ваших.

\* Хороший пример: Не так давно OpenSSL выпустил патч который дошел до OpenSource операционных систем (CentOS, ...) с серьезным опозданием. Таких примеров было много в истории.

## И вместо заключения “Вивисекции OpenSource”

- OpenSource имеет свою цену.
- Приблизительно она формируется след. образом:
  - Затраты на обучения персонала для поддержки
  - Затраты на обучения персонала для “доработки”, разработки, пользования
  - Затраты на создания систем мониторинга
  - Прямые потери от “падений” OpenSource-проектов
- Из выше сказанного следует: использования OpenSource без определенной подготовки требует вложений, причем сравнимых с лицензионным ПО.

## С чего начать

- Выше мы выделили несколько проблем:
  - Экономическая — OpenSource стоит денег, и может даже переплюнуть лицензионное ПО
  - Практическая — OpenSource-проекты не обязаны следовать за вами и не обязаны поддерживать вас.
- Разберем эти проблемы подробно, начиная с “практической”, так как она прямо связана с Экономической



## Решение #1. Путь крупных Интернет корпораций

- Создать собственную команду(ы) которая будет внутри компании развивать \***“форк”** OpenSource проектов(ов)
- И/или создавать свой собственный OpenSource
- Минусы:
  - Долгое вложение средств\*\*
  - Сложности в планирование\*\*\*

\* Например: Mail.Ru Group имеет свои патчи/форки нескольких OpenSource проектов

\*\* на примере одной китайской компаний: 3 - 4 года на изучения проектов (NGINX, PostgreSQL) и их доработки. На примере Mail.Ru Group: развитие

\*\*\* Не каждый ПМ (или владелец продукта) может понять, что он хочет от OpenSource проектов в разных временных диапазонах

## Решение #2. Внешняя экспертиза из OpenSource проекта

- Команду можно взять в “аренду” с целями: обучить, помочь протолкнуть какие-то патчи в OpenSource проект, помочь запуститься
- Минусы:
  - Экспертиза будет постоянно зависит от “Внешней команды”, если исчезнет она, то и исчезнет большая часть экспертизы
  - Стоимость таких специалистов высока, и их не очень много на рынке.

## Решение #3. Enterprise-версии OpenSource проектов

- Купить \*Enterprise-версию OpenSource-проекта (продукта)
- Заказать обучение
- Купить поддержку
- Минусы:
  - Экспертизы остается в другой компании
  - Ежегодная оплата услуг и лицензии

## Экономическая проблема

- Любая попытка использовать OpenSource приведет к вложениям.
- Считать. Просчитайте вложения в ресурсы, сайзинги, возможные риски до того как вы начали использовать OpenSource
- Позвать умных. Приглашайте экспертов - их огромное количество на российском рынке, устраивайте митапы, технические сессии.
- Поумнеть самим. Вкладывайте в свою собственную экспертизу, это принесет прибыль в виде: улучшений в Time2Market, как минимум. А как максимум сделает вас менее зависимыми от разных вендоров (а за общение с ними надо платить).

## Как начать пользоваться OpenSource

Несколько правил:

- Нарращиваете экспертизу, не бойтесь привлекать вендоров на первых этапах (с целью принять экспертизу).
- OpenSource OpenSource'у рознь. Всегда оцениваете проект и его применимость под ваши задачи, а также обязательно узнайте RoadMap-проекта на несколько лет вперед
- Помните о том, что рано или поздно оно упадет, и вы должны иметь план на этот случай (вендор, внутренняя экспертиза, что угодно)
- Правильно формируйте цель на старте. Например: заменить А на Б — это не цель, а вот целью может быть: заменить А на Б, чтобы получить какую-то возможность сейчас или в будущем.

## Как начать пользоваться OpenSource

- Не бойтесь развивать отделы по модели Китая, или крупных интернет корпораций (ведь все слышали о Mac OSX, Chrome, Go, C/C++? — А это все прямое следствия применения таких моделей).

Однако, всегда помните, что такие отделы не могут существовать без продуктовогов, KPI, связи с бизнес заказчиком

# Вопросы?

[yermakov@corp.mail.ru](mailto:yermakov@corp.mail.ru)



@mail.ru  
group