

Управление расходами в ИТ.  
Практика применения Open Source  
Тимур Колёскин

# Будем ГОВОРИТЬ О...

01

Структура расходов на ИТ  
и система учёта ИТ-расходов

02

Прикладные сервисы и сервисы  
доступа

03

Технологические сервисы

04

Open Source

# Структура ИТ расходов

## Прикладные сервисы

*Приложения, которые формируют ценность для бизнеса*

## Технологические сервисы

*Сервера, сеть ЦОД, СУБД, Kubernetes и т.д.*

## Сервисы доступа

*ИТ инфраструктура офисных площадок, телефония, удалённый доступ и т.д.*

- **3 области управления в ИТ**  
*Должна быть какая-то тактика и лучше её придерживаться.*
- **Система учёта затрат на ИТ**  
*Невозможно эффективно управлять тем, что нельзя измерить!*
- **ИТ персонал**  
*Тоже расходы и главный актив!*

# Пишем стратегию управления расходами сами

- Когда уже можно считать эффективным ИТ?
- CapEx versus OpEx  
*Учётная политика творит чудеса*
- Адекватность потребностям бизнеса  
*Готов ли бизнес к Rocket Since*
- Работа с поставщиками/вендорами

3-6% дохода – расходы на ИТ

**IT Spending Ratios**  
Between 25th and 75th Percentiles, by Industry

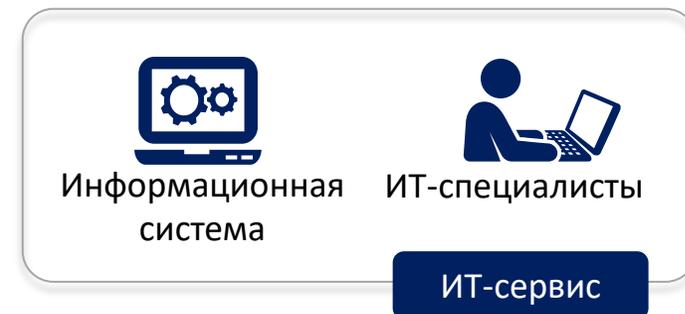
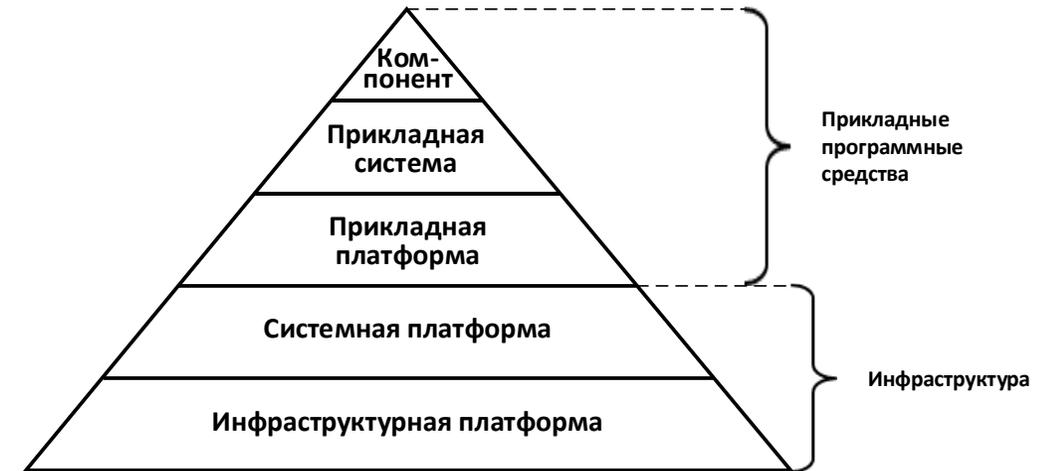
IT Spend as...	Discrete Mfg	Fin'l Services	High Tech	Retail	Health care
Percentage of Revenue	1.4%-3.2%	4.4%-11.4%	2.6%-4.7%	1.2%-3.0%	3.0%-5.9%
Per User	\$3,733-\$9,864	\$13,772-\$26,667	\$6,191-\$11,653	\$3,913-\$14,685	\$3,157-\$6,143
Per Desktop/Laptop	\$4,658-\$9,395	\$12,171-\$23,882	\$5,452-\$9,218	\$4,806-\$13,533	\$3,280-\$7,273

Source: Computer Economics, 2019

Figure 1

# Система учёта затрат в ИТ

- ИТ сервис – совокупность всех ассоциированных ресурсов (ПО, вычислительные ресурсы, ИТ-персонал, контракты)
- Создание точек контроля *этапы процедур внутри организации, которые невозможны без ассоциации действия с ИТ-сервисом (закупка, выделение выч. ресурсов, доработка и т.д.)*
- Умение прогнозировать затраты для определения ТСО на 3/5 лет.  
*Вы точно ничего не забыли посчитать?*
- Как расходы вяжутся с доходами? NPV?
- Без фанатизма, это не бух. учёт.



# Уменьшаем ТСО Сервисов доступа

- **Переход с коробок на OpenSource**
- **Управление лицензиями и мощностями**
- **Управление технологической архитектурой**
- **Переход на аутсорсинг/SaaS**

## ВАРИАНТЫ

Windows – Linux  
Office - OpenOffice

Enterprise Agreement  
Учёт, нормы оснащения

VDI, унифицированная  
среда коммуникаций

Полный аутсорсинг.  
Только поддержка и  
оргтехника  
Office 365, облачная  
телефония

## СЛОЖНОСТИ

Тяжело идёт

Нужно внедрять CMDB

VDI далеко не всегда что-то экономит

Сложность вывода на аутсорсинг  
Зависимость от чужой инфраструктуры  
Может быть дороже

# Уменьшаем ТСО Прикладных сервисов

- **Переход с коробок на собственную разработку**
- **Управление лицензиями**
- **Управление прикладной архитектурой**
- **Переход с OnPrem на SaaS**

## ВАРИАНТЫ

Уникальные бизнес процессы

Инвентаризация Enterprise Agreement

Создание роли корп. архитектора, инвентаризация функционала

Идеально для Орех модели  
Второстепенные прикладные сервисы

## СЛОЖНОСТИ

Нужно уметь в собственную разработку  
Бизнес не может столько ждать

Часть вендоров имеют очень сложную систему лицензирования

Перспектива 3-5 лет.  
Правильно vs быстро.

Полная зависимость от поставщика решения  
Никакой уникальности

# Уменьшаем ТСО Технологических сервисов

- **Переход с коробок на OpenSource**
- **Управление лицензиями и мощностями**
- **Управление технологической архитектурой**
- **Переход с OnPrem на PaaS**

## ВАРИАНТЫ

Oracle, MSSQL – Postgres  
Windows - Linux

Enterprise Agreement  
Политика управления  
лицензиями и  
мощностями

Максимально  
высокоуровневый PaaS

Облачная модель.  
Pay as You Go  
Идеально для OpEx

## СЛОЖНОСТИ

Не всегда расходы оправдывают экономию на лицензиях.  
Проще внедрить новое решение

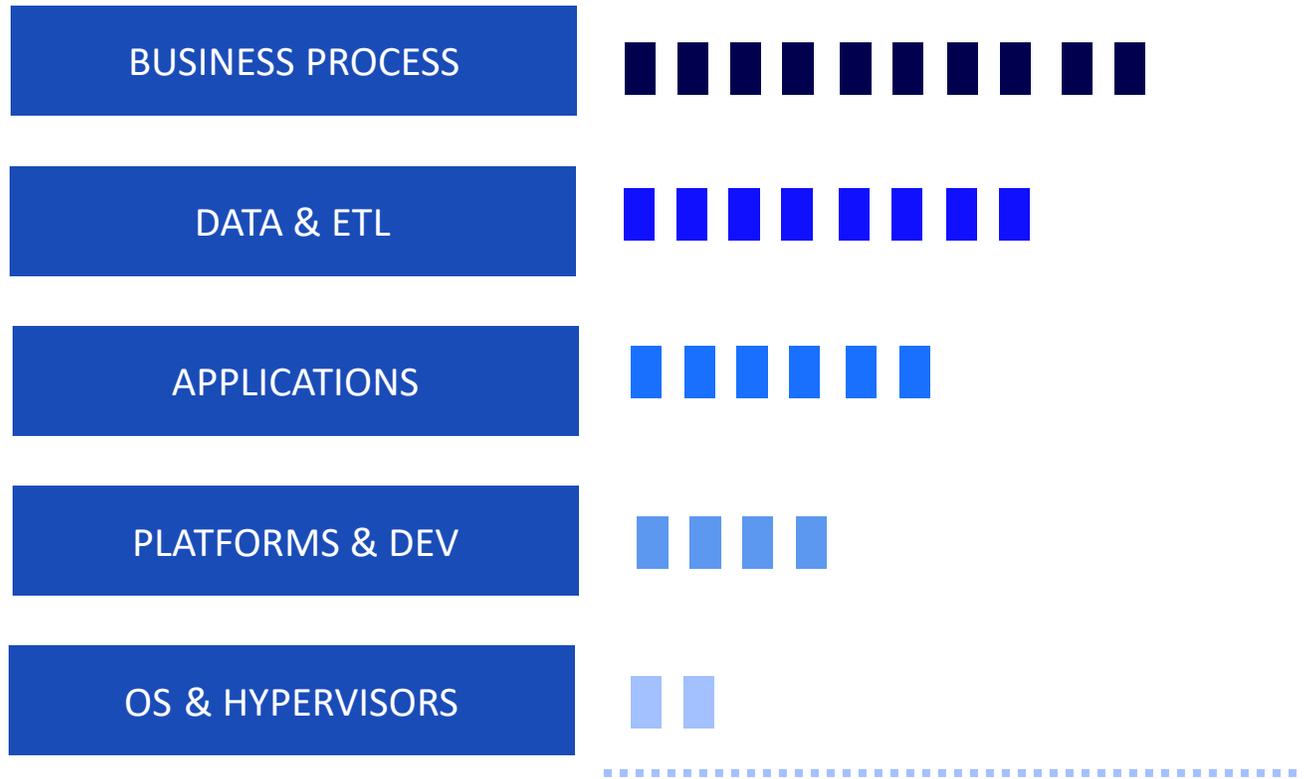
Часть вендоров имеют очень сложную систему лицензирования  
Сложно прогнозировать потребление мощностей

Приклад должен в это всё уметь интегрироваться.

Зависимость от чужой инфраструктуры  
Может быть дороже

# Место Open Source в ИТ архитектуре

## Вариативность

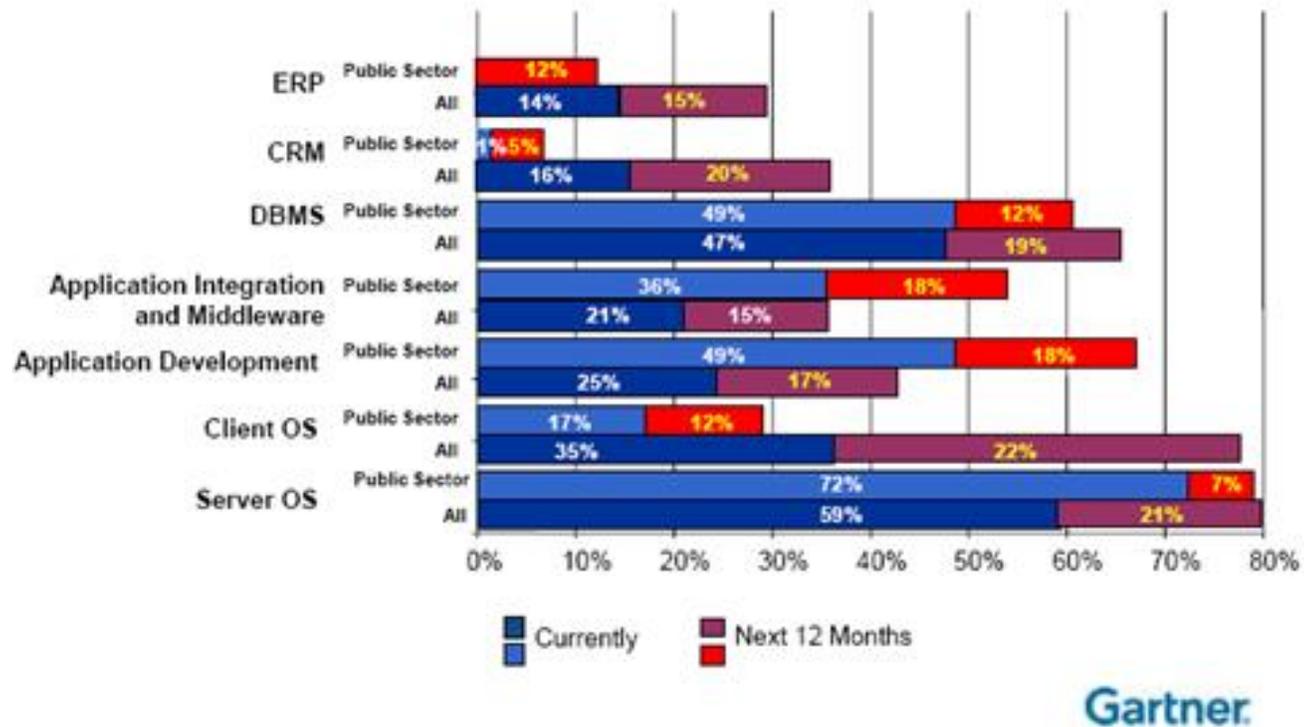


## GNU – GNU not UNIX

- **Open Source проекты зародились в низкоуровневой части ИТ архитектуры**  
Первый крупный проект под GNU - Linux
- **Успешный Open Source проект максимально универсален**  
После Linux – Apache, MySQL, PHP, Perl  
Позднее – СУБД, шины, языки программирования
- **Что за GPL, Apache, BSD, MIT ?**  
В этом придётся разбираться только если вы будете продавать софт, который разработан с использованием.

# Место Open Source в ИТ архитектуре

## Open Source Is Climbing the Stack



## Время одиночек прошло (почти)

- **Начинается как внутренний проект крупной компании/университета**  
LinkedIn — “Apache Kafka”  
University of California at Berkeley — “Apache Spark”, Postgres  
Cloudera — “Impala”  
Yahoo! — “Apache Hadoop”  
Google — “Kubernetes”  
Facebook — “Apache Hive”
- **Стартап поглощённый/развитый крупной компанией**  
Rabbit MQ, Redis, Greenplum
- **Проекты выросшие самостоятельно**  
Zabbix, Elasticsearch



# Преимущества и недостатки Open Source

## ХОРОШО

- Ниже стоимость владения при определенных условиях
- ИТ специалистам интереснее/перспективнее работать на OpenSource
- Можете создать команду, подхватить fork и развить в нужную Вам сторону
- Иногда хороший инструмент давления на проприетарных партнёров
- Вы можете быть большими, но вам может хватать Community Edition

## ПЛОХО

- Проект может свернуться
  - *потухнуть*
  - *купят и сделают платным с урезанным Community Edition*
  - *купят и пойдут не туда*
- Проект может подвести не сразу, после того как на него "пересели"
  - *архитектурные ограничения,*
  - *проблемы с производительностью*
  - *безопасность*
- Ваша ИТ-архитектура может быть не готова к этому

# Рекомендации по выбору Open Source

- Open Source должен быть вашей стратегией
- Определяем стадию жизненного цикла  
Смотрим в GitHub Insights
- Насколько Community Edition отличается от платной
- Наличие доп. услуг
- Какой проект интересен команде

Сильные команды разработки повышают долю OpenSource решений. Например в части СУБД

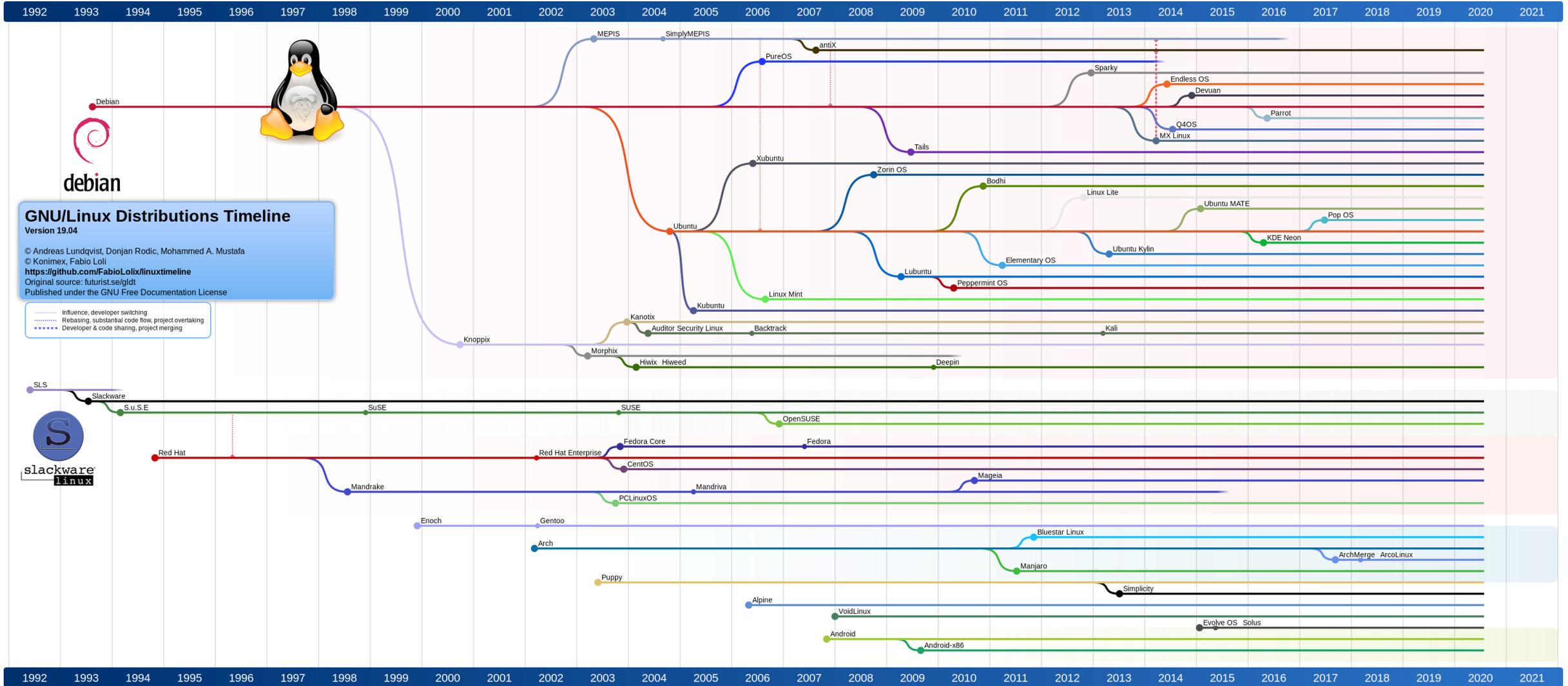
Gartner HypeCycle,  
Количество активных Contributor  
Количество форков  
Динамика Commits rate

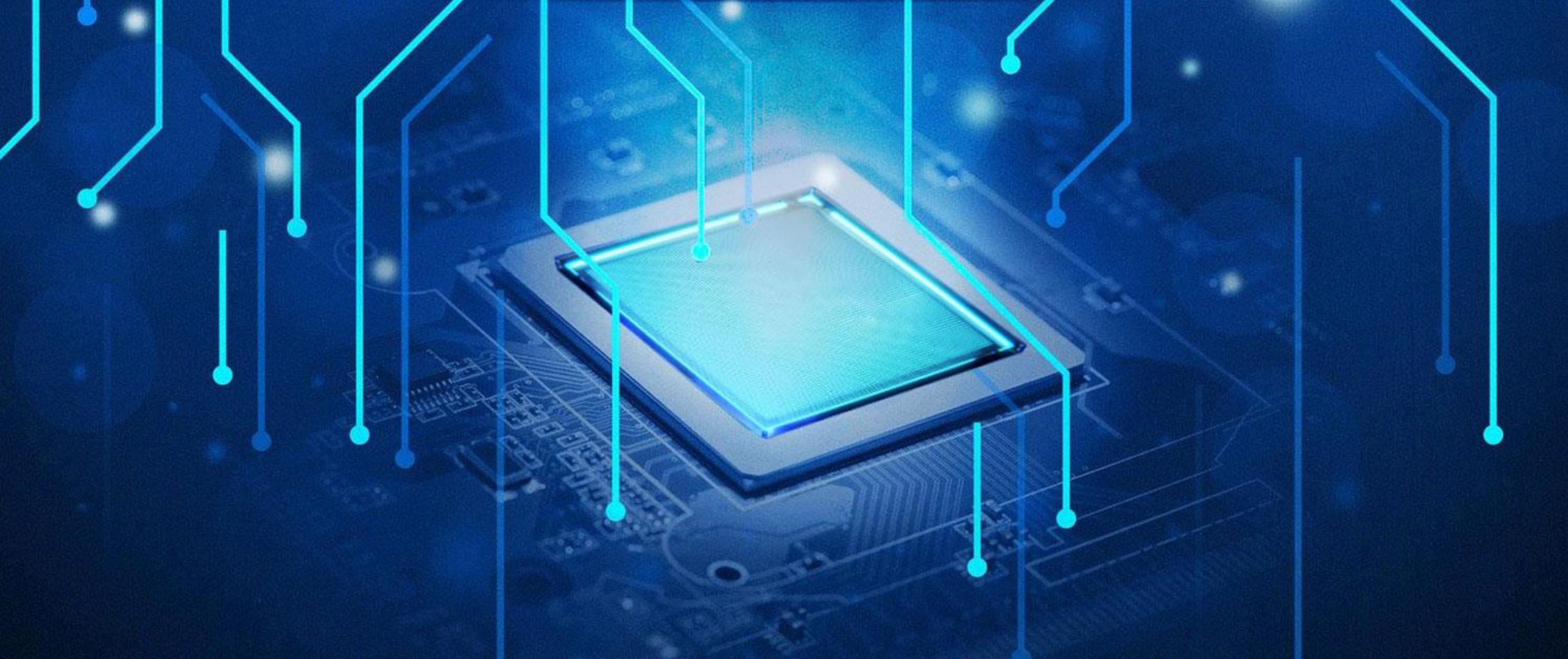
Монетизация проекта зачастую обеспечивается доп. услугами  
Платный более функциональный fork может заметно отличаться от "Community Edition"

Поддержка, аудит, интеграция, SaaS

Навязывать проект команде – такое себе

# Жизненный цикл Open Source на примере Linux



A glowing blue microchip is the central focus, resting on a circuit board. The chip and board are illuminated with a vibrant blue light, creating a futuristic and high-tech atmosphere. Numerous glowing blue lines and dots are scattered across the scene, some resembling circuit traces and others like data points or nodes. The background is a dark, deep blue, which makes the glowing elements stand out prominently. The overall composition is clean and modern, emphasizing technology and innovation.

**Спасибо за внимание!**