Портирование 11-летнего Windows приложения на Linux

Рустам Салимзибаров

Директор по разработкам в Macroscop

https://t.me/rustamsel

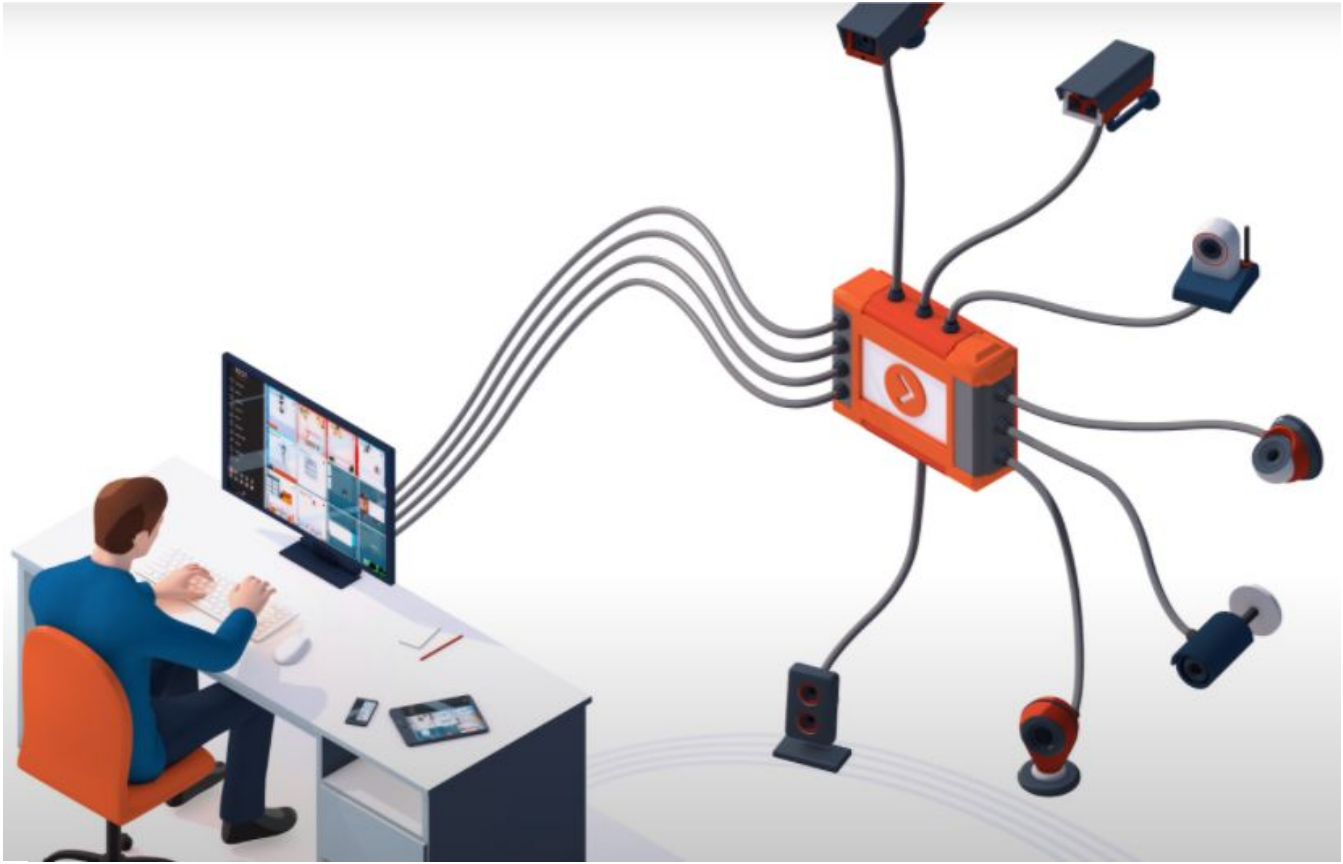
Алексей Поздеев

Тимлид команды Backend в Macroscop

https://t.me/Aleksey_Pozdeev

macroscop

Macroscop - система видеонаблюдения



macroscop

Основные приложения Macroscop

Десктопный клиент - *WPF / .NET FX 4.8*

Мобильные клиенты - *C# / Xamarin.Forms*

Web-клиент - *TypeScript / Angular*

Сервер видеонаблюдения - *C# / .NET FX 4.8*

macroscop

История портирования на Linux

- 2009г - появление Macroscop / .NET FX 3.5
- **2012г - портируем на Mono 2** FAILED

Цель - сэкономить на лицензиях Windows

Результат. Портировали за пару месяцев, т.к. в продукте было мало функций

Проблема 1: Зависал Macroscop или ОС при высокой загрузке ЦП
Проблема 2: Зависали бесконечные HTTP-подключения

Итого: Жалобы пользователей на стабильность и “заморозка” версии

***Совет**: Стабильность новой технологии в ваших кейсах - самое важное*

macroscop

История портирования на Linux

- 2009г - появление Macroscop
- ~~2012г - портируем на Mono 2~~ FAILED
- **2015-16г - портируем на Mono 4** FAILED

Цель - продавать софт для импортозамещения

Результат. Портировали 80% функционала за 1 год т.к. продукт за годы “раздулся” и на портировании не было фокуса

Проблемы. Сетевые проблемы в Mono решили с помощью пул-реквестов*, а позже форка Mono

Итого: За год сменился продукт менеджер, цель потеряла актуальность

***Совет**: Не затягивай портирование, иначе через год станет не актуально*

macroscop

* примеры пул-реквестов: https://github.com/mono/mono/pull/3483, https://github.com/mono/mono/pull/2272

История портирования на Linux

- 2009г - появление Macroscop
- ~~2012г - портируем на Mono 2~~ FAILED
- ~~2015-16г - портируем на Mono 4~~ FAILED
- **2016г - портируем на ARM** FAILED

Цель - сэкономить на железе с предустановленным софтом благодаря портированию софта на ARM-процессоры

Результат. Портировали за пару месяцев благодаря старым наработкам

Итого: Отдел продаж не смог продать железки с этими ARM-процессорами

***Совет**: Новая технология не является залогом финансового успеха*

ARM POWERED

macroscop

История портирования на Linux

- 2009г - появление Macroscop
- ~~2012г - портируем на Mono 2~~ FAILED
- ~~2015-16г - портируем на Mono 4~~ FAILED
- ~~2016г - портируем на ARM~~ FAILED

Была ли польза от попыток портирования на Mono?

- Выделили сборки для платформо-зависимого кода и ввели кроссплатформенные стандарты
- Осознали необходимость портировать быстро

***Совет**. Любая неудача делает вас сильнее и умнее*

macroscop

История портирования на Linux

- 2009г - появление Macroscop
- ~~2012г - портируем на Mono 2~~ FAILED
- ~~2015-16г - портируем на Mono 4~~ FAILED
- ~~2016г - портируем на ARM~~ FAILED
- **2016г - портируем на .NET Core 1** FAILED

Цель - ради любопытства, вдруг получится

Результат. Не было встроенной бинарной сериализации и ничего не билдилось

Совет. *Будь любопытен! 4 часа работы с исходниками дадут тебе больше, чем месяцы чтения статей!*

macroscop

История портирования на Linux

- 2009г - появление Macroscop
- ~~2012г - портируем на Mono 2~~ FAILED
- ~~2015-16г - портируем на Mono 4~~ FAILED
- ~~2016г портируем на ARM~~ FAILED
- ~~2016г - портируем на .NET Core 1~~ FAILED
- **2019г - прототип .NET Core 2 за выходные** ✓

Цель - проверить не врет ли реклама, что .NET Core 2 стал хорошо совместим с .NET Framework и даже появилась встроенная бинарная сериализация

Результат. Сел утром в субботу, а к вечеру воскресенья уже “билдилось”

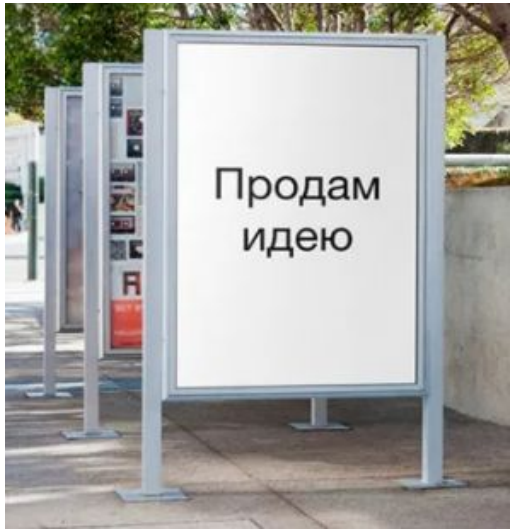
***Совет**. Прочел рекламную статью - не думай, а бери и ~~“билди”~~ пробуй*

macroscop

Как мы продали бизнесу идею переход на .NET Core?

Плюсы для бизнеса:

- Облачная версия продукта
- Клиенты просят Linux
- Экономия на лицензиях Windows
- Мотивация разработчиков благодаря .NET Core



***Совет**. Хочешь внедрить новую технологию - учись “продавать” идею бизнесу*

macroscop

История портирования на Linux

- 2009г - появление Macroscop, .NET FX 3.5
- ~~2012г - .NET FX 4.0, портируем на Mono 2~~ FAILED
- ~~2015-16г - портируем на Mono 4, .NET FX 4.5~~ FAILED
- ~~2016г портируем на ARM (Mono 4)~~ FAILED
- ~~**2016г - портируем на .NET Core 1**~~ FAILED
- 2019г - прототип .NET Core 2 за выходные
- Далее
  - Доработали прототип, оценили сроки (4 месяца команды “Бэкенд”)
  - Окончательно договорились с бизнесом об условиях:
    - Портирование разбиваем на 2 релиза
    - Полная совместимость Linux и Windows-версии
  - С августа 2019г - активная фаза разработки в команде Алексея

А если сборки нет в .NET Core?

- Наш кейс - WindowsBase сборка
- По хорошему - не использовать такие классы\сборки
- По быстрому, для решения проблемы с бинарной сериализацией:
  - создать классы с такими же названиями
  - с такими же неймспейсами
  - реализовать кастомный SerializationBinder

```
var bf = new BinaryFormatter();
bf.Binder = new NetCoreSerializationBinder();
bf.Serialize(stream, obj);
```

А если сборки нет в .NET Core?

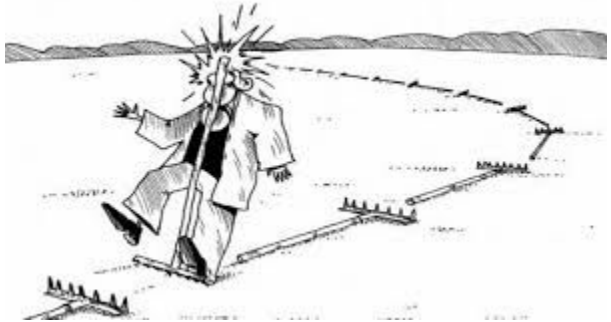
- По быстрому всю сборку:
  - дизассемблировать сборку (например, через .NET Reflector)
  - скопипастить оттуда все классы
  - скомпилировать сборку под .NET Core

Есть еще проблемы с бинарной сериализацией?

- **AssemblyVersion** в .NET FX должна быть не выше, чем в .NET Core
- .NET FX 4.6 на клиенте, .NET Core 2.1 на сервере - тоже ломается.
- Правильно - не использовать бинарную сериализацию
- Хардкод **AssemblyVersion** и обновление runtime до .NET FX 4.8.
- **Совет:** *Ищите простые решения!*

А еще проблемы с сетью были?

- HttpListener не поддерживает HTTPS на Linux в .NET Core*
- Правильно - использовать Kestrel
- Быстро - nginx, а Macroscop только по HTTP
- OWIN для WebSocket соединений - пришлось еще и Kestrel завозить.
- **Совет:** *Используйте кроссплатформенные библиотеки!*

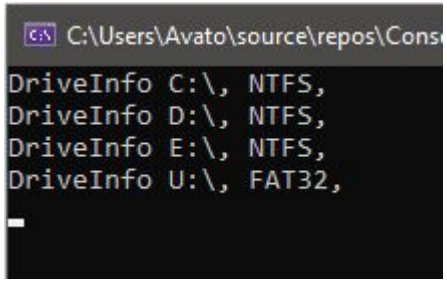


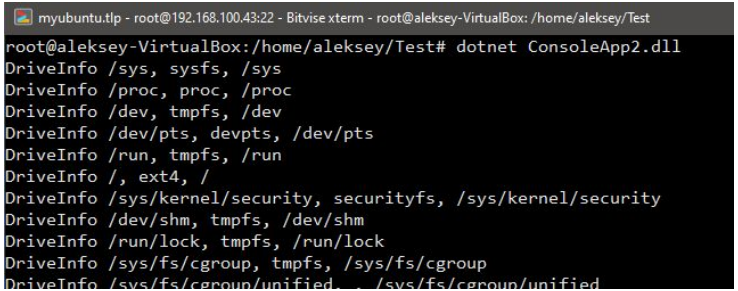
macroscop

* https://github.com/dotnet/corefx/issues/14691, https://github.com/dotnet/platform-compat/issues/88

А что если метод работает не как в Windows?

- DriveInfo.GetDrives() - на Linux выдаёт точки монтирования





- Консольные команды решают
- Прописали в .service файле Environment=LANG=en_US.UTF-8
- **Совет:** *Ищите обходные решения!*

А что если класса нет в .NET Core?

- Счетчики производительности PerformanceCounter
- Реализовали свои для Linux
- В .NET Core 3.0 рекомендуется использовать EventCounter
- **Совет:** *Если чего-то не хватает - попробуйте реализовать это сами!*

А что если метод не реализован в .NET Core?

- Thread.Abort
- PlatformNotSupportedException в .NET Core
- Правильно - не использовать Thread.Abort вообще
- Переделали на CancellationToken в Linux
- **Совет:** *Не стоит отчаиваться!*

А что если не работает целая подсистема?

- WCF - у нас есть и клиентский и серверный.
- С клиентским повезло - у нас все заработало
- Серверный - есть nuget пакет *(https://github.com/CoreWCF/CoreWCF)*
- **Совет:** *В общем надо пробовать - вдруг и вам повезёт!*

А что если не работает целая подсистема?

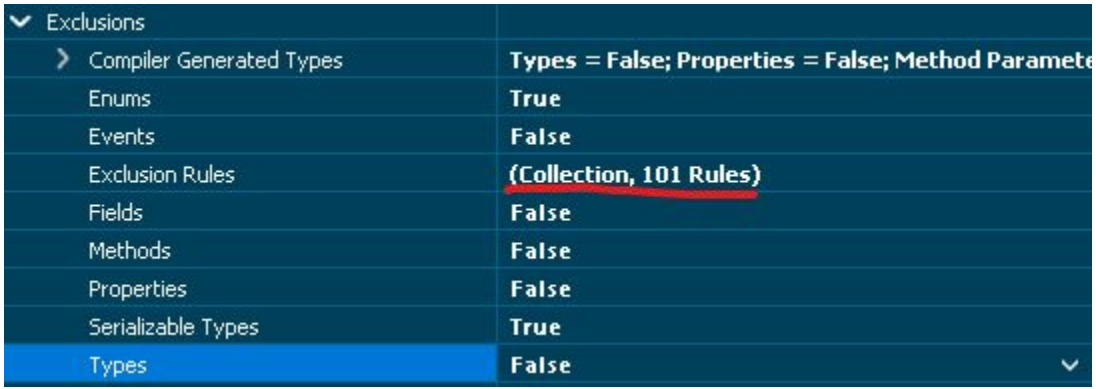
- ActiveDirectory - вначале вообще выпустили без него
- Самый популярный пакет Novell.Directory не заработал
- Попробовали несколько других пакетов - в итоге нас устроил LdapForNet
- **Совет:** *Если сразу не срослось - нужно еще немного поискать!*

Ну всё, продакшн?

- OutOfMemoryException при длительной работе на виртуалках
- Обновились до 3.0_alpha и, о чудо, перестало падать!
- С релизом 3.0 можно в продакшн!

Обфускация

- Большой продукт, 120+ проектов
- Перепробовали 20 обфускаторов и никто не мог обфусцировать!
- .NET Reactor - слабо обфусцировал в прототипе, но смогли настроить



Итоги портирования на Linux

- авг 2019 - начата разработка Линукс-версии
- мар 2020 - Linux версия с базовым функционалом

Итоги портирования на Linux

- авг 2019 - начата разработка Линукс-версии ✓
- мар 2020 - Linux версия с базовым функционалом ✓
- ноя 2020 - Linux версия с 99% функционала ✓

macroscop

Итоги портирования на Linux

- авг 2019 - начата разработка Линукс-версии
- мар 2020 - Linux версия с базовым функционалом
- ноя 2020 - Linux версия с 99% функционала
- уже сейчас - развитие облачной версии

Итоги портирования на Linux

- авг 2019 - начата разработка Линукс-версии ✓
- мар 2020 - Linux версия с базовым функционалом ✓
- ноя 2020 - Linux версия с 99% функционала ✓
- уже сейчас - развитие облачной версии ✓
- ближайшее будущее - переход на .NET 5

Самое важное

- Проявляй любопытство
- Проверяй стабильность
- Внедряй быстро
- Продай идею “бизнесу”

- Пишите свои реализации
- Соблюдайте кроссплатформенность
- Ищите сторонние решения
- Не сдавайтесь, пробуйте и всё получится!

Рустам Салимзибаров

https://t.me/rustamsel

Алексей Поздеев

https://t.me/Aleksey_Pozdeev

macroscop