



Как IT-стартапу стать бизнесом: выстраиваем конвейер разработки

Долбилов Михаил, CEO Itgrix
Атангулов Роман, разработчик

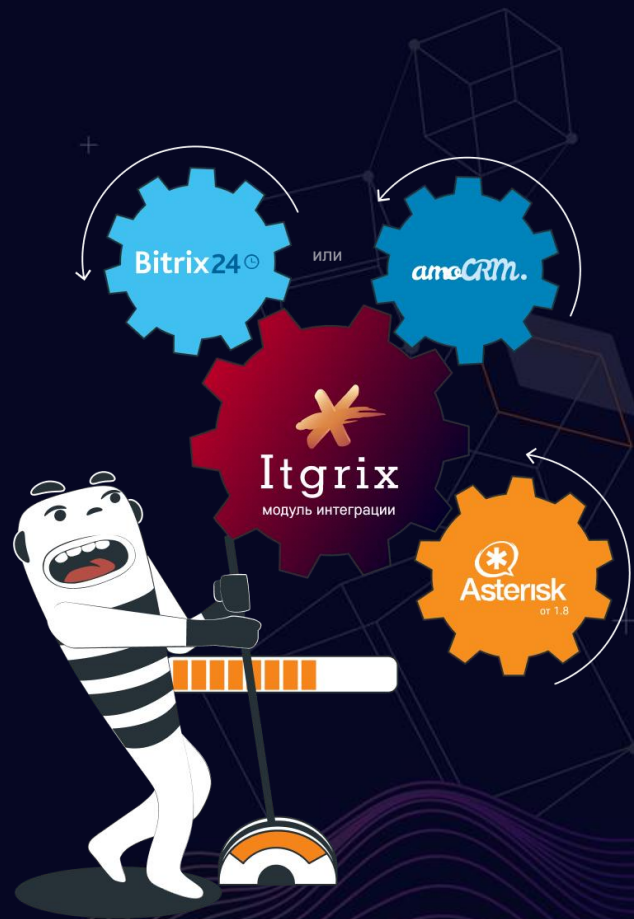
Пара слов про себя



Itgrix – Asterisk-коннектор для CRM (Битрикс24, amoCRM)

О коннекторе в цифрах:

- 17 стран, где уже используют Itgrix
- >100 партнеров в России и СНГ
- 5 место в номинации “Лучший автор Приложений24”
- 95% – индекс удовлетворенности клиентов



От программы к продукту



Разработка софта – довольно просто,
но когда начинаешь делать бизнес/продукт, начинаются приключения.

Реальность – баги, ограниченность ресурсов.

Инвестиции во внутренние продукты – необходимость.

Ради выпуска стабильного продукта:

- Система версионирования кода и автоматической сборки версии
- Система юнит-тестирования и тестирования бизнес-логики перед релизом
- Система доставки изменений и обновления продукта
- Коммуникация с клиентами и получение обратной связи для быстрой адаптации



Репозиторий




Система контроля версий, например, git:

- Быстрый доступ к истории кода
- Одновременная работа нескольких разработчиков
- Параллельная работа над разными версиями продукта
- GitLab позволяют делать code-review через Merge Request



Репозиторий



bx: v3.5.0: Новое: * Добавлен supervisor с функциями обновления и баг-репорта Админка: * Добавлена ссылка на пользователя
Merge branch 'development'

supervisor: v1.0.0: * Начальный релиз
Merge branch 'front_features' into development
Merge remote-tracking branch 'origin/70706-ff-update_by_supervisor' into front_features
Merge branch 'remove_testing_flag' into '70706-ff-update_by_supervisor' [origin/70706-ff-update_by_supervisor](#)

bx: v3.5.0-rc.3: Отдельный скрипт бэкапа * Не-special версия для теста обновления через админку
Merge branch 'supervisor-scripts' into development

В рамках Task #71335: * Добавлен скрипт `make_backup.sh`. * `install.sh`: * Убрана реализация [origin & supervisor-scripts](#)

bx: v3.5.0-rc.2: * Обновление `sudoers` с учётом расширения бэкапов
Merge branch 'front_features' into development
Merge remote-tracking branch 'origin/hotfix' into development
Merge remote-tracking branch 'origin/supervisor-scripts' into development
Merge branch '69033-ff-no_subs_app_hide_timeman_check' into 'hotfix'

В рамках Task #71335: `install.sh`: * Резервное копирование расширено: в копию добавлены: * все конфиги `logrotate`
`ap_bx`: change `app_type` in store after success load token [origin/69033-ff-no_subs_app_hide_timeman_check](#)

Merge branch 'development' into front_features
Merge remote-tracking branch 'origin/70706-ff-update_by_supervisor' into front_features

bx: v3.5.0-rc.1 * Лого в админке - ссылка на статусы * Исправлено логирование ошибки при успешной авторизации * Время
Merge branch 'hotfix' into 'development'

Автоматизация сборки релиза



В релиз входит:

- Исполняемый файл (itgrix_bx, itgrix_amo)
- Файлы веб-интерфейса для настройки (отдельный подпроект для фронта)
- Скрипты установки, обновления, удаления (install.sh, update.sh, remove.sh...)
- Кастомизатор
- Конфиг по умолчанию

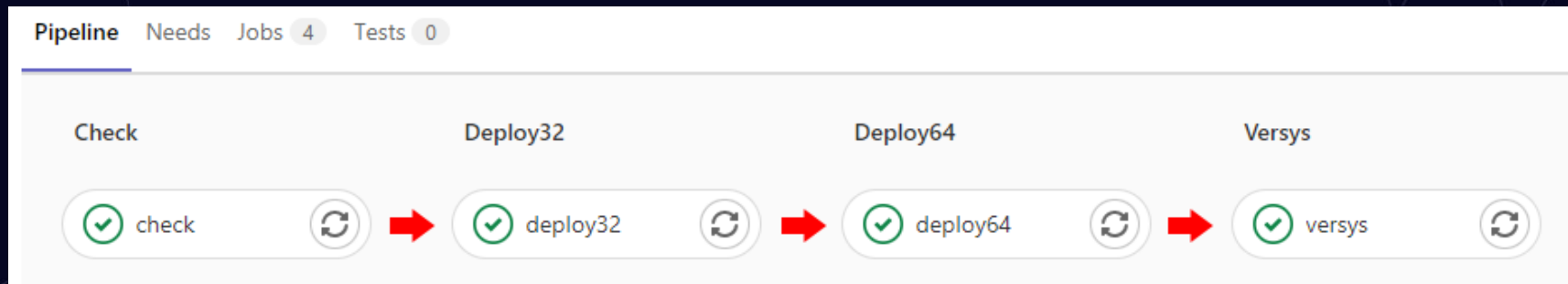
build.sh



itgrix_bx-3.5.1-amd64.tgz

Сборочный конвейер – GitLab CI/CD

- Основная разработка в ветке **development**
- В репозитории специальные ветки для конвейерной сборки: **dev, beta, master**



Коммуникация с клиентами



По мере роста количества клиентов **техподдержка** становится необходимостью:

- Сильно разгружает разработчиков по времени
- Снижает количество отвлечений разработчиков => снижается вероятность ошибок
- Повышает качество работы с клиентами



Техподдержка

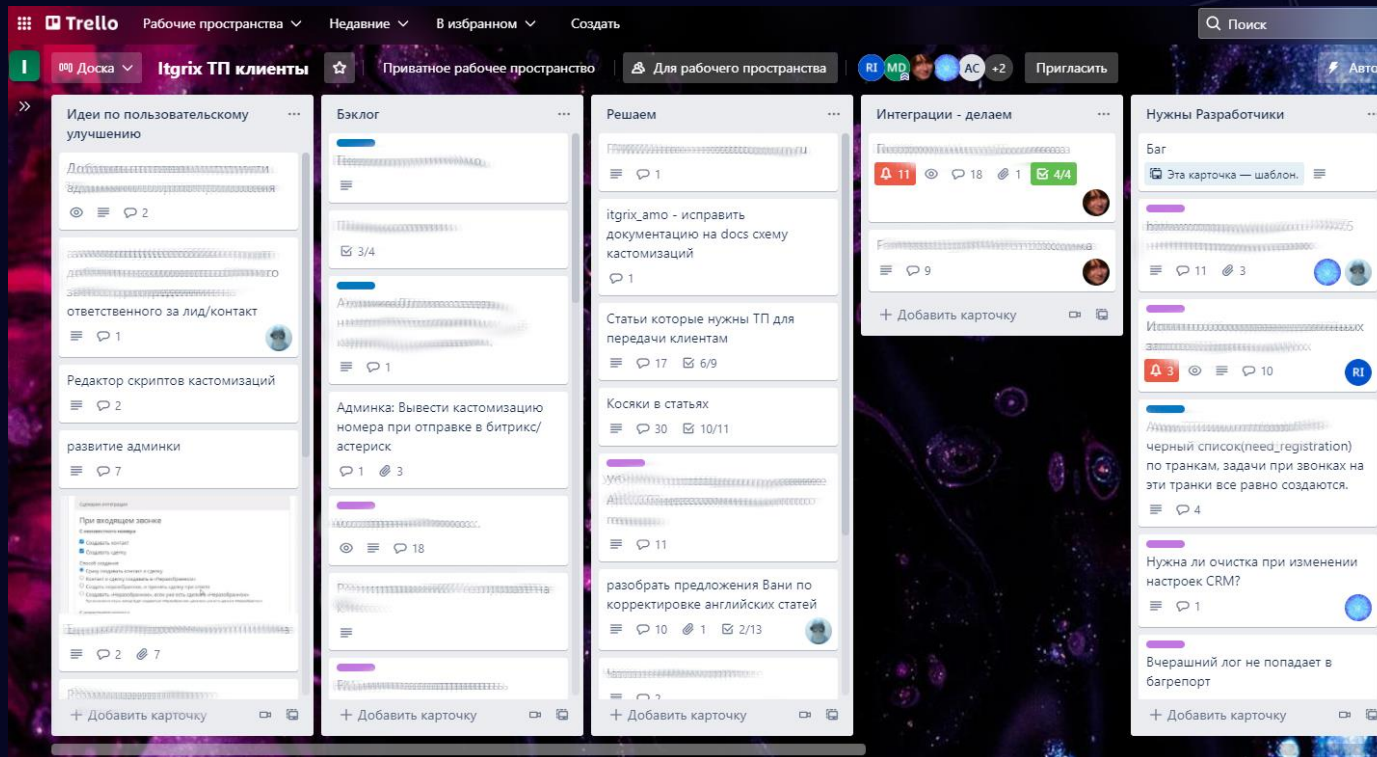


Трекер задач, багрепортов и идей – Trello:

- Удобно и бесплатно для стартапа
- Полная картина по каждой задаче/клиенту
- Позволяет не терять клиентские задачи – как новые, так и все архивные
- Можно фиксировать идеи для новых фич



Техподдержка



Техподдержка <+> разработчики



Польза для ТП:

- Консультации от разработчиков по сложным кейсам
- Текущий статус по разработке (планы по багфиксам, фичам)

Польза для разработчиков:

- ОС от клиентов. Видны проблемы и направления доработок

Польза для Product owner:

- Корректировка плана разработки (быстрая правка багов и клиентские фичи)
- Идеи по развитию продукта



Анализ обратной связи



Распространенные запросы клиентов в техподдержку:

- В целом модуль работает, но в каком-то звонке что-то не так (не зарегистрировался, не тот ответственный и т.п.)
 - нужно смотреть запросы в CRM
- Обновился, частично работает, работает неправильно
 - конфиг изменился в новой версии, часть настроек не применяется, нужно обновить конфиг



Тестирование



- Модульное тестирование (юнит-тесты) – проверка исходного кода
- Системное тестирование (тестовый запуск готовой сборки) – проверка бизнес-логики



Юнит-тесты



Проверяют отдельные части кода – можно быть уверенным
в каждом отдельном элементе (функции, классе)

```
✓ Tests passed: 8 of 8 tests – 10 ms
=== RUN   TestAmoConnector_New
--- PASS: TestAmoConnector_New (0.00s)
=== RUN   TestAmoConnector
--- PASS: TestAmoConnector (0.01s)
----- PASS: TestAmoConnector/not-connected (0.00s)
----- PASS: TestAmoConnector/no-params (0.00s)
----- PASS: TestAmoConnector/with-params (0.00s)
----- PASS: TestAmoConnector/error-199 (0.00s)
----- PASS: TestAmoConnector/error-402 (0.00s)
----- PASS: TestAmoConnector/error-401 (0.00s)
PASS
```



Тестирование: бизнес-логика



Внутренний продукт TestSystem:

- Системное тестирование – проверка готовой сборки приложения в его реальном рабочем окружении (по принципу “черного ящика”)
- На входе данные о звонках, на выходе перехватываем и проверяем запросы в API CRM



Тестирование: бизнес-логика



Структура тестов:

- Один тип звонка – один тест
- Разные конфиги (настройки бизнес-логики) – разные наборы тестов
- itgrix_bx: **7 наборов**, суммарно **18 тестовых звонков**
itgrix_amo: **8 наборов**, суммарно **23 тестовых звонка**

С помощью TestSystem до релиза уже не допущено несколько серьезных багов.

Например, из последнего: при релизе в itgrix_amo неправильно назначался ответственный за звонок.



TestSystem: технологии

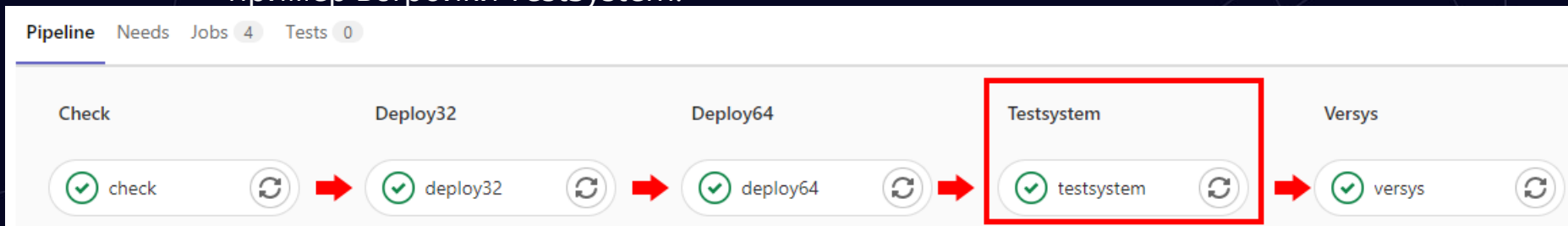


- Три docker-контейнера:
 - имитатор Asterisk с данными тестовых ЗВОНКОВ
 - имитатор CRM – программа сравнивает каждый запрос в CRM с эталоном
 - сама сборка коннектора Itgrix
- Bash-скрипты для запуска всех наборов тестов



Автоматизация тестирования

- Можно встраивать через git-hook
 - выполнять юнит-тесты перед пушем
- Можно встраивать в конвейер
 - пример встройки TestSystem:



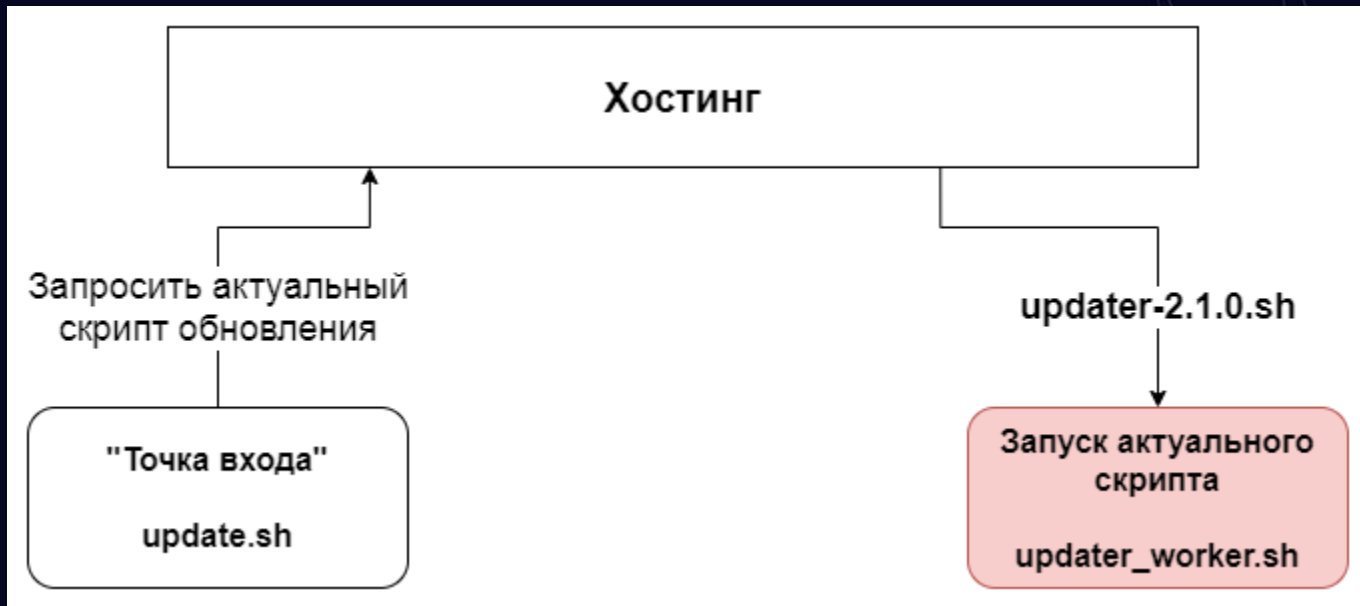
Обновление v1: это НЕ переустановка



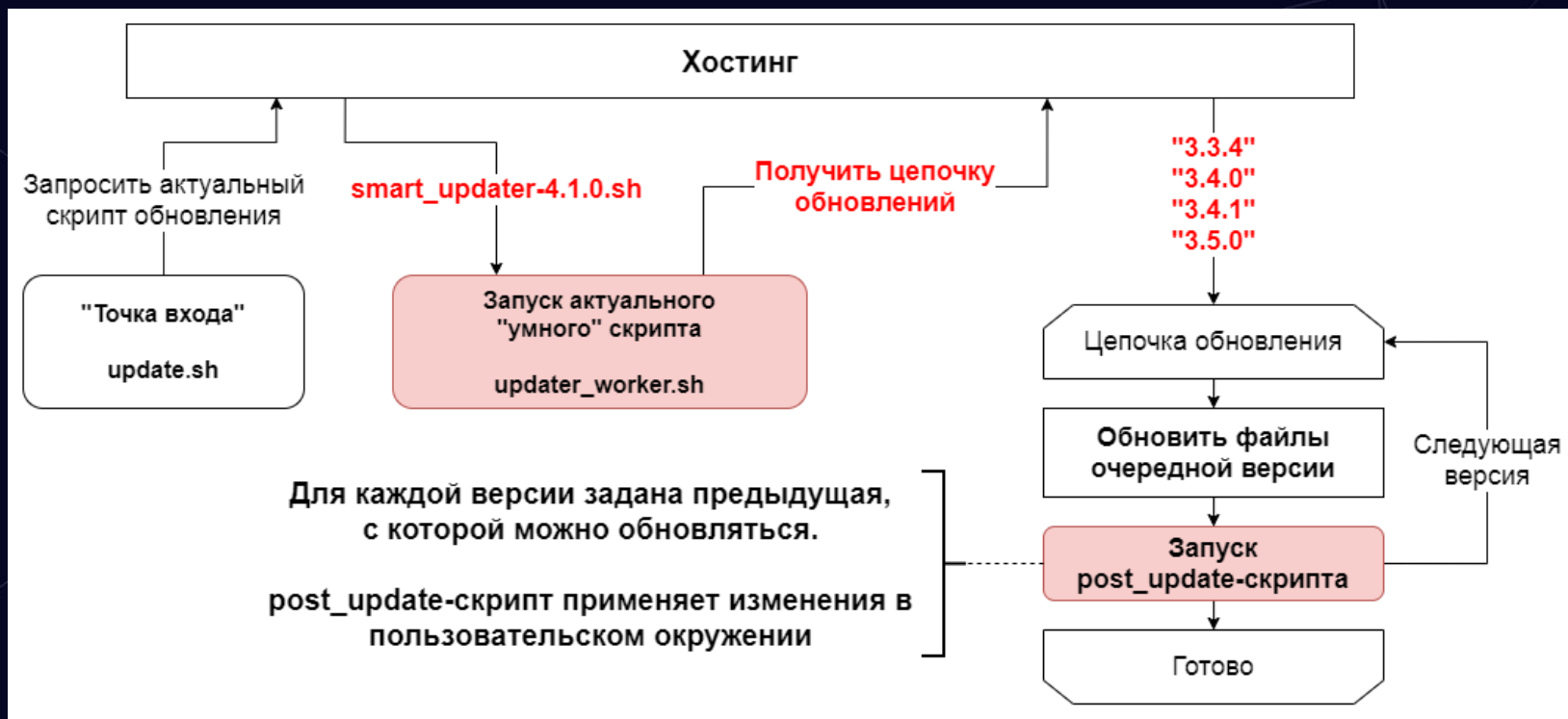
- Изначально считаем установку и обновление разными действиями – нужно сохранить пользовательские настройки
- Отдельно скрипт установки и скрипт обновления
- Достаточно простого скрипта на bash



Обновление v2: два этапа



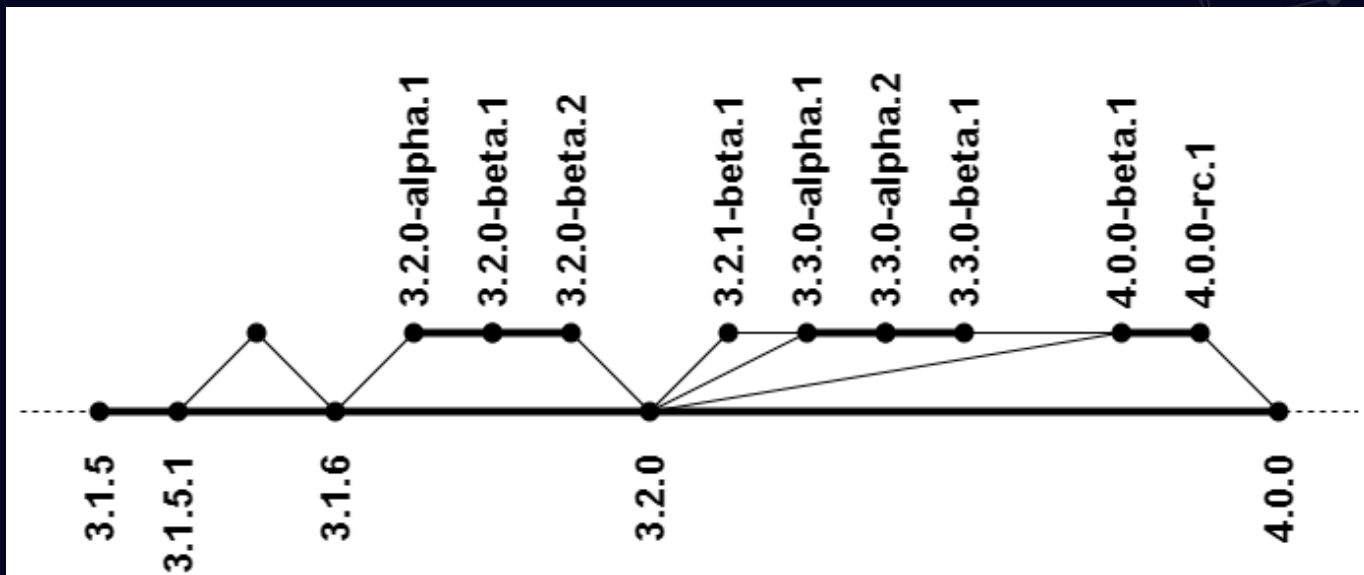
Обновление v3: "умное"



Обновление v3: система версий



Вспомогательный внутренний продукт – система версий VerSys.
Нужна для построения цепочки обновления.



Резюме



Как наладить конвейер разработки:

- Где и как работать с кодом – система контроля версий (git, GitLab)
- Автоматизация сборки – конвейер GitLab CI/CD
- Автоматическое тестирование перед релизом
- Доставка обновлений клиентам – “умное” обновление

Выводы



Для успешной поддержки и развития собственного продукта:

- На старте стоит задуматься о технической организации работы разработчиков
- По мере роста числа клиентов нужно инвестировать ресурсы на внутренние продукты и выделение технической поддержки



Спасибо за внимание!

Itgri

У есть ли вопросы?

dma@itgro.ru
+7 (343) 272-30-33
+7 (499) 350-88-90
<http://itgrix.ru/>

