



Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

Импортозамещение ESB: переводим интеграции на 1С:Шину

Миграция с западной ESB на 1С:Шину 4.1 / Datareon
Platform 3.0: аудит потоков, каналы, монитор...

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

Суть проблемы

Западные ESB и брокеры (BizTalk, MuleSoft, IBM MQ/IIB) остались без лицензий и обновлений: обмены 1С, сайта, склада и банка держатся на замороженной шине или россыпи прямых интеграций. Мы переводим контур на поддерживаемую российскую платформу — 1С:Шину или Datareon — без остановки бизнеса; иначе первый же внезапный отказ шины без вендора парализует отгрузки, оплаты и обмен да...

Почему это важно бизнесу

- Отказ неподдерживаемой шины останавливает отгрузки, оплаты и обмен 1С↔сайт↔банк; эскалировать некому — вендор ушёл с рынка
- Продлить западные лицензии нельзя: работа на замороженных версиях — незакрываемые уязвимости и риски при проверках
- Прямые обмены «каждый с каждым» растут квадратично: каждая новая система — недели доработок вместо одного канала на шине
- Платформа из реестра российского ПО закрывает требования импортозамещения в контрактах с госсектором и крупными заказчиками

Ключевые параметры реализации

4.1

редакция 1С:Шины в наших внедрениях: сервер на Linux, данные в PostgreSQL, не в файловой СУБД

по докам 1С:Шина 4.1

AMQP 1.0

протокол каналов к RabbitMQ; рядом — Kafka, JMS (ActiveMQ Artemis), JDBC, SOAP из коробки

по докам 1С:Шина

3.0.x

ветка Datareon Platform (.NET Core, реестр росПО) для ландшафтов без ядра 1С

по докам Datareon Platform

3-5

поток в пилоте: обкатка каналов, трансформаций и мониторинга до переноса критичных обменов

наш стандарт миграции

60 с

интервал опроса метрик шины в Zabbix; триггер — рост очереди недоставленных 15 мин подряд

наш стандарт мониторинга

100/500/∞

градации лицензий 1С:Шины; считаются только пользователи систем на платформе 1С:Предприятие

по докам 1С (v8.1с.ru)



Аудит: реестр интеграционных потоков и выбор платформы

Что настраиваем

Весь контур обменов: базы 1С, сайт, WMS, банк-клиент, EDI; работаем на сервере 1С и узлах старой шины

Как мы это делаем

- 1 Снимаем инвентарь: планы обмена и HTTP/веб-сервисы из конфигураций 1С, очереди и порты старой шины, cron/планировщики с файловыми обменами
- 2 Заполняем наш реестр потоков: источник, приёмник, формат (EnterpriseData/XML/JSON/CSV), объём в сутки, допустимая задержка, ответственный
- 3 Классифицируем: критичные (оплаты, отгрузки) и фоновые (справочники, отчёты); для критичных фиксируем RPO/RTO обмена
- 4 Выбираем платформу по матрице: ландшафт вокруг 1С → 1С:Шина 4.1; гетерогенный .NET/веб → Datareon Platform 3.0; потоковая телеметрия → каналы Kafka

РЕЗУЛЬТАТ

Ещё до покупки лицензий понятно, сколько каналов и трансформаций реально нужно: обычно треть прямых обменов — мёртвые, их не переносим. Смета и срок миграции считаются по реестру, а не «на глаз».

КЛЮЧЕВОЙ НЮАНС

Лицензирование смотрим до выбора: градации 1С:Шины — 100/500/без ограничений, причём считаются только пользователи подключаемых систем на платформе 1С:Предприятие, сторонние системы не считаются — на ландшафте...



Развёртывание 1С:Шины и перенос потоков

Что настраиваем

Отдельная Linux-ВМ под сервер шины + PostgreSQL; среда разработки — на машине инженера

Как мы это делаем

- 1 Ставим сервер 1С:Шины на Linux-ВМ (2 vCPU/8 ГБ на старт), хранилище переключаем с файловой СУБД на PostgreSQL — под нагрузку и нормальные бэкапы
- 2 В среде разработки собираем проект: каналы (HTTP-сервисы, SOAP, AMQP 1.0 к RabbitMQ), процессы маршрутизации и трансформации сообщений
- 3 Базы 1С подключаем через сервисы интеграции платформы — обмен идёт штатным механизмом, без доработок конфигураций
- 4 Пилот на 3–5 фоновых потоках параллельно со старой шиной, сверка сообщений; затем домен за доменом переносим критичные и гасим старые маршруты

РЕЗУЛЬТАТ

Новая интеграция — это канал и процесс в проекте шины, а не доработка каждой из систем: типовой обмен вводим за 1–2 дня. Гарантированная доставка из коробки: сообщение лежит в персистентном хранилище, пока приёмник не подтвердит получение.

КЛЮЧЕВОЙ НЮАНС

Трансформации и маппинг держим в процессах шины, а не в коде баз 1С: иначе логика размазывается по системам и при очередном обновлении конфигурации обмен снова ломается.



Эксплуатация: мониторинг очередей и регламенты

Что настраиваем

Zabbix 7.0 LTS на нашей площадке мониторинга + бэкап хранилища шины на внешний репозиторий

Как мы это делаем

- 1 Снимаем метрики шины (число сообщений, статистика по каналам) HTTP-агентом Zabbix раз в 60 с; триггеры — рост очереди 15 мин и канал без трафика
- 2 Внешние HTTP-каналы закрываем TLS и публикуем через reverse-proxy; каждой подключённой системе — своя учётка с минимальными правами
- 3 PostgreSQL шины — в ежедневный pg_dump на внешний репозиторий; восстановление проверяем раз в квартал
- 4 Обновления шины катим сначала на стенд с копией проекта и пилотными потоками, затем прод — в согласованное окно

РЕЗУЛЬТАТ

Зависший приёмник виден по алерту через 15 минут, а не по звонку бухгалтерии через сутки. После аварии шина поднимается из бэкапа вместе с очередью недоставленных сообщений — обмены доезжают сами, без ручного перепроведения.

КЛЮЧЕВОЙ НЮАНС

Гарантированная доставка — это ещё и растущий диск: при лежащем приёмнике хранилище копит сообщения, поэтому отдельно мониторим свободное место тома шины с порогом 15 %.



Подводные камни

✗ Перенос point-to-point «как есть»

Старые прямые обмены тащат на шину один в один — получается та же лапша. Сначала нормализуем: канонический формат и маршрутизация на шине.

✗ Файловая СУБД шины в проде

По умолчанию хранилище файловое; под потоком сообщений деградирует и плохо бэкапится. Сразу переключаем на PostgreSQL и включаем pg_dump.

✗ Kafka вместо ESB

Брокер даёт транспорт, но не трансформации и маршрутизацию — их пришлось бы писать самим. Kafka берём как канал шины, а не как её замену.

✗ Очередь растёт молча

При лежащем приёмнике гарантированная доставка копит сообщения до заполнения диска. Ставим триггеры на глубину очереди и свободное место тома.

✗ Логика обмена в конфигурациях 1С

Маппинг в коде баз ломается при каждом обновлении конфигурации. Трансформации живут в процессах шины, базы отдадут штатный EnterpriseData.

✗ Лицензии посчитаны после проекта

Градации 1С:Шины — 100/500/без ограничений; считаются пользователи систем на платформе 1С. Пересечение порога после внедрения — незапланированный бюд...

✗ Big-bang миграция за выходные

Единовременное переключение всех потоков не оставляет пути отката. Ведём параллельный прогон со сверкой сообщений и переносим по доменам.

✗ Обновление шины без стенда

Апдейт сразу на проде ломает каналы в рабочее время. Держим стенд с копией интеграционного проекта и пилотным набором потоков.

Как правильно

МИНИМУМ

- Реестр всех интеграционных потоков с владельцами, форматами и объёмами
- Шина на отдельной ВМ, хранилище на PostgreSQL, ежедневный бэкап
- Пилот на 3-5 фоновых потоках параллельно со старой интеграцией

НОРМАЛЬНО

- Мониторинг очередей и каналов в Zabbix: опрос 60 с, алерты в Telegram
- Канонический формат сообщений, трансформации — только на шине
- TLS на внешних HTTP-каналах, отдельная учётка на каждую систему
- Стенд для обновлений шины с копией интеграционного проекта

ХОРОШО

- Поэтапная миграция по доменам со сверкой сообщений и планом отката
- RPO/RTO на каждый критичный поток, кварталный тест восстановления
- Kafka-каналы под потоковые данные, шина — под маршрутизацию и маппинг



Чек-лист самопроверки

☐ Есть ли актуальный реестр интеграций: кто с кем обменивается, в каком формате и с каким объёмом?

☐ Остались ли обмены на неподдерживаемой западной шине или самописных скриптах без владельца?

☐ Выбрана ли платформа из реестра российского ПО и посчитаны ли лицензии по числу пользователей?

☐ Хранилище шины переведено с файловой СУБД на PostgreSQL и попадает ли в ежедневный бэкап?

☐ Есть ли алерт на рост очереди недоставленных сообщений и на свободное место тома шины?

☐ Трансформации данных живут на шине, а не в коде каждой из подключённых систем?

☐ Закрыты ли внешние HTTP-каналы TLS и разведены ли учётки подключённых систем?

☐ Есть ли стенд, где обновление шины проверяется до продакшена?

☐ Задokumentирован ли план отката для каждого мигрированного критичного потока?

Если хотя бы на два вопроса ответ «нет» или «не знаю» — тема требует внимания.



Как поможет ITFresh

ITFresh — ИТ-аутсорсинг для юридических лиц до 50 рабочих мест в Москве и области. 15+ лет практики, собственная инфраструктура в дата-центре МТС (8 серверов Dell Xeon Platinum).

- Аудит интеграционного ландшафта: реестр потоков, зависимости, смета миграции с западной ESB
- Внедрение 1C:Шины или Datareon Platform под ключ: сервер, каналы, трансформации, перенос потоков
- Мониторинг шины в Zabbix 7.0 с алертами в Telegram и регламентом реакции
- Сопровождение: обновления через стенд, бэкапы хранилища, ввод новых обменов за 1–2 дня

15+

лет в ИТ-поддержке

50

рабочих мест — наш профиль

МТС

дата-центр, Москва



КОНТАКТЫ

Обсудить вашу задачу

Сайт **itfresh.ru**

Телефон **+7 903 729-62-41**

Telegram **@ITfresh_Boss**

Бесплатно посмотрим вашу инфраструктуру по этому чек-листу и скажем, где тонко — без обязательств.



itfresh.ru



Техническая база

- 01** 1С:Шина. Руководство разработчика, издание 4: сервер, каналы, PostgreSQL (its.1c.ru — 4.1)
- 02** 1С:Шина — описание продукта, протоколы и лицензии (v8.1c.ru — 2026)
- 03** Datareon Platform. Документация администратора: ESB (docs-platform.datareon.ru — 3.0.53)
- 04** Apache Kafka Documentation (KRaft) (kafka.apache.org — 4.0)
- 05** Zabbix Manual: HTTP agent, triggers (zabbix.com — 7.0)
- 06** Наш шаблон реестра интеграционных потоков (itfresh.ru — 2026)

