

# Ред Платформа - системный каталог и подсистема обновления



Программная платформа  
для быстрой разработки  
информационных систем  
различных классов

Разработка  
кроссплатформенных  
приложений на языке Java

Быстрая (low-code) разработка  
информационных систем  
с применением встроенного  
конфигуратора

Объектно-реляционный подход  
к управлению данными на основе  
СУБД Ред База Данных

Конфигуратор  
на основе модели  
деловых процессов

Интегрированная  
криптография и методы  
защиты информации

Включена в Единый реестр  
российских программ  
для ЭВМ и БД

# Технологии и парадигмы



# Функциональная структура РЕД Платформы



Подсистема ORM



Интерфейсы пользователя



Подсистема документо-оборота



Подсистема обновлений



Подсистема безопасности и ограничения доступа



Подсистема SOAP-интеграции



Планировщик задач



Подсистема репликации



Механизмы кластеризации



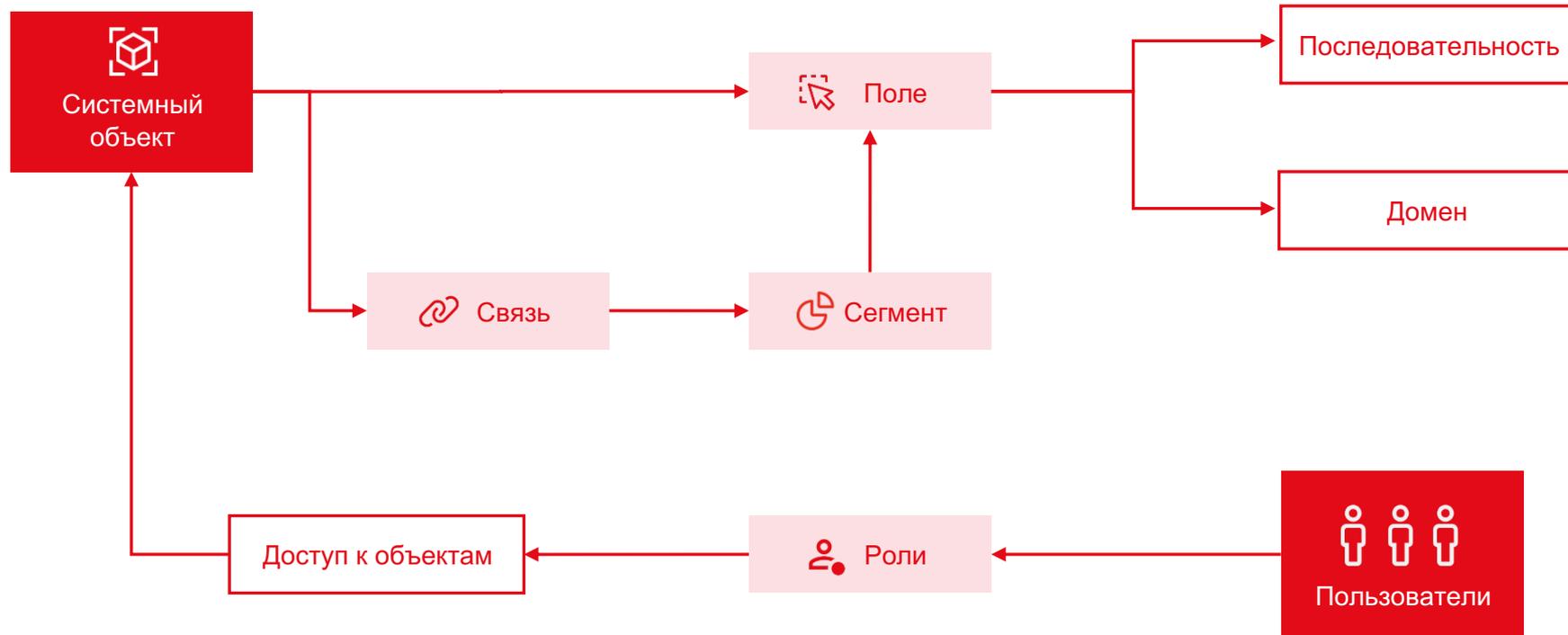
Подсистема отчетности

Дополнительные модули

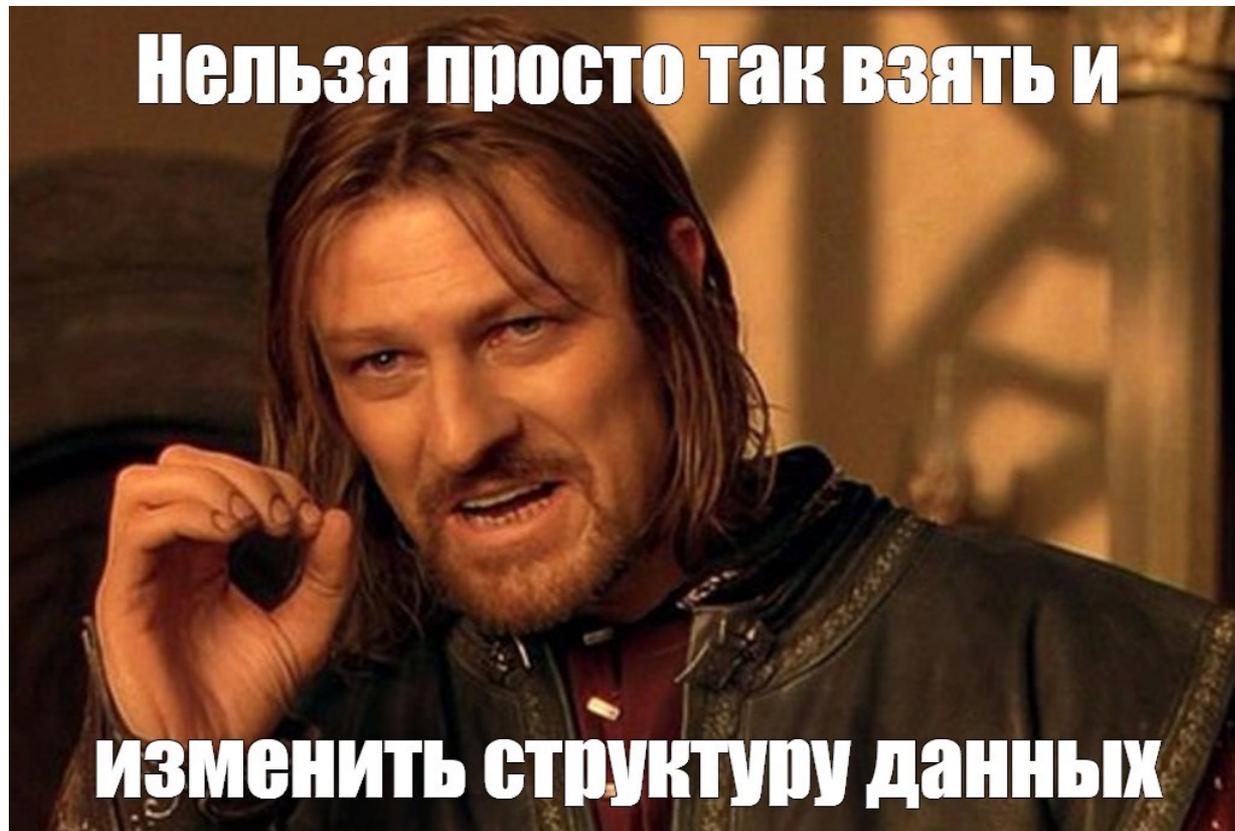
Rest API

Поддержка ФИАС

# Структура системного каталога



## Но Ред База контролирует DDL?



# Синхронизация системного каталога



**Новые домены, последовательности, таблицы**

Кроме вычисляемых полей. Не имеют зависимостей, значит безопасны



**Удалить констрейнты, вычисляемые поля**

Максимум зависимостей, легко восстановить



**Превратить процедуры, функции, представления в пустышки**

Убрать все зависимости, оставить только заголовки



**Изменить таблицы, последовательности**

На этом этапе уже безопасно



**Удалить всё лишнее**

Теперь удаление не вызовет ошибок



**Восстановить процедуры и функции**

Все метаданные для них доступны



**Восстановить представления**

Безопасно пользуемся таблицами, процедурами, функциями



**Восстановить оставшееся, обновить журнальные и репликационные объекты**

Завершающий этап



## Обычная таблица:

- Основной тип системных объектов
- Хранение данных
- Практически неограниченное количество реквизитов



## Временная таблица:

- Данные хранятся, пока цело подключение
- Использование в сложной бизнес-логике
- Удобно для кэширования сложных наборов данных



## Транзакционная временная таблица:

- Хранение данных, пока действует транзакция
- Кратковременное хранение данных



## Связи между объектами:

- Ссылки на справочники
- Вложенные объекты
- Подчиненные списки
- Списки связанных сущностей
- Связи многие ко многим через развязочные таблицы
- Возможность построения иерархических структур



## Индексы:

- По одному или нескольким столбцам
- Первичные ключи, уникальные и неуникальные
- Индексы по вычисляемым выражениям



## Вычисляемые поля:

- Вычисляемые поля по связям
- Вычисляемые поля по выражениям
- Подключение последовательностей или генерация GUID



## Представления:

- Вводим SQL руками
- Именованный запрос
- Настройка полей



## Хранимые процедуры и функции:

- Входные и выходные параметры
- Возможность использования в представлениях и вычисляемых полях
- Сложная бизнес-логика
- Безопасная работа с данными



## Триггеры:

- Обеспечение целостности данных
- Сложная каскадная бизнес-логика
- Аудит
- Безопасность



## Панели фильтрации:

- Особые параметры фильтрации
- Поля с выбором из справочника
- Разметка



## Формы ввода параметров:

- Поля с выбором из справочника
- Разметка



## Интерфейсы:

- Общие наборы полей
- Полиморфизм



## Списки и карточки:

- Настройка видимости полей или столбцов
- Разные представления

# Ред Платформа как средство low-code разработки



Широкое применение  
конфигурирования  
в разработке



Привлечение бизнес-  
аналитиков к работе над  
системой, снижение нагрузки  
на разработчиков



Создание высокофунк-  
циональных прототипов  
систем без написания кода



Возможность развития  
прототипа до уровня готовой  
системы



Снижение необходимости  
в тестировании



Укорачивание  
производственного цикла  
внедрения изменений

# Подсистема ORM



Полноценная  
поддержка  
транзакций



Настройка структуры  
базы данных на основе  
метаданных



Поддержка структур  
данных любой  
сложности



Генерация  
кода на основе  
метаданных



Бизнес-классы на основе  
сгенерированных  
data-классов



Удобные инструменты  
доступа к данным  
с возможностью кастомизации

# Интерфейсы пользователя



Три варианта интерфейса с единой конфигурацией



Генерация экранных форм на основе метаданных, включая вкладки, подчиненные списки, группы полей, вложенные объекты



Генерация списковых форм на основе метаданных, с широкой настройкой фильтрации



Возможность настройки индивидуального представления списков данных для каждого конечного пользователя системы, включая видимость и порядок столбцов, параметры фильтрации



Поддержка сохранения индивидуальных конфигураций через профили



Конфигурирование системного меню через метаданные

# Подсистема документооборота



Конфигуратор статусной модели документооборота



Контроль сроков, ЭЦП, вложения, штрихкодирование, сканирование



События, действия, сообщения



Деревья обработки событий



Методы дерева обработки



Выгрузка в BPMN-представление



## Подсистема SOAP-интеграции

- Простое создание адаптеров
- Пакетный и подокументный обмен
- Протокол гарантированной доставки
- Режимы «точка-точка» или с использованием транзитных узлов
- Поддержка СМЭВ



## Модуль API:

- REST API
- GraphQL API
- Авторизация и аутентификация
- Пользовательские ресурсы
- Системные ресурсы
- Конфигуратор на основе метаданных



## Интеграция с брокерами сообщений

- Apache Kafka
- Apache ActiveMQ Artemis

# Подсистема безопасности и ограничения доступа



Использование сертифицированной СУБД в качестве средства защиты информации



Множественные факторы аутентификации, мульти-факторная аутентификация



Ролевая модель доступа к функциям и данным



Меточный доступ к документам



Разграничение доступа к конфиденциальной информации



Ведение журналов безопасности и аудита

# Подсистема конфигурирования



Структуры данных любой сложности с возможностью синхронизации в режиме реального времени



Бизнес-процессы, включая междокументное взаимодействие и интеграцию



Формирование пакетов обновления



Параметры безопасности и аудита



Разграничение доступа к данным и операциям



Пользовательский интерфейс, включая экранные формы, списки, меню и пр.



SOAP-интеграция, включая СМЭВ и протокол гарантированной доставки



Правила преобразования данных в формат xml с генерацией схем данных



Настройки точек входа REST API с генерацией спецификаций OpenAPI

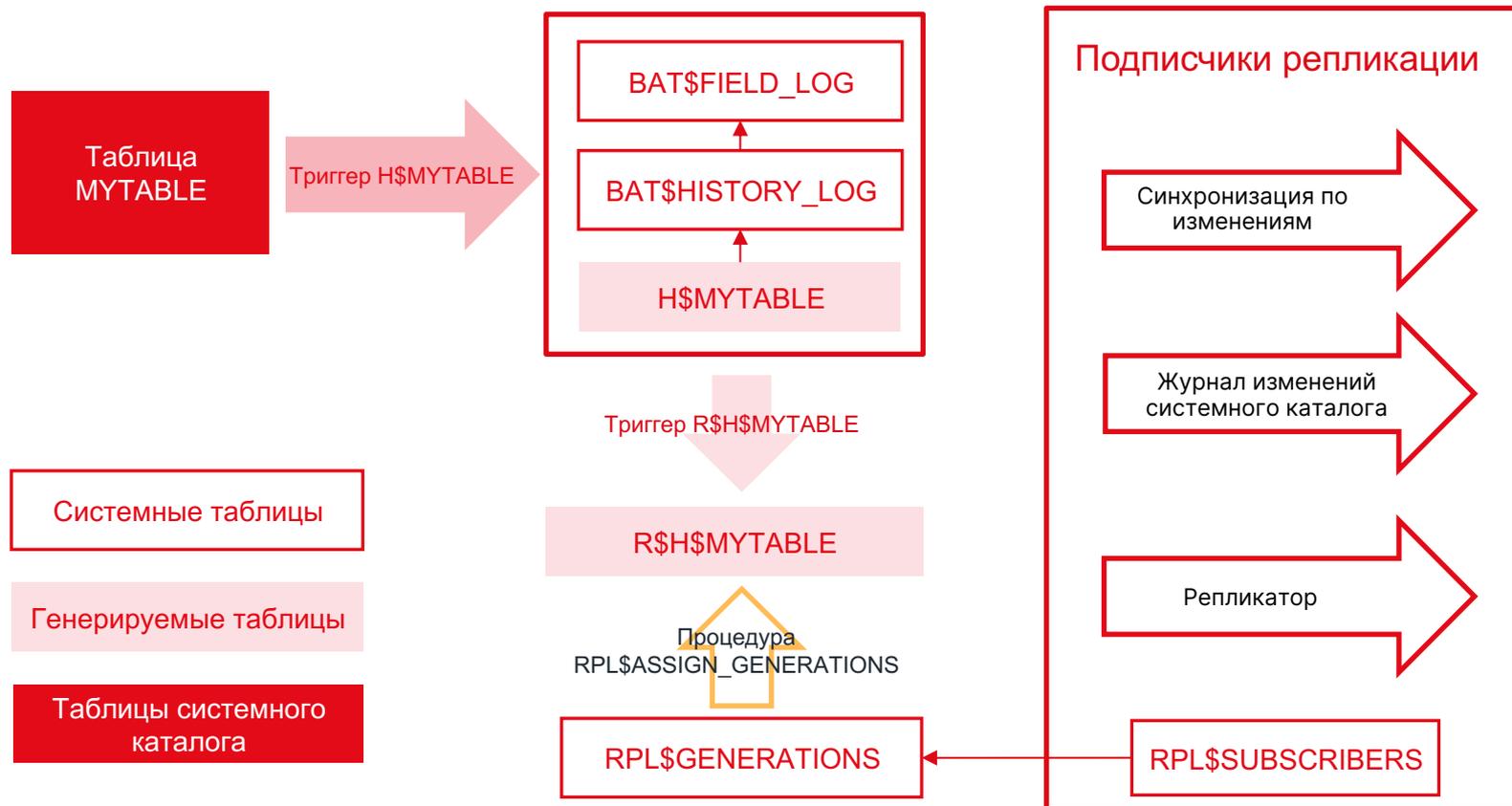


Ресурсы для базового доступа к данным с иерархической вложенностью

# Наконфигурировали, что дальше?



# Немного о фиксации изменений





## H\$-таблицы:

- Служит также и для отслеживания изменений и аудита
- Отдельная таблица для каждой отслеживаемой таблицы
- Хранит ключ записи и ссылку на запись в журнале аудита `BAT$HISTORY_LOG`



## R\$H\$-таблицы:

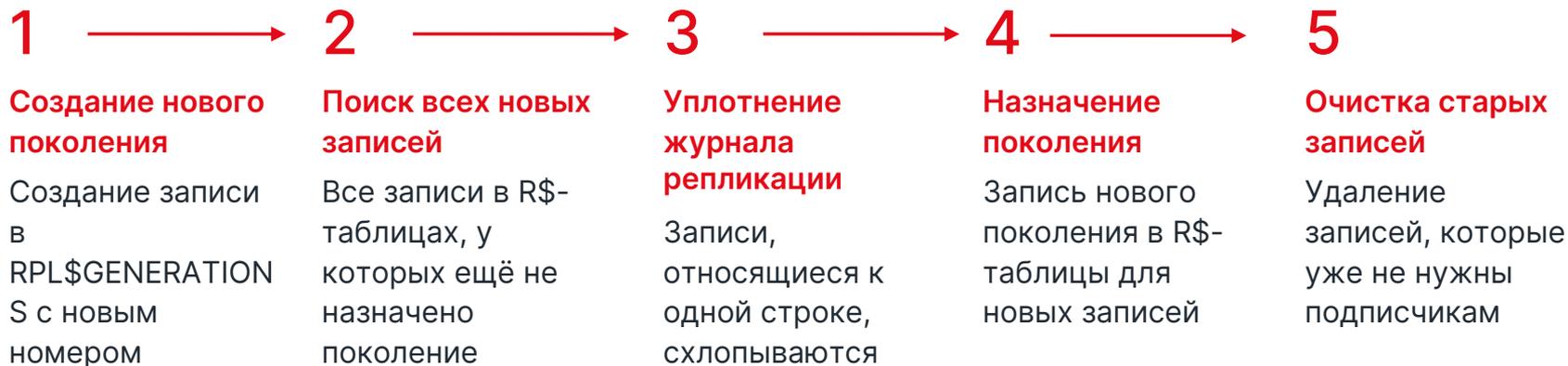
- Репликационный журнал для таблиц аудита
- Отдельная таблица для каждой H\$-таблицы
- Хранит ключ записи и номер поколения



## Триггеры изменений

- H\$-триггеры
- R\$-триггеры
- Наполняют соответствующие таблицы

# Процедура RPL\$ASSIGN\_GENERATION



# Наполнение журнала изменений

1 →

## Получение дельты между поколениями

Получаем перечень всех записей в R\$H\$-таблицах для помеченных объектов, которые новее заданного поколения

2 →

## Подтягиваем данные из H\$- и VAT\$- таблиц

Строим структуру данных изменений

3 →

## Преобразование данных с одновременным уплотнением

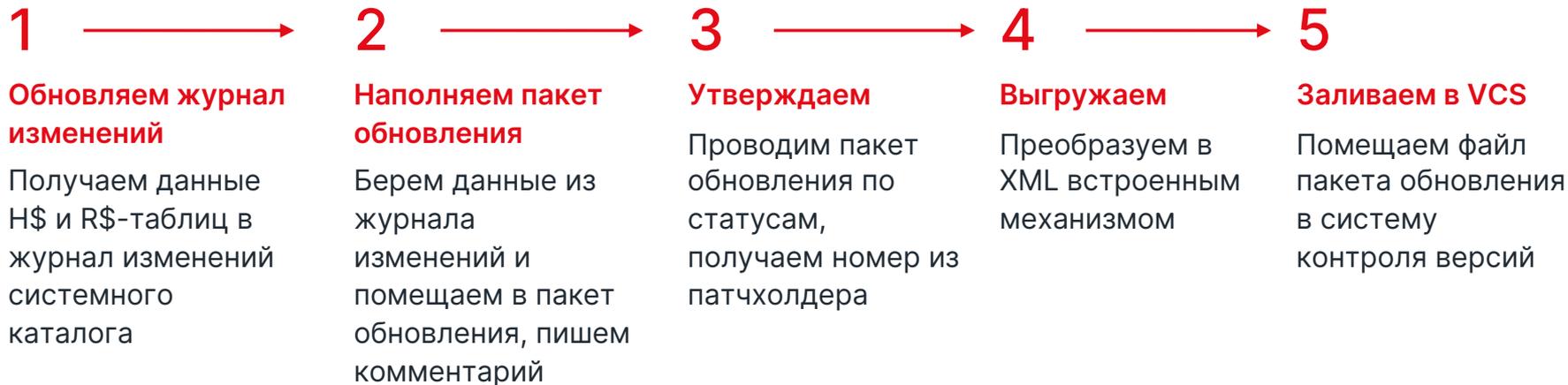
Все изменения в рамках одной сущности помещаются в одну строку журнала

4

## Добавляем записи в журнал изменений

Запись нового поколения в R\$-таблицы для новых записей

# Делаем пакет обновления





## Когда нужны скрипты:

- Сложное изменение типа поля
- Переименование столбцов таблиц
- Генерация данных
- Изменение данных
- Перемещение данных



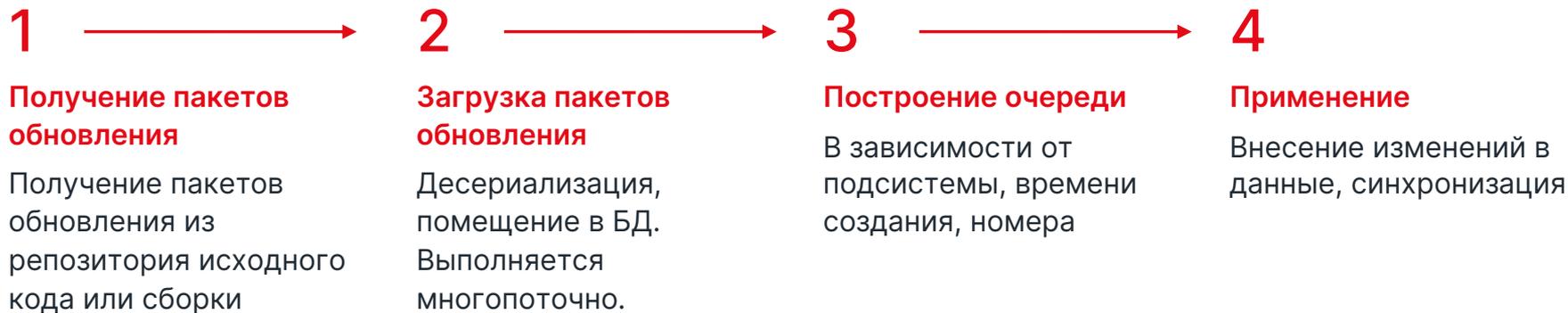
## До применения журнала

При необходимости выполняется промежуточный коммит

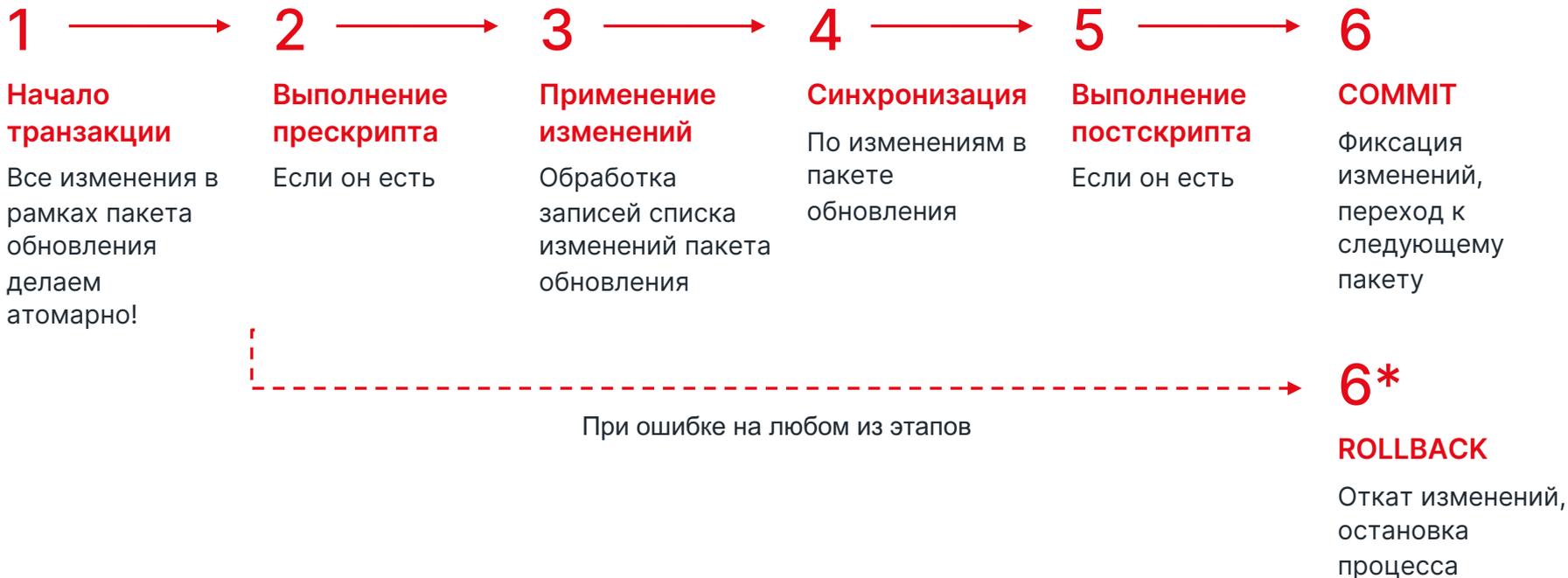
## После применения журнала

Изменение данных в новых объектах не поддерживается!

# Загрузка пакетов обновления в новую базу



# Принципы применения пакетов обновления





## Причины ошибок:

- Отличие контекста
- Неполный набор изменений
- Некорректный набор изменений
- Ошибки скриптов



## Последствия ошибок:

- **Данные целы**
- Придётся переделать пакет обновления
- Или устранить отличие контекста



## Устранение ошибок:

- Перезаполнить журнал изменений пакета обновления
- Исправить скрипты
- Устранить проблемы в целевой базе
- Выгрузить заново, перезалить в репозиторий
- Заново применить

# Можно ли в прикладных системах?

 Да, можно!



**Где следует использовать подсистему обновления:**

- Метаданные системных модулей
- Прикладные метаданные
- Статические справочники



**Где не следует использовать подсистему обновления :**

- Документы
- Реестровые данные
- Динамические справочники

# Модульность



Системные модули – REST, GraphQL, ФИАС и пр.



Прикладные модули как основа для АИС любой сложности



Раздельные наборы пакетов обновления для модулей



Иерархия модулей: структура метаданных в одном модуле, наполнение в другом



Легкое добавление модулей в проект через конфигурацию

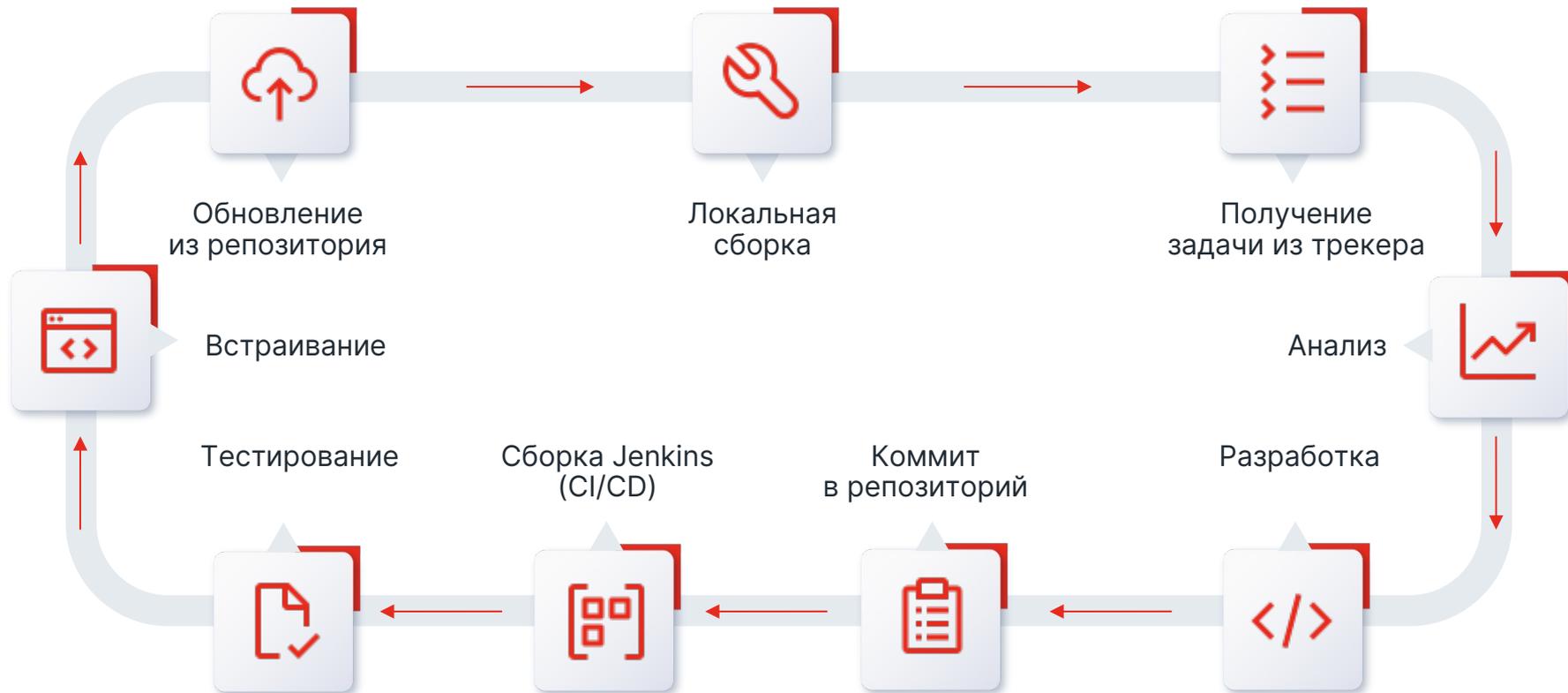


Встраивание прикладного кода

# Статистика подсистемы обновления

Подсистема	Пакетов обновления	Системных объектов	Таблиц
Ред Платформа	2935	745	367
АИС ФССП	31548	5491	3285
МВВ ФССП	7405	1800	1195
РЕД ШЛЮЗ	14435	8252	7810
РЕД ШЛЮЗ РОИВ	1718	469	461
СУ БВС	1332	472	430
Лицензирование 93-ФЗ	2108	655	579

# Производственный цикл разработки



# Альтернативы в плане миграций



Flyway



Liquibase



## Основные преимущества:

- Автоматическая фиксация изменений
- Создание миграций через пользовательский интерфейс
- SQL (почти) не нужен!
- Работа с любыми данными
- Не требуется внешних инструментов

# Что есть в Ред Базе Данных, и чего нет в PG

Функция	Назначение	В PostgreSQL
Доверенная аутентификация	Используется для SSO	Нет
Аутентификация по сертификату	Аутентификация в клиенте, SOAP	Только SSL/TLS
Параллельные транзакции в одном соединении	Используется в толстом и тонком клиенте, базовая функция платформы	Нет
Вычисляемые колонки	Активно используется в Платформе и прикладных системах	Появились только в версии 12 в
Временные таблицы	Используются в платформе и прикладных системах	С ограничениями

# Что есть в Ред Базе Данных, и чего нет в PG

Функция	Назначение	В PostgreSQL
Аудит производительности и событий в реальном времени	Подсистема безопасности, диагностика	Нет
64-разрядный счетчик транзакций	Важно для высоконагруженных систем	Enterprise
Контроль зависимостей объектов в операциях DDL	Подсистема обновления	Нет
Сжатие данных таблиц	Снижаются требования к хранилищу (SSD) и число операций ввода/вывода	Enterprise
Сжатие данных индексов	Снижаются требования к хранилищу (SSD) и число операций ввода/вывода	Нет

# Наши партнеры

Компания Id Systems



Компания «Эрлан»



Лицензирование



Обучение

Компания «Метран»



Компания «Вебби»



Поддержка  
и сопровождение



Совместные  
проекты

# Используйте потенциал продуктов РЕД СОФТ

РЕД СОФТ  
[red-soft.ru](http://red-soft.ru)  
[info@red-soft.ru](mailto:info@red-soft.ru)

РЕД ОС  
[redos.red-soft.ru](http://redos.red-soft.ru)  
[redos@red-soft.ru](mailto:redos@red-soft.ru)



Российское  
программное  
обеспечение

