

Логинов Алексей Николаевич  
Кот Павел Николаевич

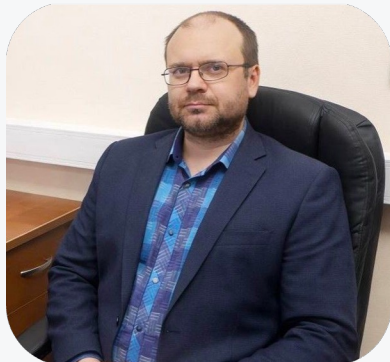
Мигрировать работающую  
информационную систему  
за один день. Почему так долго?

# О докладчиках



## **Алексей Логинов**

Лид проекта / Архитектор миграции  
Руководство проектом, архитектура решения, стратегия миграции и выбор СУБД  
*Начальник сектора разработки прикладного ПО*



## **Павел Кот**

Технический лидер / Ведущий разработчик  
Техническая реализация, перенос и адаптация кода, тестирование и запуск  
*Ведущий инженер-программист*

# Почему возникла задача миграции

## Импортозамещение

**Указ Президента РФ  
№166 от 30.03.2022**

Обеспечение  
технологической  
независимости КИИ

**Постановление  
Правительства №1912**

горизонт завершения –  
до 2030 года

## Технологические риски

**Зависимость  
от иностранной СУБД**

Риски санкционных  
ограничений и недоступности  
сопровождения

**Ограничения  
масштабируемости**

Риски снижения  
производительности

## Требования к развитию ИС

**Модернизация  
технологического стека**

Переход  
на платформонезависимые  
решения

**Унификация  
архитектуры**

Единый подход к разработке  
и сопровождению ИС

# Ландшафт информационных систем

Внешние ИС по взаимодействию с заводами-строителями (IBX)  
Answer

Сервисные модули (IBX)  
Поиск • Проверка КВЗ • Структуры КД

Инженерные ИС (BDE)  
- Электрическая часть  
- Спецпокрытия  
- Трубопроводы

Инженерные ИС (IBX)  
- Корпус  
- Установка  
- Мебель  
- МСЧ  
- Заказные ведомости

Инженерные ИС (FireDAC)  
- Системы  
- Ведомости ТД

PLM-система  
СУБД ORACLE

MDM-система (НСИ)  
СУБД INTERBASE

Сервер приложений 1  
СУБД INTERBASE

Сервер приложений 2  
СУБД INTERBASE

Журналирование  
MONGODB

Проектная база данных (КБД)  
СУБД INTERBASE

Стек: Delphi • Python • C++ • SQL/PSQL • XML • TensorFlow • Jupyter • BDE • IBX • FireDAC • FastReport • InterBase • Oracle • MongoDB • Windchill • Windows • Astra Linux • Git • Scrum • YouGile • MS Project

# Постановка задачи перед руководством

## Инициация НИР «Кроссплатформа 1.0» (2023)

Исследование путей модернизации ИС  
в условиях санкционных ограничений

## Цель НИР

Разработка прототипа  
клиент-серверного приложения  
на платформонезависимом стеке

## Ключевые задачи НИР

- Выбор технологического стека разработки
- Определение СУБД для миграции
- Формирование дорожной карты перехода
- Оценка применимости для модернизации ИС



# Выбор решения в контексте СУБД

## Рассмотренные варианты

- PostgreSQL-based решения (Postgres Pro, Jatoba, Arenadata)
- Firebird
- Ред База Данных

## Критерии выбора

- Совместимость с существующими системами
- Трудоемкость миграции
- Производительность в многопользовательской среде
- Доступность и независимость от санкций

## Результат анализа

Ред База Данных выбрана как оптимальное решение для миграции с InterBase с минимальными изменениями в архитектуре



# Сравнительный анализ СУБД

| Таблица                                   | PostgreSQL | Firebird  | Ред База Данных |
|---|------------|-----------|-----------------|
| Трудоемкость миграции                     | 2          | 4         | 4               |
| Совместимость с текущими ИС               | 2          | 4         | 4               |
| Сложность сопровождения                   | 3          | 5         | 5               |
| Управляемость лицензирования              | 3          | 4         | 5               |
| Соответствие требованиям импортозамещения | 4          | 2         | 5               |
| <b>ИТОГО</b>                              | <b>14</b>  | <b>19</b> | <b>23</b>       |

# Почему выбрали Ред База данных

## Совместимость с существующей архитектурой

Высокая совместимость с InterBase/Firebird

Минимальные изменения прикладных систем

## Соответствие требованиям импортозамещения

Российская СУБД, сертификация ФСТЭК  
Отсутствие санкционных рисков

## Минимизация трудоемкости миграции

Сохранение структуры БД, процедур и логики

Снижение сроков и рисков перехода

## Организационная готовность

На момент старта миграции: 3  
сертифицированных администратора СУБД

# Дорожная карта замещения СУБД



## Модернизация пользовательского интерфейса (не входит напрямую в контур СУБД)

- Обновление GUI
- Внедрение кросс-платформенных решений

# Описание пилотной системы

## ИС ПО ФОРМИРОВАНИЮ И ВЫПУСКУ КД



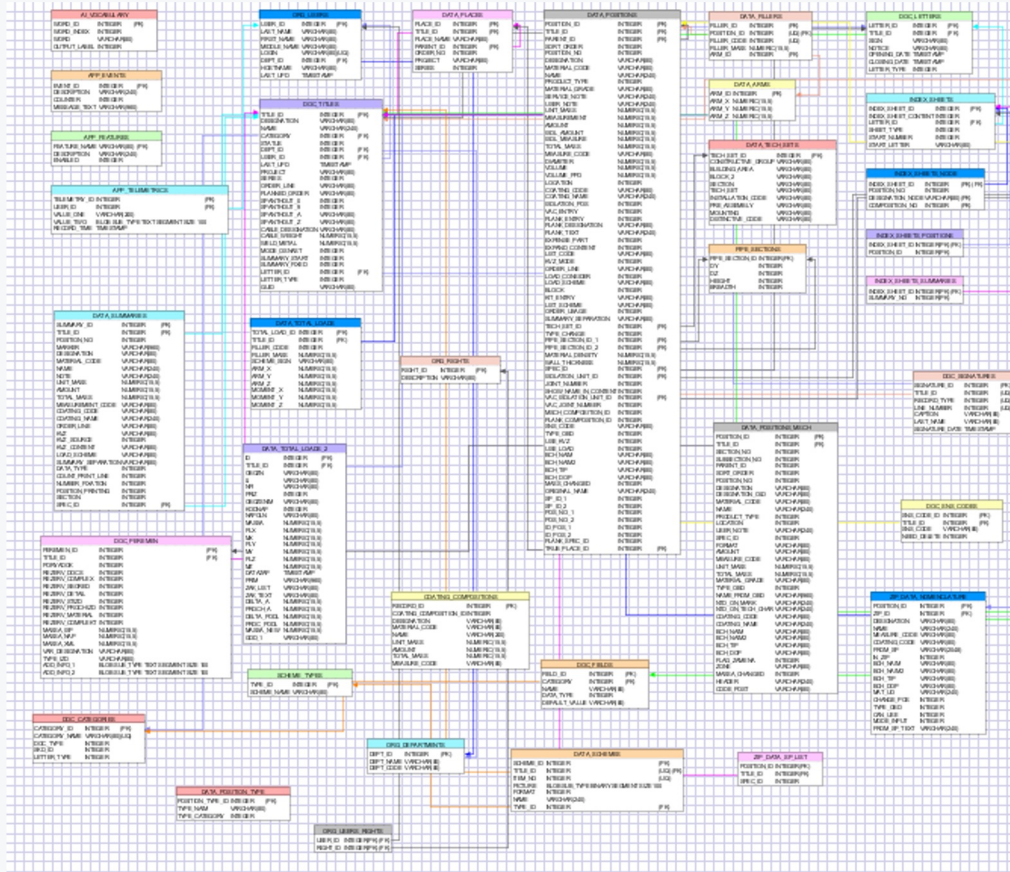
14 модулей

50-100 ПОЛЬЗОВАТЕЛЕЙ ПОСТОЯННО, НА ПИКЕ 500-600



# Архитектура БД пилотной системы

47 таблиц  
4 представления  
61 процедура  
56 триггеров  
146 индексов



# Подход к миграции

1. Поэтапный переход  
(на каждом этапе решается конкретная задача, в рамках перехода)
2. Разворачивание тестового стенда для тестирования успешности перехода на каждом этапе
3. Создание реплики БД (на Ред БД) для анализа и тестирования функционала ИС
4. Конечный переход без остановки функционирования (кроме периода ~1 час на репликацию данных)



# Основные этапы миграции

1. Анализ и корректировка структуры
2. Модернизация доступа к БД из ПО – переход на FireDAC
3. Создание реплики БД с последующим периодическим обновлением из рабочей БД (с изменением кодировки win1251 ->utf8)
4. Поэтапное тестирование всего функционала на работоспособность (боевая версия ПО, позволяющая переключаться между IB и Ред БД на уровне настроек)
5. Собственно миграция

# Анализ и корректировка структуры

1. Исключение использование зарезервированных слов

- а) создание дублирующего поля;
- б) настройка триггеров по поддержке синхронности данных;
- в) доработка комплекса ПО на переход использования нового поля, с контролем перехода пользователей на новые версии;
- г) исключение «старого» поля.

2. Переименование переменных, совпадающих с ключевыми словами в триггерах и хранимых процедурах.

|  |   |  |
|--|---|--|
| <pre>ZONE          SHORTSTRING COLLATE PXW_CYRL /* SH "POSITION"    "INT" /* "INT" = INTEGER */, OBOZ          STRING COLLATE PXW_CYRL /* STRING</pre> | → | <pre>ZONE          SHORTSTRING COLLATE PXW_CYRL /* SH NUM_POSITION  DOMAIN INTEGER /* DOMAIN_INTEGER = OBOZ          STRING COLLATE PXW_CYRL /* STRING</pre> |
|--|---|--|

```
AS
declare variable LOC_INT integer; /* ID подчинённой записи */
declare variable ROW_COUNT integer; /* счётчик обработанных записей */
BEGIN
```

# Анализ и корректировка структуры

3. Проверка не селективных процедур на отсутствие suspend.

4. Анализ и тестирование запросов с GROUP BY.

```
SELECT CASE WHEN (a.section IS NULL) THEN 1 ELSE a.section END
FROM data_positions a
WHERE (a.title_id = :LOC_TITLE_LINK_ID) AND (a.isolation_pos = 1)
GROUP BY a.section
ORDER BY 1
```

```
SELECT DISTINCT CASE WHEN (a.section IS NULL) THEN 1 ELSE a.section END
FROM data_positions a
WHERE (a.title_id = :LOC_TITLE_LINK_ID) AND (a.isolation_pos = 1)
ORDER BY 1
```

```
SELECT CASE WHEN a.isolation_unit_id IS NULL THEN 0 ELSE a.isolation_unit_id END
FROM data_positions a
WHERE (a.title_id = :title_link_id) AND (a.joint_number > 0)
GROUP BY a.isolation_unit_id
ORDER BY a.short_name
```

```
SELECT CASE WHEN a.isolation_unit_id IS NULL THEN 0 ELSE a.isolation_unit_id END
FROM data_positions a
WHERE (a.title_id = :title_link_id) AND (a.joint_number > 0)
ORDER BY a.short_name
```

# Технические проблемы миграции

1. Использование стандартных UDF функций IB → корректировка триггеров, хранимых процедур, представлений в части замены UDF функций на встроенные в Ред БД.
2. Запросы в ПО, не соответствующие стандартам, реализованным в Ред БД → исправление текстов запросов.

```
V_FIRST_LETTER = SUBSTR(OLD.SNS_CODE, 1, 1);
```

```
V_FIRST_LETTER = SUBSTRING(OLD.SNS_CODE FROM 1 FOR 1);
```

```
NEW.DEPART_CODE = LTRIM(RTRIM(NEW.DEPART_CODE));
```

```
NEW.DEPART_CODE = TRIM(NEW.DEPART_CODE);
```

# Технические проблемы миграции

3. Статичная типизация полей БД, приводящая к ошибкам в связи со сменой кодировки → переход на динамическую типизацию.
4. Функциональное тестирование приложения (в части запросов) в связи с различием результатов на BDE и FireDAC.

# День X: миграция за ~1 час

Предварительно подготовлена «пустая» БД  
(только мета-данные)

- Уведомление пользователей о тех. работах
- Остановка «боевой» БД
- Репликация данных на новую СУБД (~1 час)
- Переключение настроек ПО
- Запуск системы
- Проверка работоспособности

# День 1 после миграции

В рамках действующего функционала, осуществляется импорт данных из других БД под управлением InterBase.

При перегрузке данных (через ПО) возникла ошибка с кодировкой в работе с текстовыми BLOB-полями

Причина проблемы:

архитектурно в процессе перегрузки данных, BLOB поля всех субтипов обрабатывались как бинарные, без учёта различий в кодировках для случаев «BLOB sub\_type 1».

Исправления внесены в ПО, данные повторно импортированы без ошибок без необходимости отката всей системы.

# День 2 после миграции

Ошибка при работе агрегирующей функцией SUM.

Проблема: поля, возвращаемые как SUM от NUMERIC полей в ПО интерпретировались как VARCHAR поля, преобразованные в строку по правилу DecimalSeparator на серверной ОС. Чтение на клиенте данного поля как числа, в случае установки другого разделителя, приводило к исключению.

Решение: явное указание возвращаемого типа в запросе  
замена SUM(AMOUNT) на CAST(SUM(AMOUNT) AS NUMERIC(15,5))

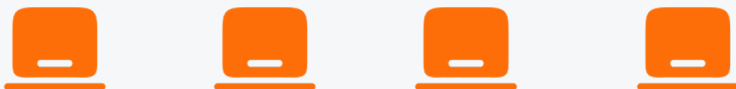
Ошибка к потере данных не приводила, оперативно устранена.  
Анализ причин возникновения внесён в to-do list.

# Что улучшилось в ИС

1. Использование кодировки UTF8 открывает новые функциональные возможности по использованию спец.символов, необходимых при формировании выпускаемой документации.
2. Встроенная система MON\$ таблиц предоставляет значительно больше функциональных возможностей и сведений для анализа работы пользователей с БД
3. Диагностика запросов предоставляет значительно больше возможностей для анализа и поиска «тяжёлых» запросов. После миграции происходит боевое «тестирование» корректности работы всего функционала. Следующий этап – анализ запросов, поступающих от приложения для их оптимизации.

# Изменения в архитектуре БД пилота

БЫЛО



Приложения

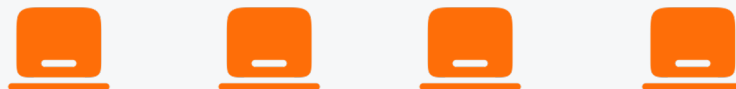


Сервер IB

Сервер IB

Сервер IB

СТАЛО



Приложения

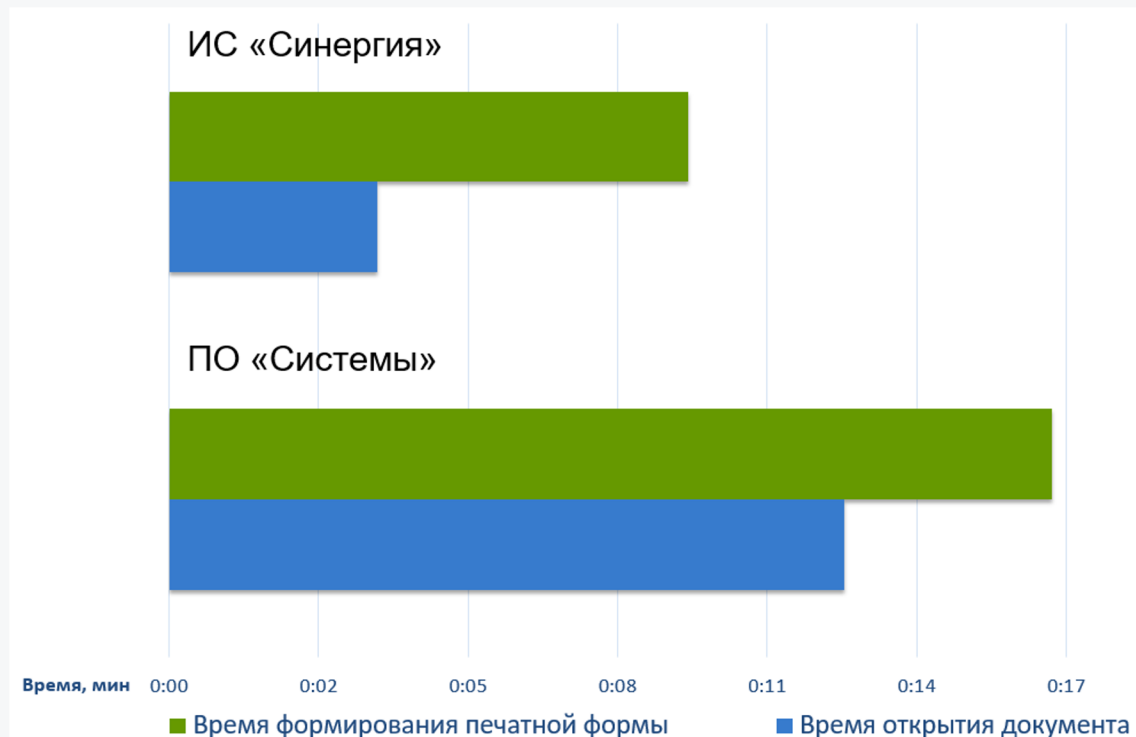


Сервер IB

Сервер IB

Сервер Ред БД

# Метрики увеличения быстродействия



Общее увеличение  
быстродействия

до 40%



# Увеличение надежности после миграции



# Дорожная карта трансформации СУБД

2023

2024

2025

19.01.2026

2027-2030



Сервер IB

Сервер IB

Сервер IB



Сервер Ред БД

Сервер IB

Сервер IB



Кластер Ред БД

Кластер Ред БД

Кластер Ред БД

# Тиражирование подхода

1.

Выработана методология перевода исходного кода на технологию FireDAC, которая предоставляет возможность работы с разными СУБД с возможностью переключения на уровне настроек.

2.

Разработан порядок исправления структуры БД для подготовки к миграции.

3.

Применимо к другим ИС на IB (вторая ИС на этапе 2 миграции на Ред БД).

# Выводы

1. Полный пакет подготовки ИС к миграция возможен без остановки функционирования.
2. В рамках перехода между СУБД не зафиксировано потери функциональности ИС.
3. Зафиксирован рост производительности и надежности.
4. Обнаруженные проблемы относятся к архитектурным особенностям ПО – оперативно идентифицируются и исправляются в штатном режиме поддержки.

# Результаты для бизнеса

1. Снижение  
технологических рисков  
уменьшена зависимость  
от иностранной СУБД



2. Рост производительности  
до 40%



3. Повышение надежности  
отсутствие падений сервера  
с 19.01.2026



4. Масштабируемость решения  
все готово к тиражированию  
на другие ИС



# Что дальше

- Тиражирование на другие ИС
- Завершение миграции ИВ-контуров
- Развитие мониторинга и оптимизации
- Переход к кроссплатформенному клиенту

Спасибо за внимание!