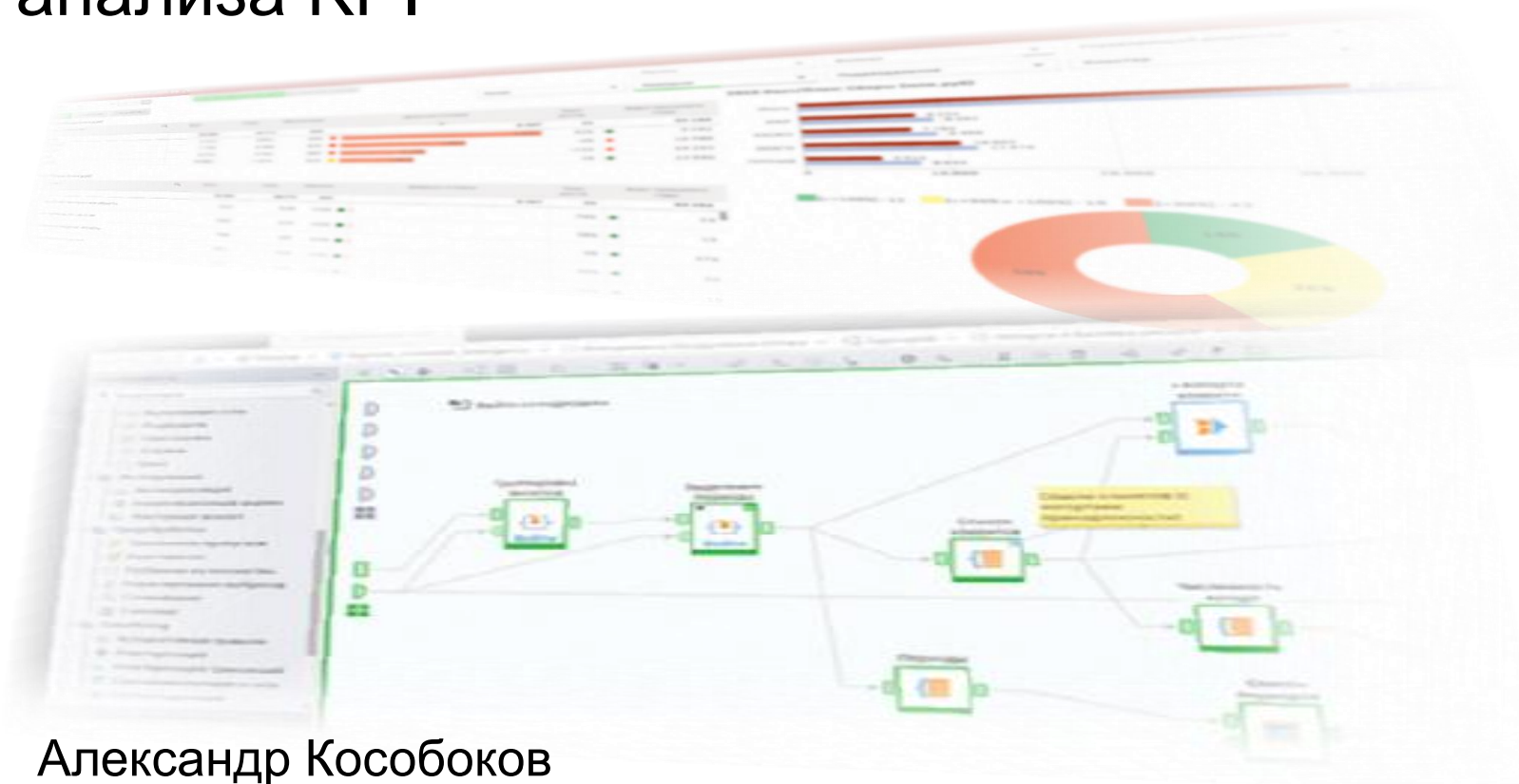
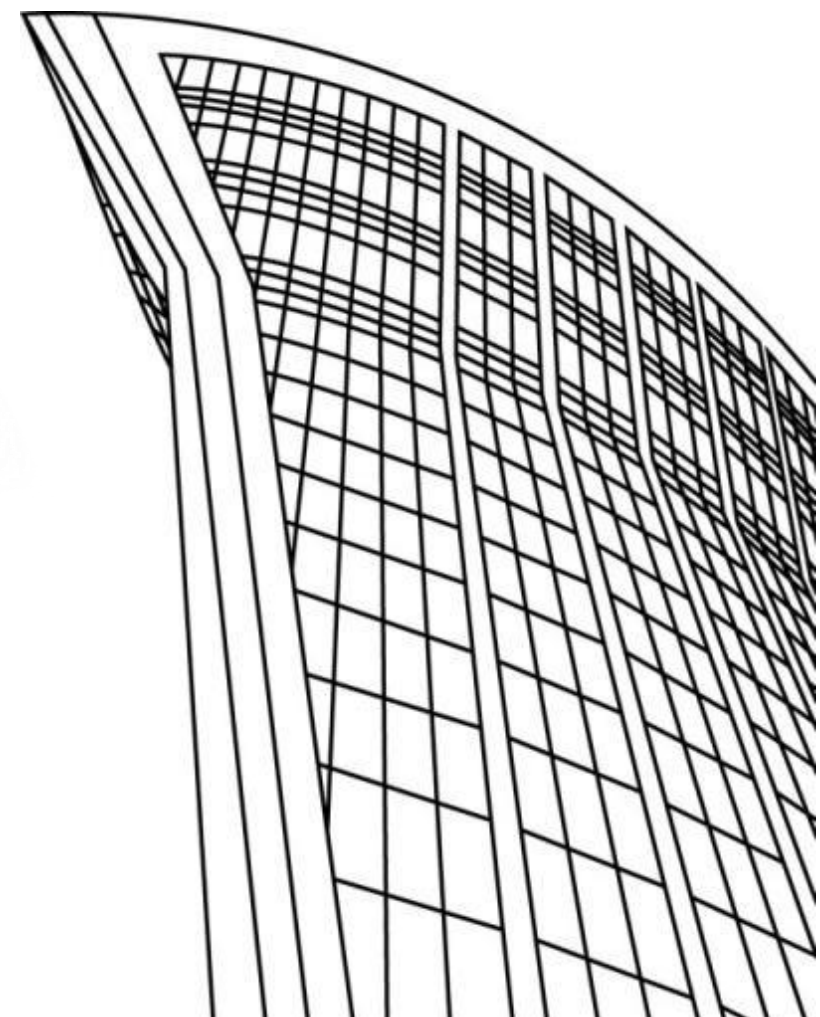


Автоматизация сбора и комплексного анализа КРІ



Александр Кособоков
Архитектор систем бизнес-анализа



Крупнейшая в России страховая компания:

- более **2 000** офисов;
- свыше **17 млн.** клиентов;
- свыше **40 000** агентов;

Бизнес-задачи

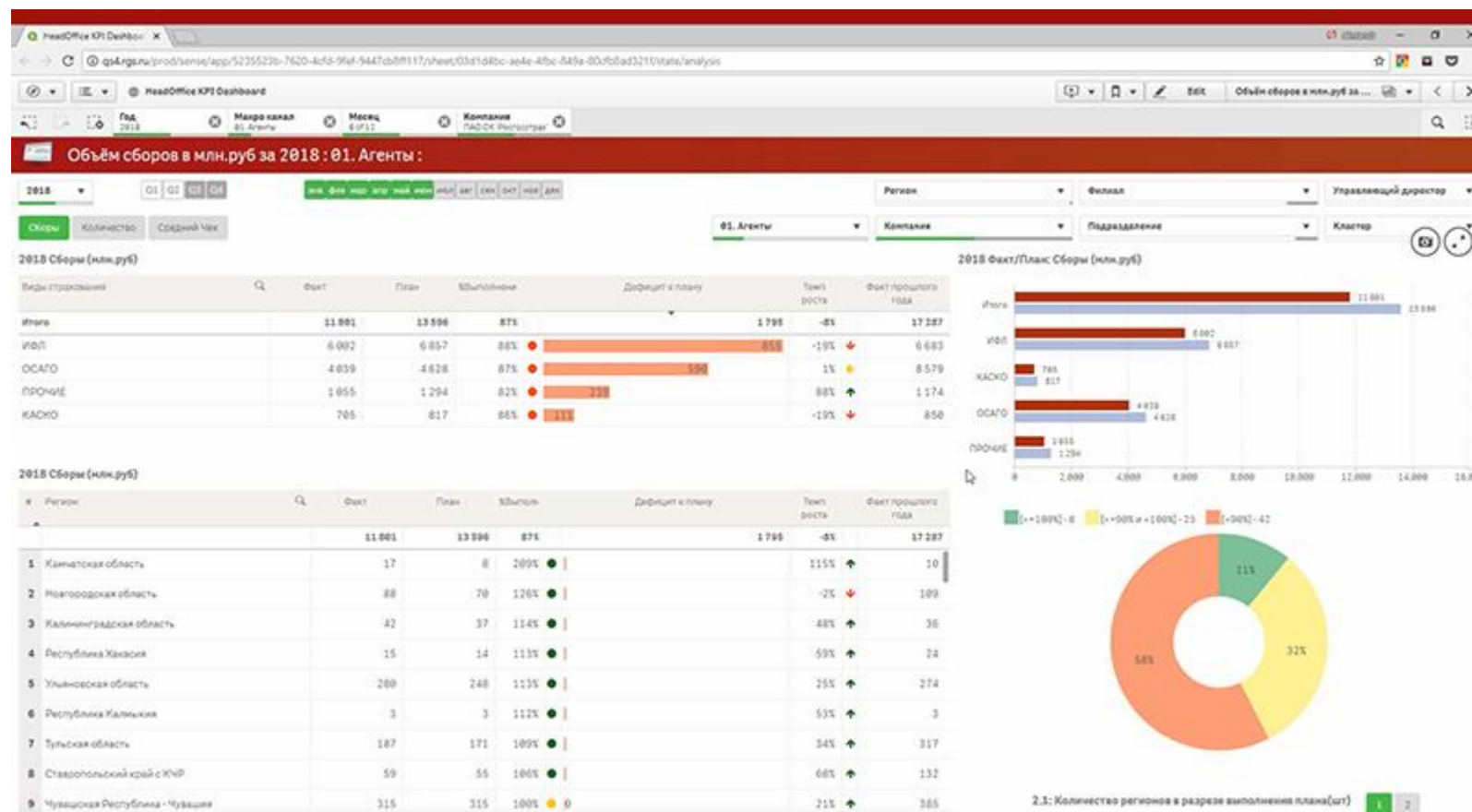
- Отслеживание ключевых показателей (KPI)
- Сокращение времени на поиск информации
- Единый источник истины в компании

Пользователи

- Руководители блоков
- Руководители филиалов
- Руководители отделов
- Менеджеры агентских групп

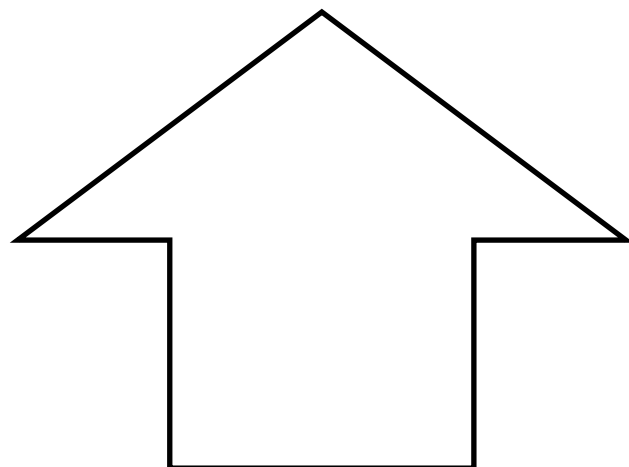
Источники

- Корпоративное хранилище и витрины данных
- Файлы MS Excel, MS Access, .xml, .csv, .txt и многие другие
- Web-источники данных



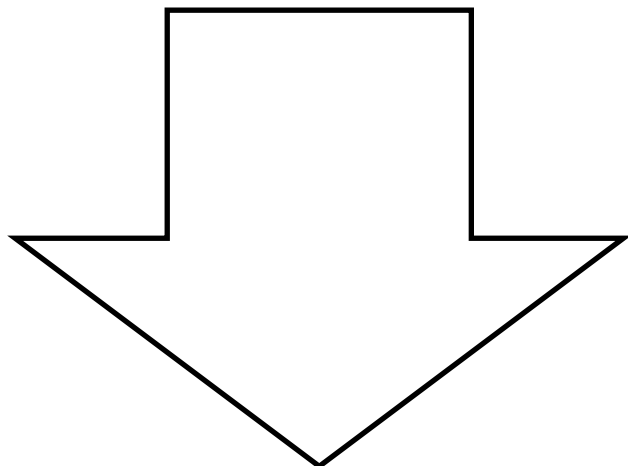
Для визуализации и дистрибуции бизнес-аналитик применяется QlikSense.

- интерактивные сводки в реальном времени;
- мониторинг изменений бизнес-процессов;
- поиск взаимосвязей;
- анализ данных из разных источников.



Плюсы:

- Интерактивные дашборды
- Удобный интерфейс
- Большой выбор способов отображения
- Высокая скорость



Минусы:

- Сложность подготовки данных
- Отсутствие методов продвинутой аналитики



Проблемы:

- Привлечение IT департамента
- Excel hell
- Программирование на Python
- Сложность поддержки скриптов
- Усложнение логики обработки
- Увеличение объемов данных

Пожелания:

- Минимизация участия IT
- Минимизация кодирования
- Интеграция с разнородными источниками
- Гибкость настройки логики
- Высокая скорость обработки

Единый инструмент консолидации,
очистки и обогащения данных:

- Доступность
- Визуальный конструктор
- Низкий порог входа
- Быстрый результат
- Централизация

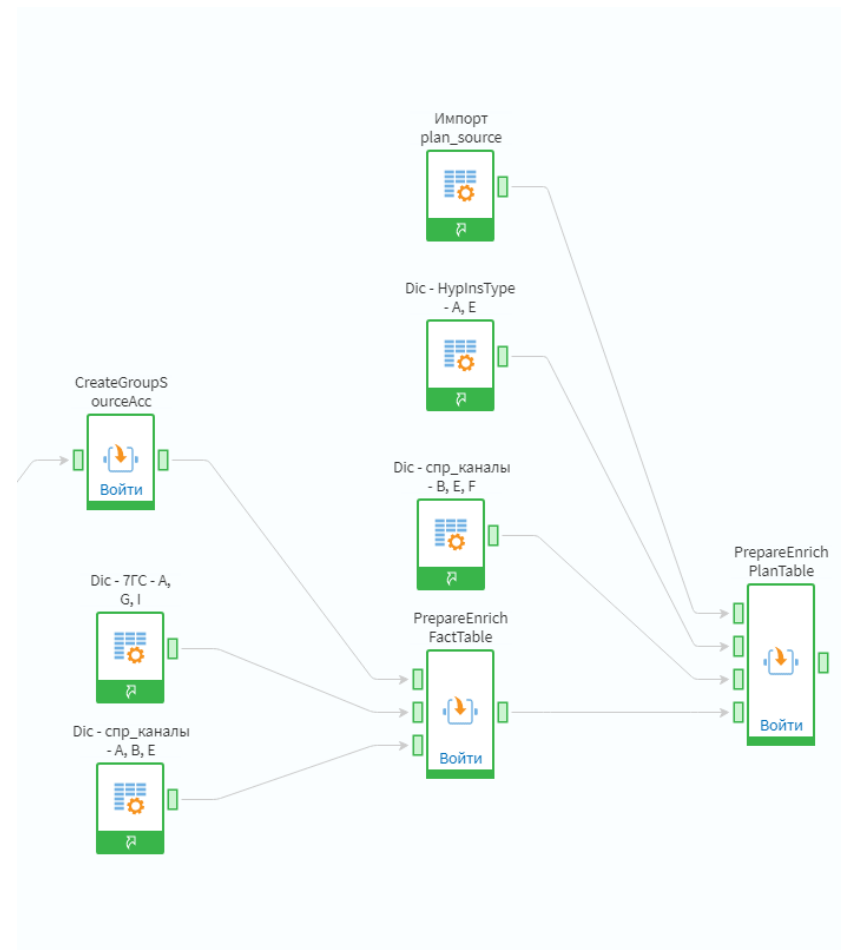


Loginom

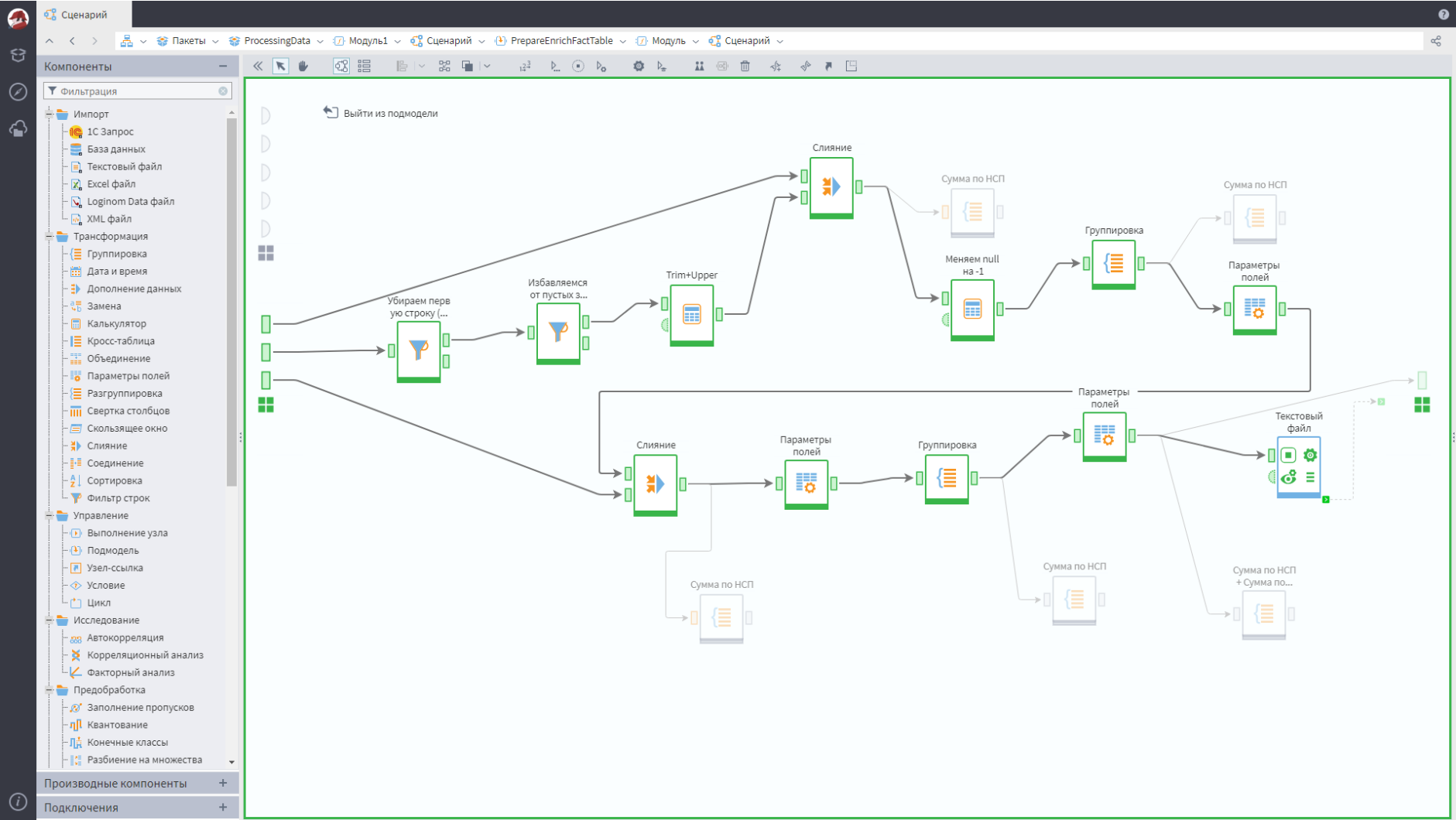
Миграция сценария



```
nsnm(big_dataframe['Договор для формы 162'])
print('#####')
def PrepareEnrichFactTable(fact, dic):
    df_sum = fact
    #Обогащаем таблицу факта - Глобальными видами страхования (ГВС) и Глобальными линиями бизнеса по
    df_7gs = pd.read_excel(dic, '7ГС', usecols = "A,G,I")
    df_7gs.dropna(inplace=True)
    df_7gs['ГВС'] = df_7gs['ГВС'].str.upper().str.strip()
    df_7gs['ГЛБ'] = df_7gs['ГЛБ'].str.upper().str.strip()
    df_7gs['Код 7-ГС'] = df_7gs['Код 7-ГС'].astype(int)
    df_sum = pd.merge(df_sum, df_7gs, left_on=['Код 7-ГС'], right_on=['Код 7-ГС'], how='left')
    print('1.Проверка после объединения с 7-ГС:')
    nsum(df_sum['НСП'])
    df = df_sum.fillna(-1).groupby(columns_groupby2)[measure_to_sum].sum().reset_index()
    print('1.Проверка после группировки 7-ГС:')
    nsum(df['НСП'])
    #####
    #Добавляем микроканал
    dfch = pd.read_excel(dic, 'спр_каналы', usecols = "A,B,E")
    df = pd.merge(df, dfch, left_on=['Признак партнера', 'Код канала продаж'], right_on=['Партнер', 'Кана
    print('2.Проверка после объединения с макроканалами:')
    nsum(df['НСП'])
    df.drop(['Канал продаж 2008', 'Партнер'], axis=1, inplace=True)
    df.rename(columns={'Дата начала отчетного периода': 'Дата', 'Спр 1 Канал 1 Макро': 'Макро канал'})
    df = df.groupby(columns_groupby3)[measure_to_sum].sum().reset_index()
    print('2.Проверка после группировки с макроканалом:')
    nsum(df['НСП'])
    #####
    df.rename(columns={'Договор для формы 162': 'Кол-во Договоров'}, inplace=True)
    print('Контрольная проверка перед сохранением в файл:')
    nsum(df['НСП'])
    nsum(df['Кол-во Договоров'])
    #print('Сохраняем результат в файл:')
    #df.to_csv(source+'fact.txt', index=None, sep='|')
    return df
def PrepareEnrichPlanTable(plan, dic, fact):
    #Загружаем и обогащаем данные из Hupreion - план на 2018
    dfhyp = pd.read_csv(plan, sep='|', nrows = None)
    print('Статистика по файлу с планом:')
    nrow(dfhyp)
    #Важно: колонки постоянно меняются местами
    dfhyp[['c1', 'c2', 'c3', 'c4', 'v1', 'v2', 'v3', 'c8']] = dfhyp['Point-of-View'].str.rsplit('|', expan
    dfhyp.drop(['v1', 'v2', 'v3', 'c4', 'c8'], axis=1, inplace=True)
    dfhyp.rename(columns={'c3': 'v1', 'c1': 'v3', 'c2': 'v2'}, inplace=True)
    #Удаляем лишние символы - так, чтобы привязаться к коду филиала
    dfhyp['v1'] = dfhyp['v1'].str.replace('E_TBD_', '', regex=False)
    dfhyp['v1'] = dfhyp['v1'].str.replace('E_', '', regex=False)
```



Сценарий подготовки данных



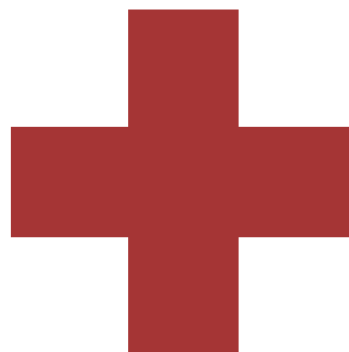
© 2018 «Росгосстрах»

- **Избавление от «проблемы Excel»** – централизованная реализация логики предобработки
- **Снижение трудозатрат** – весь цикл работы с данными в одной среде без кодирования
- **Вовлеченность сотрудников** – самостоятельная реализация сложной логики анализа без привлечения IT
- **Обогащение данных** – повышение информативности за счет использования в сценариях данных из множества источников

От данных до принятия решений



- Подготовка данных
- Обогащение данных
- Расчет показателей
- Моделирование



- Визуализация
- Дашборды
- Доставка данных
- Исследование

Что дальше?



1. Актуарные расчёты
2. Построение моделей анализа рисков
3. Обогащение данных из внешних источников
4. Прогнозирование для задач планирования
5. Клиентская аналитика

Вопросы?