gRUG

Консультационная группа АТК
ATK Consulting Group
www.atkcg.ru

Qlik

On-Demand Application Generation

теория и практика использования

Сергей Полехин

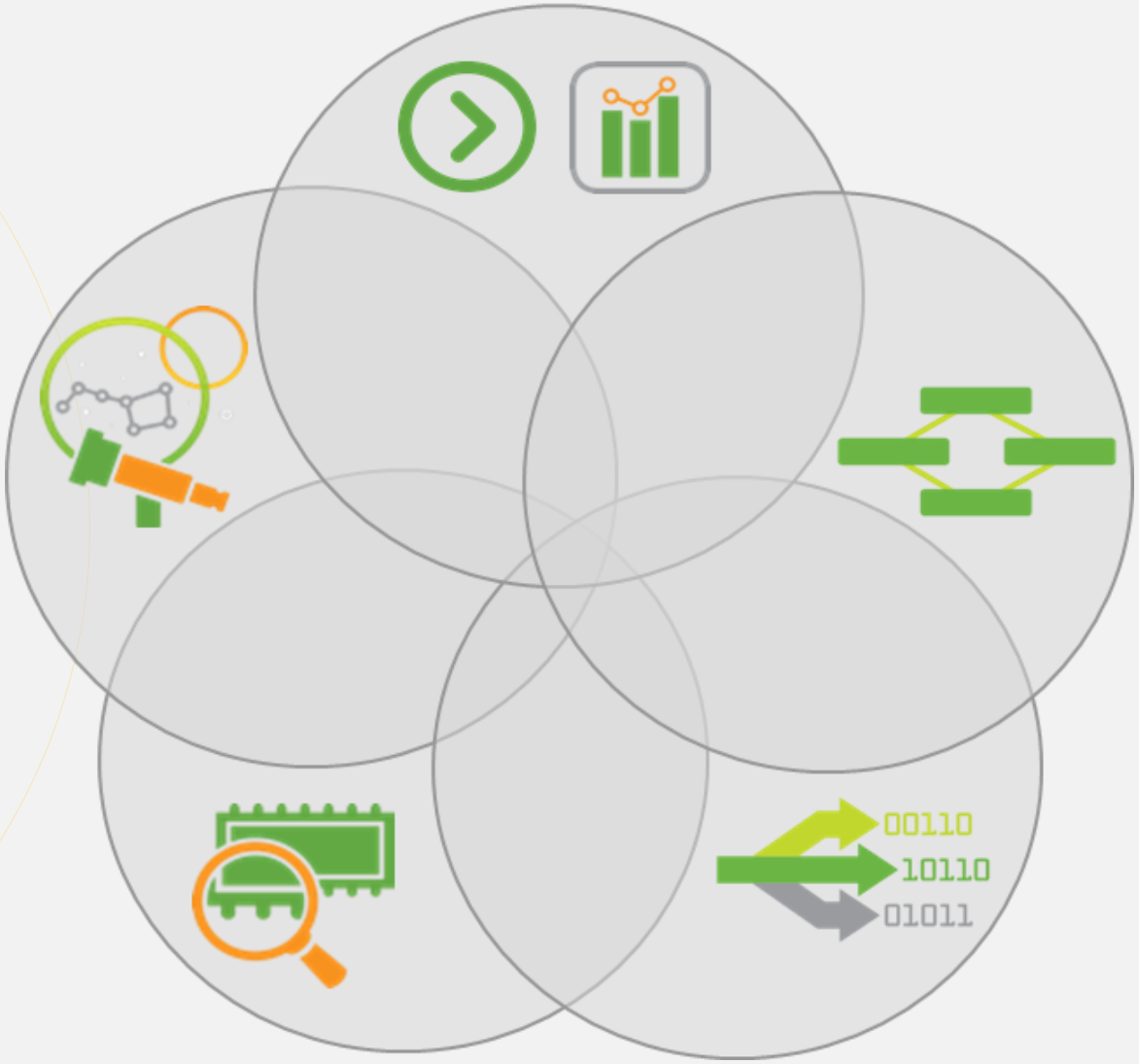
Qlik®

На чём фокусируемся

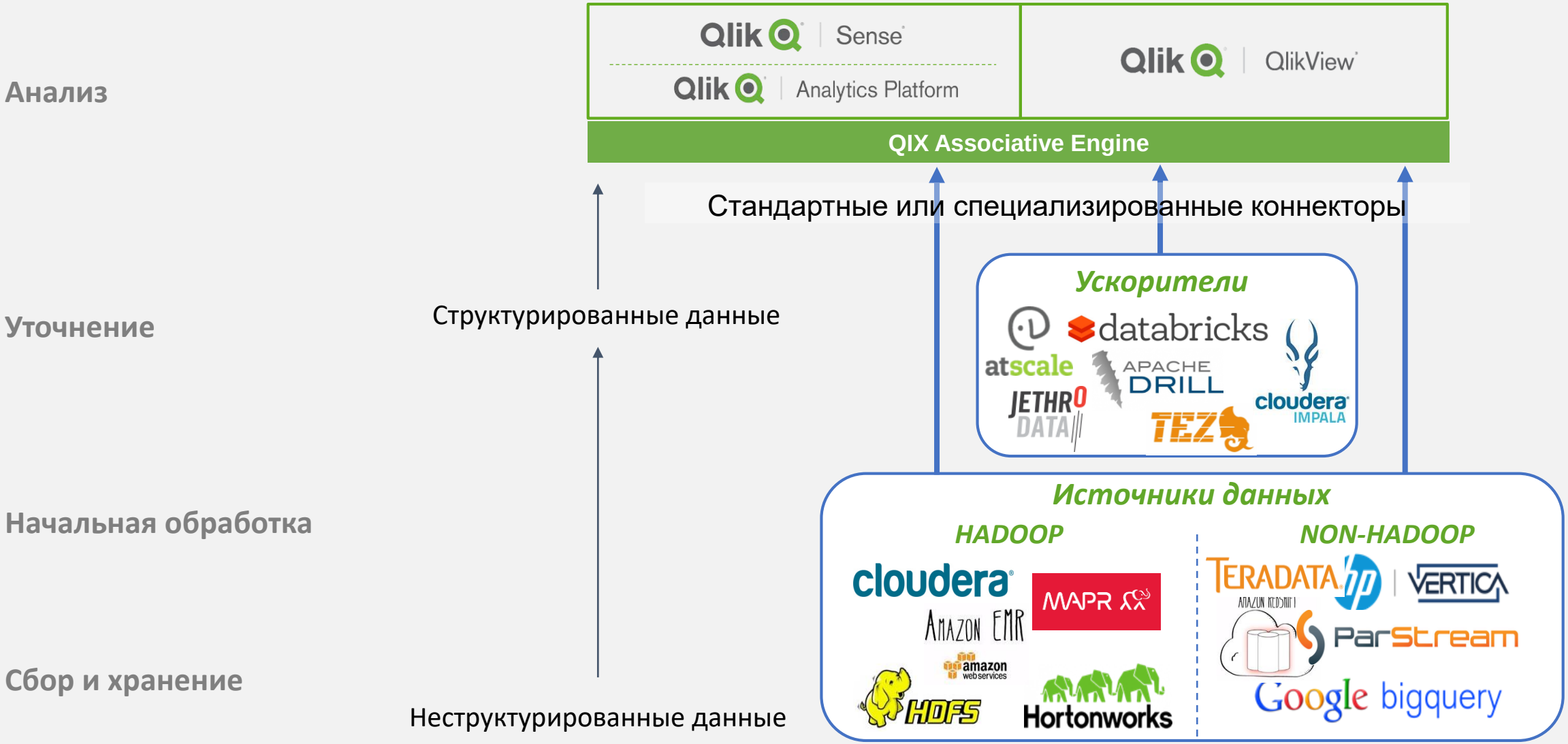
- Подходы к работе с большими объемами данных
- On-Demand Application Generation (ODAG)
  - Демонстрация

gRUG

Подходы к работе с большими объемами данных



Картина мира сегодня



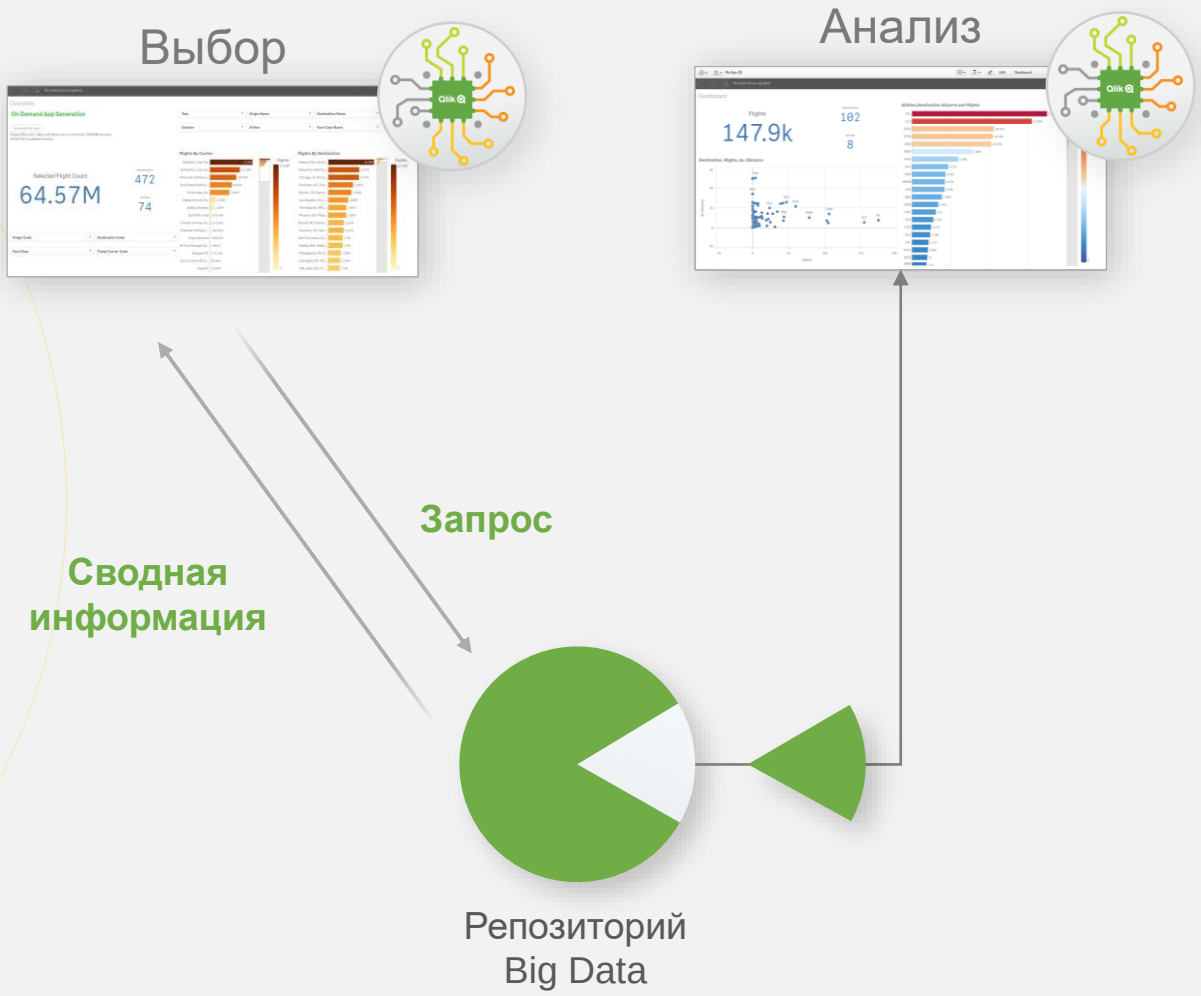
Методы работы с Big Data в Qlik

- *Разнообразие объемов и степень сложности данных наилучшим образом учитываются с использованием различных методов*
- ***Методы могут комбинироваться для удовлетворения специфических требований***
- *Методы различаются по сложности развертывания*

Объемы данных
- Количество (строки)
- Измерения (столбцы)
- Мощность (степень уникальности)

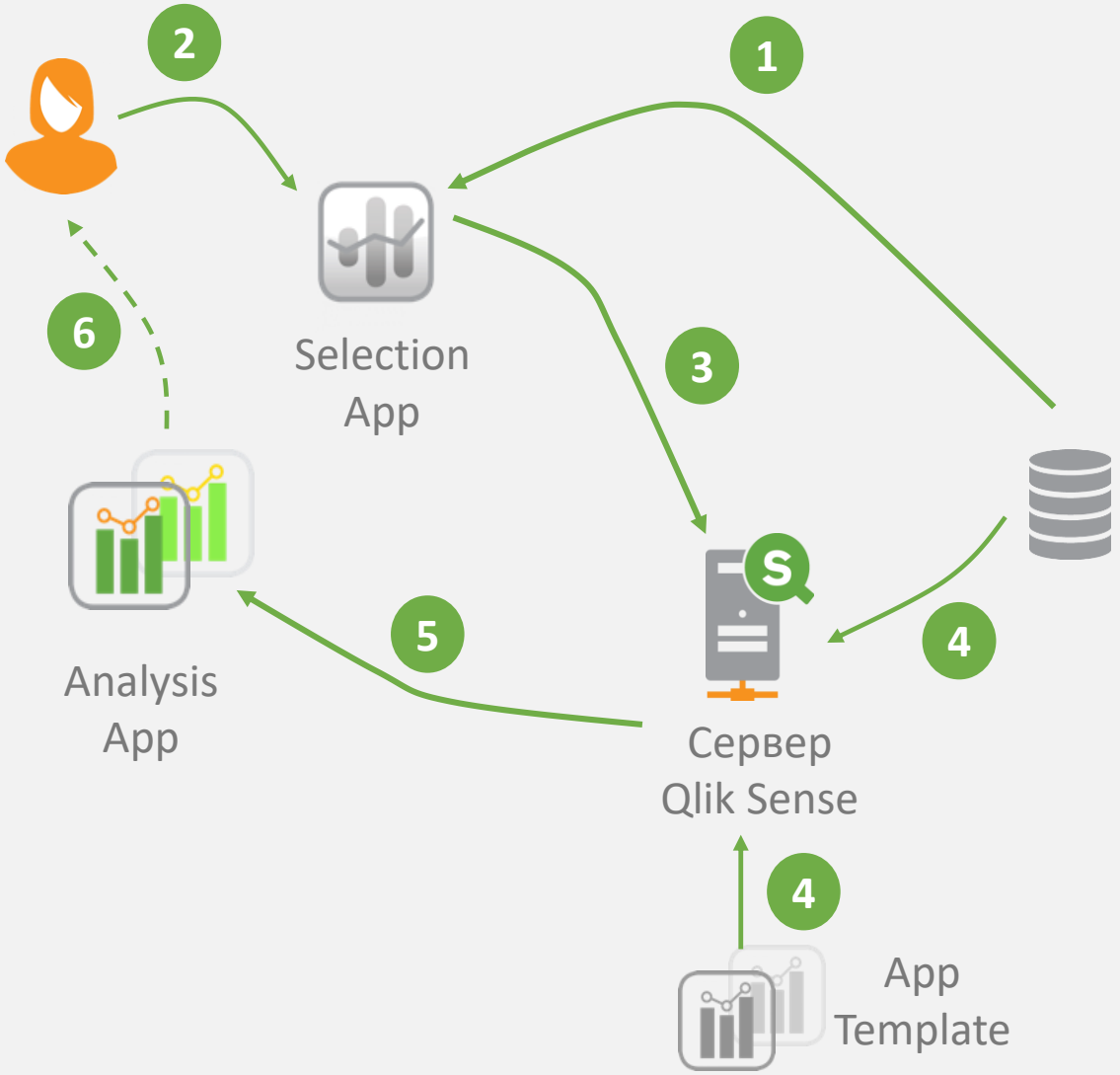
Сложность приложения
- Сложность вычислений (например, анализ множеств)
- Количество объектов визуализации

Генерация приложений «по требованию» (ODAG)



ODAG – идея реализации

1. Приложение для отборов (Selection App) наполняется данными измерений и объектами визуализации обеспечивающими высокий уровень агрегирования; Создается шаблон приложения для анализа данных
2. Пользователь выбирает требуемый срез данных в приложении с использованием и интуитивно понятных действий в привычных объектах визуализации
3. Пользователь инициирует создание приложения для анализа данных (Analysis App)
4. Приложение для анализа данных автоматически создаётся на основе шаблона, наполняется данными
5. Приложение для анализа данных публикуется
6. Если пользователю требуется любой другой срез данных, то он просто повторяет шаги 2-5



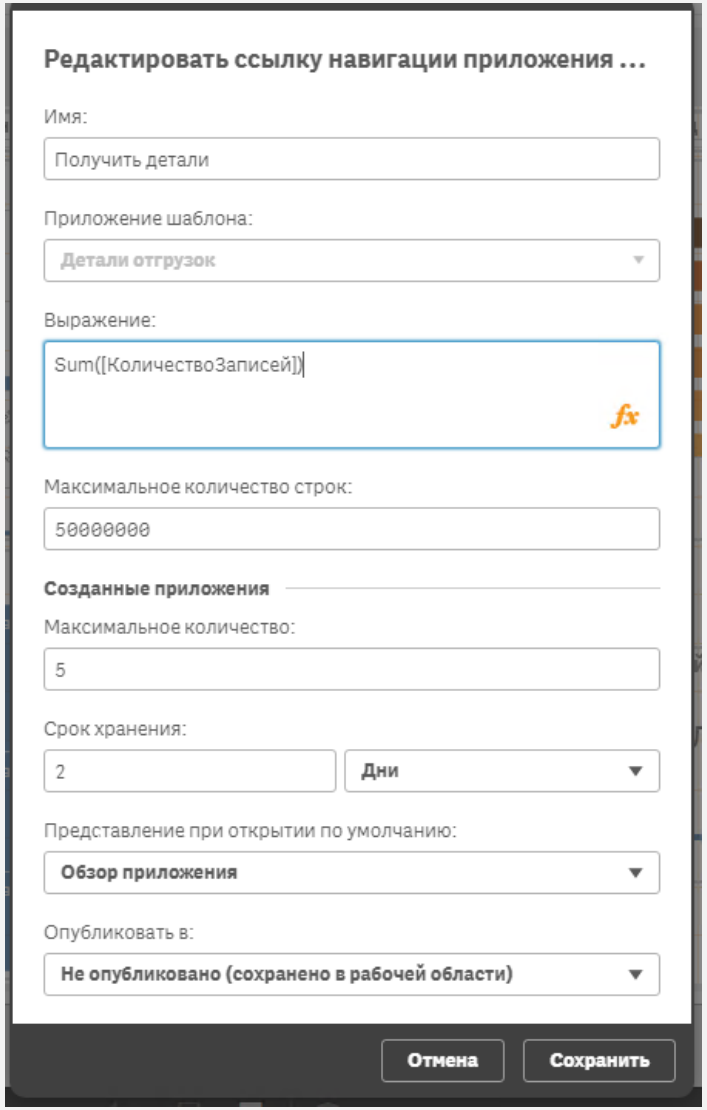
ODAG – идея управляемости

Ресурсоёмкость приложения

- Сложность расчетов
- Количество записей (строк)

Объем данных на сервере

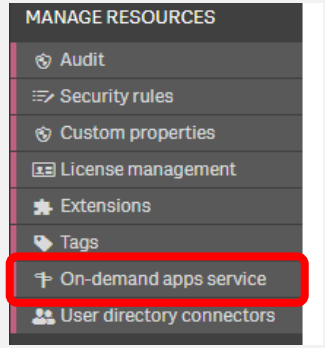
- Количество приложений
- Срок хранения

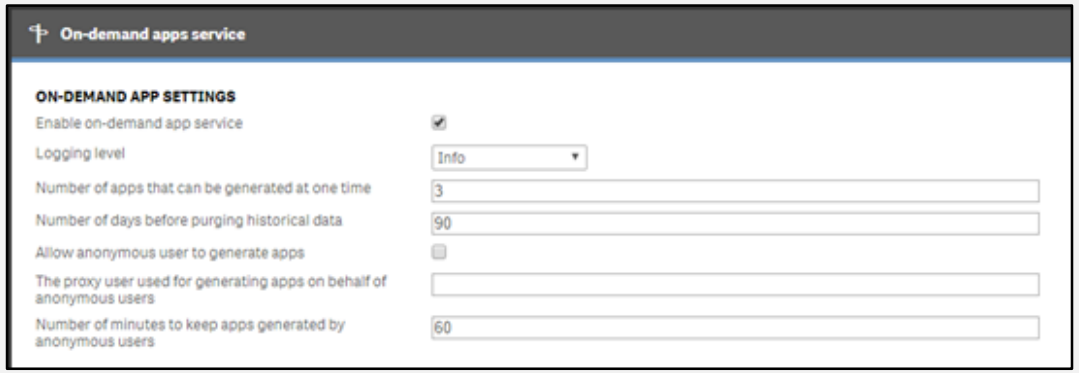


ODAG – реализация

Составные элементы решения

Серверная служба для автоматизации и контроля процесса





Сценарий загрузки приложения (Load Script)

- Стандартный функционал
- Ряд дополнительных соглашений расширяющих синтаксис

```
// DO NOT ALTER THIS SUBROUTINE
SUB ExtendQVDWhere(Name, ValVarName)
  LET T = Name & '_COLNAME';
  LET ColName = $(T);
  LET Values = $(ValVarName);
  IF (len(Values) > 0) THEN
        IF len(WHERE_PART) > 0 THEN
        LET WHERE_PART = '$(WHERE_PART) AND mixmatch([$(ColName)],$(Values) )';
    ELSE
        LET WHERE_PART = ' WHERE mixmatch([$(ColName)],$(Values))';
    ENDIF
  ENDIF
END SUB;

// DO NOT ALTER THIS SUBROUTINE
SUB ExtendSQLWhere(Name, ValVarName)
  LET T = Name & '_COLNAME';
  LET ColName = $(T);
  LET Values = $(ValVarName);
  IF (len(Values) > 0) THEN
    IF len(WHERE_PART) > 0 THEN
        LET WHERE_PART = '$(WHERE_PART) AND $(ColName) IN ( $(Values) )';
    ELSE
        LET WHERE_PART = ' WHERE $(ColName) IN ( $(Values) )';
    ENDIF
  ENDIF
END SUB;

// DO NOT ALTER THIS SUBROUTINE
SUB BuildValueList(VarName, TableName, ColName, QuoteChrNum)
  IF ($(QuoteChrNum) = 0) THEN
    LET LOADEXPR = 'Concat($(ColName),' & chr(39) & ',' & chr(39) & ') AS CombinedData';
  ELSE
    LET CHREXPR = ' chr(' & '$(QuoteChrNum)' & ') ';
    LET LOADEXPR = 'Concat( $(CHREXPR) & $(ColName) & $(CHREXPR)' & ',' & chr(39) & ',' & chr(39) & ') AS CombinedData';
  ENDIF
  _TempTable:
  LOAD $(LOADEXPR) Resident $(TableName);
  Let vNoOfRows = NoOfRows('_TempTable');
  IF $(vNoOfRows)> 0 THEN
    LET $(VarName) = Peek('CombinedData',0,'_TempTable');
  ENDIF
  drop table _TempTable;
  drop table '$(TableName)';
END SUB;
```

От идеи до реализации – не один шаг

или что ещё нужно учитывать?

- QIX имеет ограничения на количество значений, возможных для конкатенации в одном операторе LET
  - Хранение отбираемых значений не в переменной, а в таблице
    - См. описание: https://help.qlik.com/en-US/sense/June2018/Subsystems/Hub/Content/LoadData/bind-expressions-OnDemand-template-app.htm

От идеи до реализации – не один шаг

или для чего расширять синтаксис?

- Binding expressions (цвет имеет значение ☺)
  - Отбор в конкретном поле (зелёный) ≠ связанным значениям (белый)
    - $(ods_FIELD_NAME) или $(od_FIELD_NAME) – отобранные значения (Selected)
    - $(odo_FIELD_NAME) – связанные значения (Optional)
    - $(odso_FIELD_NAME) – отобранные или связанные значения (Selected, Optional)
- Специфика отборов
  - Учет различных вариантов синтаксиса SQL: Where, Where … between
    - $(od_FIELD_NAME)[2] – требуется именно два значения в поле
    - $(od_FIELD_NAME)[2-4] – требуется от двух до четырёх значений в поле
    - $(od_FIELD_NAME)[2+] – требуется от двух значений в поле
    - $(od_FIELD_NAME)[2-] – требуется не более двух значений в поле

От идеи до реализации – не один шаг

или как учитывать особенности реализации баз данных?

- Не все БД поддерживают автоматическое преобразование типов
  - Требование числового представления данных
    - $(od_FIELD_NAME_n)
- Не все БД в качестве разделителей используют одни и те же символы
  - Требование использования конкретных разделителей
    - $(od_FIELD_NAME){"quote": "|", "delimiter": ";"}
    - Пример результата: |Январь|;|Февраль|;|Март|

Демонстрация

Вопросы?