

# Миграция с SQL Server на PostgreSQL

Миграция может выполняться для целей тестирования, в этом случае необходимо выполнить обезличивание данных.

Обезличивание данных состоит из двух этапов:

1. Анализ данных на наличие персональных данных. Анализ заключается в поиске имен людей и согласованной их замене на обезличенные идентификаторы.
2. Экспорт данных из SQL Server с одновременной заменой и сокрытием персональных данных в скалярных полях и полях, содержащих XML

Экспорт данных может осуществляться без процедуры обезличивания данных.

Экспорт данных может производиться непосредственно из таблиц и из запросов.

При импорте данных используются BULK INSERT (SQL Server) или BINARY COPY (PostgreSQL) для ускорения загрузки данных.

При операции экспорта в БД назначения рекомендуется удалить внешние ключи и все индексы (кроме первичных ключей)

Для целей миграции созданы две утилиты:

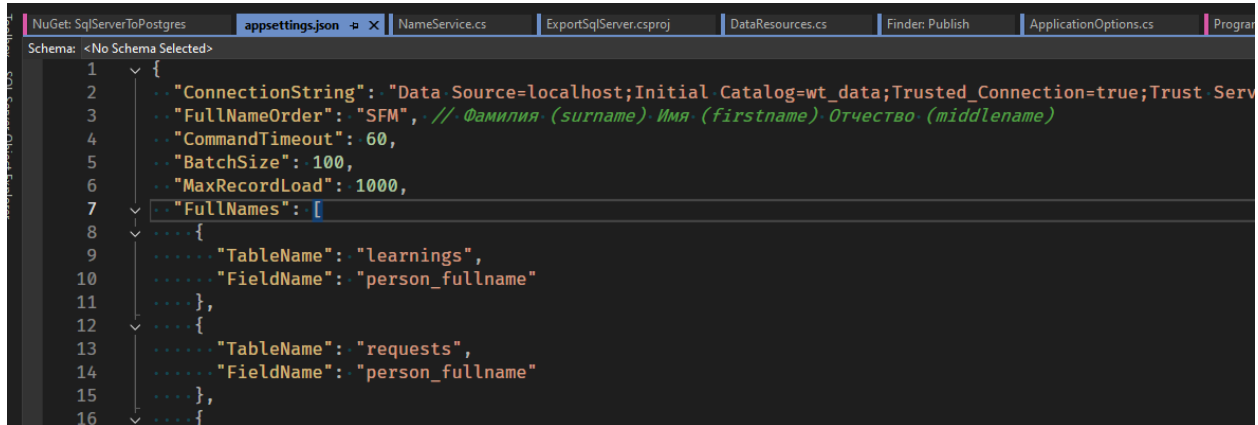
1. Finder – поиск имен и присвоение уникальных идентификаторов
2. SqlServerToPostgres – утилита экспорта данных из SQL Server.

Утилиты можно скомпилировать и выполнять как на Linux, так и на Windows.

# Обезличивание и миграция данных

## Finder

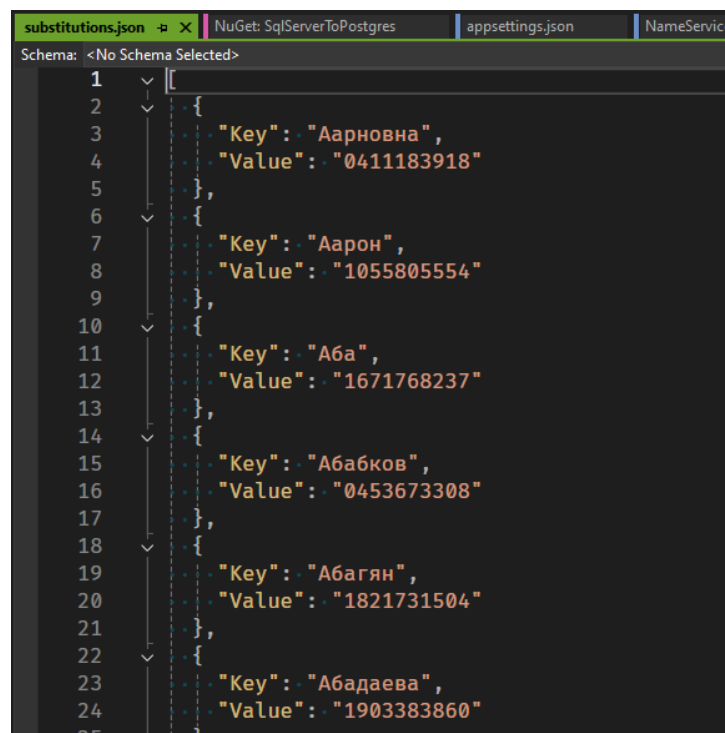
Настройки утилиты содержатся в файле appsettings.json



```
1 {
2   .. "ConnectionString": "Data Source=localhost;Initial Catalog=wt_data;Trusted_Connection=true;Trust Serv
3   .. "FullNameOrder": "SFM", // Фамилия (surname) Имя (firstname) Отчество (middlename)
4   .. "CommandTimeout": 60,
5   .. "BatchSize": 100,
6   .. "MaxRecordLoad": 1000,
7   .. "FullNames": [
8     {
9       .. "TableName": "learnings",
10      .. "FieldName": "person_fullname"
11    },
12    {
13      .. "TableName": "requests",
14      .. "FieldName": "person_fullname"
15    },
16  ]
17 }
```

В настройка необходимо указать строку подключения к БД, шаблон полного имени (SFM|FMS), а также другие технические параметры необходимые для успешного выполнения утилиты.

В результате работы утилита создает файл substitutions.json, который содержит пары имени и соответствующего идентификатора, на который будет производится замена соответствующего имени при экспорте.



```
1 [
2   {
3     .. "Key": "Аарновна",
4     .. "Value": "0411183918"
5   },
6   {
7     .. "Key": "Аарон",
8     .. "Value": "1055805554"
9   },
10  {
11    .. "Key": "Аба",
12    .. "Value": "1671768237"
13  },
14  {
15    .. "Key": "Абабков",
16    .. "Value": "0453673308"
17  },
18  {
19    .. "Key": "Абагян",
20    .. "Value": "1821731504"
21  },
22  {
23    .. "Key": "Абадаева",
24    .. "Value": "1903383860"
25  }
26 ]
```

# SqlServerToPostgres

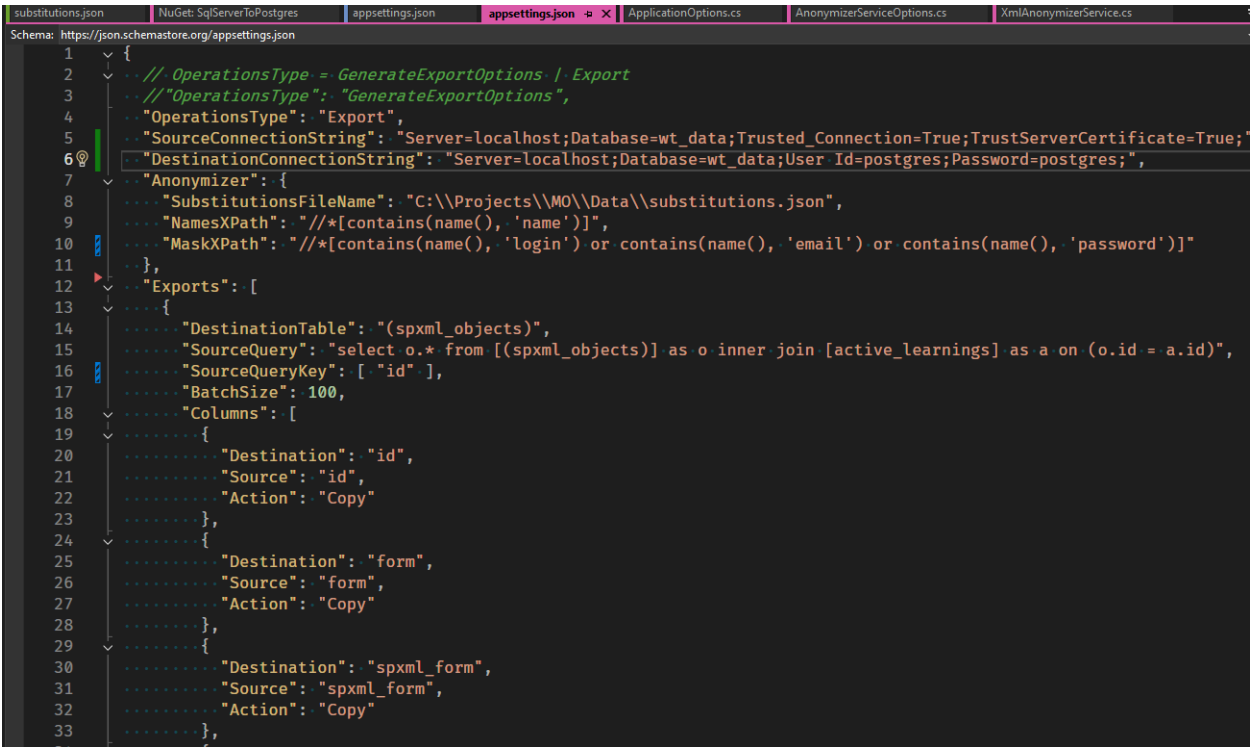
Данная утилита может работать в двух режимах:

- 1. GenerateExportOptions – в этом режиме автоматически создается файл мэппинга таблиц и полей из БД источника с БД приёмника. Результатом работы утилиты является файл exports.json, данные из которого могут быть использованы в качестве настроек секции Expors (appsettings.json)
- 2. Export – режим экспорта данных в соответствии с настройками

Для упрощения процедуры экспорта в БД приёмнике рекомендуется удалить все внешние ключи и пересоздать их после завершения процесса миграции данных.

Для обеспечения гибкости и снижения требований к объему БД миграцию данных можно производить частями.

Настройки:



```
1 {
2   // OperationsType = GenerateExportOptions | Export
3   // "OperationsType": "GenerateExportOptions",
4   "OperationsType": "Export",
5   "SourceConnectionString": "Server=localhost;Database=wt_data;Trusted_Connection=True;TrustServerCertificate=True;"
6   "DestinationConnectionString": "Server=localhost;Database=wt_data;User Id=postgres;Password=postgres;",
7   "Anonymizer": {
8     "SubstitutionsFileName": "C:\\Projects\\M0\\Data\\substitutions.json",
9     "NamesXPath": "//*[contains(name(), 'name')]",
10    "MaskXPath": "//*[contains(name(), 'login') or contains(name(), 'email') or contains(name(), 'password')]"
11  },
12  "Exports": [
13    {
14      "DestinationTable": "(spxml_objects)",
15      "SourceQuery": "select o.* from [(spxml_objects)] as o inner join [active_learnings] as a on (o.id = a.id)",
16      "SourceQueryKey": [ "id" ],
17      "BatchSize": 100,
18      "Columns": [
19        {
20          "Destination": "id",
21          "Source": "id",
22          "Action": "Copy"
23        },
24        {
25          "Destination": "form",
26          "Source": "form",
27          "Action": "Copy"
28        },
29        {
30          "Destination": "spxml_form",
31          "Source": "spxml_form",
32          "Action": "Copy"
33        }
34      ]
35    }
36  ]
37 }
```

Свойство	Тип	Комментарий
OperationsType	строка	GenerateExportOptions   Export

Свойство	Тип	Комментарий
<b>SourceConnectionString</b>	строка подключения к БД источника данных	
<b>DestinationConnectionString</b>	строка подключения к БД приёмника	
<b>Anonymizer</b>		
<b>SubstitutionsFileName</b>	строка	путь к файлу подстановок (Finder)
<b>NamesXPath</b>	строка	XPath к элементам потенциально содержащему имена сотрудников. Найденные имена в этих элементах будут заменены на идентификаторы из файла (substitutions.json)
<b>MaskXPath</b>	строка	XPath к элементам потенциально содержащему конфиденциальную информацию (login, email, password и т.п.). текст в таких элементах будет заменен на «*»
<b>Exports</b>	массив объектов	
<b>DestinationTable</b>	строка	имя таблицы приёмника данных
<b>SourceTable</b>	строка	имя таблицы источника данных.
<b>SourceQuery</b>	строка	SQL запрос – источник данных, если не указана таблица источник
<b>SourceQueryKey</b>	массив строк	ключи для сортировки и пакетирования данных из запроса
<b>BatchSize</b>	число	размер пакета
<b>Columns</b>	массив объектов	мэппинг полей источника и приёмника
<b>Destination</b>	строка	Поле в таблице приёмнике
<b>Source</b>	строка	Поле в таблице запросе источнике

Свойство	Тип	Комментарий
Action	перечисление	Skip – пропускает поле, Copy – копирует значение из источника в приёмник, Convert – использует специальный конвертер (реализовано XmlAnonymizerService и TextAnonymizerService) TypeConvert – производит конвертацию типа данных, SetNull – устанавливает значение NULL, MaskWithChar – маскирует поле «*»
Parameter	строка	Параметр зависит от Action

Пример результата работы SqlServerToPostgress

```

Starting export select o.* from [(spxml_objects)] as o inner join [active_learnings] as a on (o.id = a.id) -> (spxml_objects)
Exported records: 100
Exported records: 200
Exported records: 300
Exported records: 400
Exported records: 500
Exported records: 600
Exported records: 700
Exported records: 800
Exported records: 900
Exported records: 1000
Exported records: 1100
Exported records: 1105
Starting export active_learning -> active_learning
Exported records: 100
Exported records: 200
Exported records: 300
Exported records: 400
Exported records: 500
Exported records: 600
Exported records: 700
Exported records: 800
Exported records: 900
Exported records: 1000
Exported records: 1100
Exported records: 1105
Starting export active_learnings -> active_learnings
Exported records: 100
Exported records: 200
Exported records: 300
Exported records: 400
Exported records: 500
Exported records: 600
Exported records: 700
Exported records: 800
Exported records: 900
Exported records: 1000
Exported records: 1100
Exported records: 1105

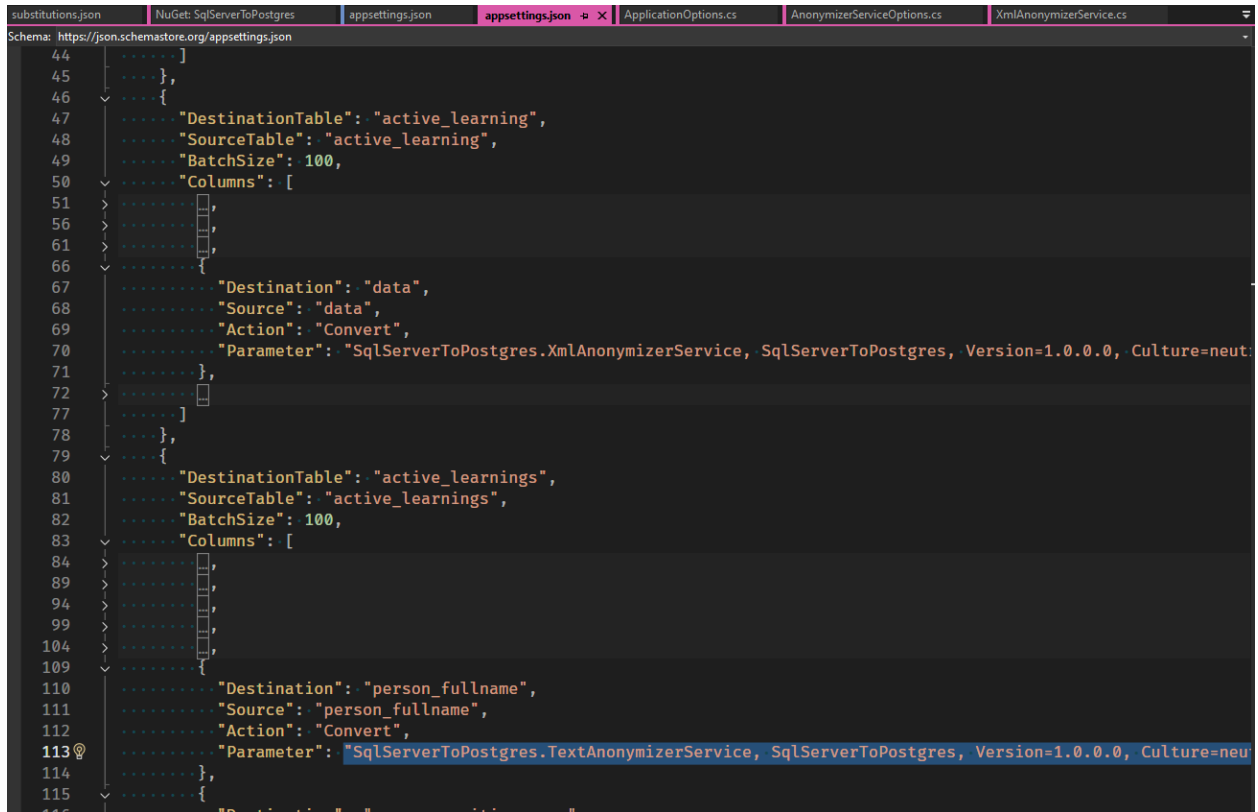
```

Для обезличивания полей, содержащих XML, используется конвертер:

"SqlServerToPostgres.XmlAnonymizerService, SqlServerToPostgres, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null"

Для обезличивания скалярных полей используется конвертер:

"SqlServerToPostgres.TextAnonymizerService, SqlServerToPostgres, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null"



```
substitutions.json | NuGet: SqlServerToPostgres | appsettings.json | appsettings.json x | ApplicationOptions.cs | AnonymizerServiceOptions.cs | XmlAnonymizerService.cs
Schema: https://json.schemastore.org/appsettings.json
44 | .....]
45 | .....},
46 | .....{
47 | .....  "DestinationTable": "active_learning",
48 | .....  "SourceTable": "active_learning",
49 | .....  "BatchSize": 100,
50 | .....  "Columns": [
51 | > .....  ],
56 | > .....  ],
61 | > .....  ],
66 | .....{
67 | .....  "Destination": "data",
68 | .....  "Source": "data",
69 | .....  "Action": "Convert",
70 | .....  "Parameter": "SqlServerToPostgres.XmlAnonymizerService, SqlServerToPostgres, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null",
71 | .....},
72 | > .....]
77 | .....}
78 | .....},
79 | .....{
80 | .....  "DestinationTable": "active_learnings",
81 | .....  "SourceTable": "active_learnings",
82 | .....  "BatchSize": 100,
83 | .....  "Columns": [
84 | > .....  ],
89 | > .....  ],
94 | > .....  ],
99 | > .....  ],
104 | > .....  ],
109 | .....{
110 | .....  "Destination": "person_fullname",
111 | .....  "Source": "person_fullname",
112 | .....  "Action": "Convert",
113 | .....  "Parameter": "SqlServerToPostgres.TextAnonymizerService, SqlServerToPostgres, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null",
114 | .....},
115 | .....{
116 | .....  "Destination": "person_email_address",
```

Если обезличивание не требует, в настройках Action = “Convert” следует заменить на Action = “Copy”