Hi,

Here I have added my experience on data migration process which I have involved.

I don't have SQL Server & Postgres in my local, because of that reason I could not able to migrate the sample database.

==========

Prerequisites

- Validate data types used in source database and identify the most suitable data types on target.
- Take care of the compatibility level.
- Select suitable OS & hardware on Target.
- User management
- Check for below special/ advanced features used in source SQL Server
    - Any special data types or user defined data types in source
    - User management
    - Encryption
    - Data masking
    - CDC/ changing tracking.
    - External stored procedures
    - SSIS packages/ SQL Jobs
    - Backup / recovery
    - HA & DR
    - Etc…..

Check the compatibility of these features, and work on each feature with script/tool to support on target database.

Approach 1:

AWS SCT (Schema conversion tool) & DMS (Data migration Service) are very robust and user friends to migrate schema from SQL Server to AWS PostgreSQL

Cross verifies the source and target data mappings after SCT map the data types.

Approach 2:

Create SSIS package.

- Install ODBC drivers for Postgres
- Add data flow task
- Create source & target components.
- Map the columns with appropriate data types.
- Setup error handling
- Run the package.

Unit testing

Check tables, data and other objects (logins/UDT /SPS/UDF etc…) with count

For data validation

Check the tables row count.

Take random samples from source , which should cover all data types used from different combination of tables and cross verify on target.

test all special features which we noted earlier.

Test data by Comparing source and target using functions like Len, max , min , sum, count etc.

If we want to test all records in source and target build an algorithm, use it on row level and store in a new column in source & target table(s) and compare.

Challenges:

Data truncation & Junk data insertions: This is solved by selecting most suitable data type in target

Unit testing: solved by generated an algorithm and implemented both source & target, compared for identifying data issues.

SQL Jobs: Implemented Con jobs

Some inbuild function of source not exists in target, so built UDFs on tagget.