# AWS Schema Conversion Tool

AWS

# Agenda

## AWS Schema Conversion Tool

- Overview
- Getting started
- Menu navigation
- Settings

# Overview

# AWS Schema Conversion Tool (SCT)

The AWS Schema Conversion Tool helps automate many database schema and code conversion tasks when migrating to a new database engine.

**Features**

Schema conversion between database engines

Database Migration Assessment report for choosing the best target engine

Code browser that highlights places where manual edits are required



aws

# Pricing and platform support

**$0**

You can download AWS Schema Conversion Tool for your platform of choice


Microsoft Windows


Apple Mac


Fedora Linux (rpm)


Ubuntu Linux (deb)

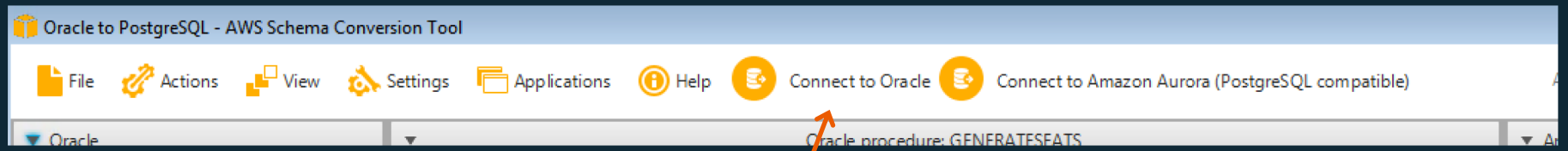


aws

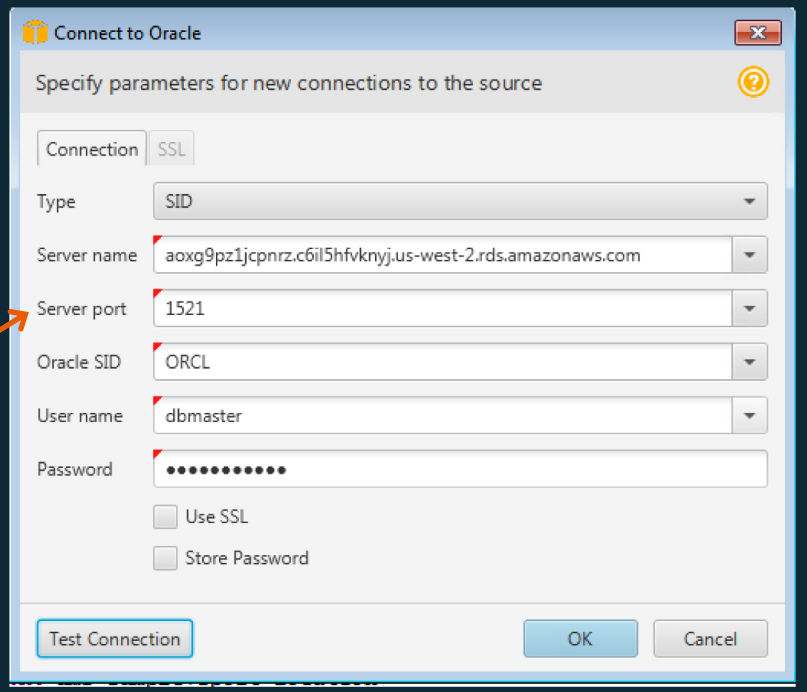
# Getting Started



# Connecting to your database




1. Create a new project
2. Connect to your source database
3. Enter the connection details






aws

# Migration assessment report

**Database Migration Assessment Report**

Source Database: RDS_ADMINISTRATION.rds_administrator@ec2-54-172-36-60.compute-1.amazonaws.com:8192:ORCL

Oracle Database 12c Enterprise Edition 12.1.0.1.0 (64bit Production)

amazon web services

**Executive Summary**

We completed the analysis of your Oracle source database and estimate that 91% of the database storage objects and 100% of database code objects can be converted automatically or with minimal changes if you select Amazon Aurora as your migration target. Database storage objects include schemas, tables, columns, constraints, indexes, sequences, synonyms, user define types and types. Database code objects include functions, procedures, packages, triggers, views, materialized views, events, SQL scalar functions, SQL inline functions, SQL table functions, attributes, variables, constants, table types, public types, private types, cursors, exceptions, parameters and other objects. Based on our analysis of SQL syntax elements of your source database schema, we estimate that 99.9% of your entire database schema can be converted automatically to Amazon Aurora. To complete the migration, we recommend 597 conversion action(s) ranging from simple tasks to medium-complexity actions to significant conversion actions.

**Database Objects with Conversion Actions for Amazon Aurora**

Of the total 1,576 database storage object(s) and 155 database code object(s) in the source database, we were able to identify 1,427 (91%) database storage object(s) and 155 (100%) database code objects that can be converted automatically or with minimal changes to Amazon Aurora.

149 (9%) database storage object(s) required 149 significant user action(s) to complete the conversion.

**Figure: Conversion statistics for database storage objects**



**Figure: Conversion statistics for database code objects**

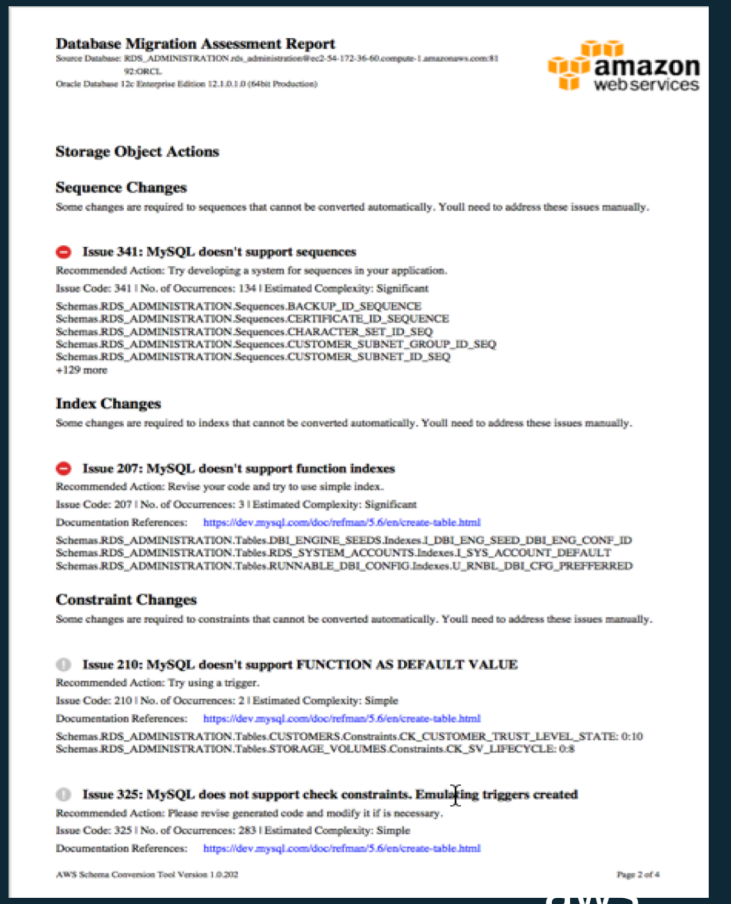


**Detailed Recommendations for Amazon Aurora Migrations**

If you choose to migrate your Oracle database to Amazon Aurora, we recommend the following actions.

AWS Schema Conversion Tool Version 1.0.202 Page 1 of 4

4. Run assessment report
5. Review executive summary
6. Review detailed instructions



**Database Migration Assessment Report**

Source Database: RDS_ADMINISTRATION.rds_administrator@ec2-54-172-36-60.compute-1.amazonaws.com:8192:ORCL

Oracle Database 12c Enterprise Edition 12.1.0.1.0 (64bit Production)

amazon web services

**Storage Object Actions**

**Sequence Changes**

Some changes are required to sequences that cannot be converted automatically. Youll need to address these issues manually.

**Issue 341: MySQL doesn't support sequences**

Recommended Action: Try developing a system for sequences in your application.

Issue Code: 341 | No. of Occurrences: 134 | Estimated Complexity: Significant

Schemas.RDS_ADMINISTRATION.Sequences.BACKUP_ID_SEQUENCE
Schemas.RDS_ADMINISTRATION.Sequences.CERTIFICATE_ID_SEQUENCE
Schemas.RDS_ADMINISTRATION.Sequences.CHARACTER_SET_ID_SEQ
Schemas.RDS_ADMINISTRATION.Sequences.CUSTOMER_SUBNET_GROUP_ID_SEQ
Schemas.RDS_ADMINISTRATION.Sequences.CUSTOMER_SUBNET_ID_SEQ
+129 more

**Index Changes**

Some changes are required to indexs that cannot be converted automatically. Youll need to address these issues manually.

**Issue 207: MySQL doesn't support function indexes**

Recommended Action: Revise your code and try to use simple index.

Issue Code: 207 | No. of Occurrences: 3 | Estimated Complexity: Significant

Documentation References: https://dev.mysql.com/doc/refman/5.6/en/create-table.html

Schemas.RDS_ADMINISTRATION.Tables.DBI_ENGINE_SEEDS.Indexes.I_DBI_ENG_SEED_DBI_ENG_CONF_ID
Schemas.RDS_ADMINISTRATION.Tables.RDS_SYSTEM_ACCOUNTS.Indexes.I_SYS_ACCOUNT_DEFAULT
Schemas.RDS_ADMINISTRATION.Tables.RUNNABLE_DBI_CONFIG.Indexes.U_RNBL_DBI_CFG_PREFFERRED

**Constraint Changes**

Some changes are required to constraints that cannot be converted automatically. Youll need to address these issues manually.

**Issue 210: MySQL doesn't support FUNCTION AS DEFAULT VALUE**

Recommended Action: Try using a trigger.

Issue Code: 210 | No. of Occurrences: 2 | Estimated Complexity: Simple

Documentation References: https://dev.mysql.com/doc/refman/5.6/en/create-table.html

Schemas.RDS_ADMINISTRATION.Tables.CUSTOMERS.Constraints.CK_CUSTOMER_TRUST_LEVEL_STATE: 0:10
Schemas.RDS_ADMINISTRATION.Tables.STORAGE_VOLUMES.Constraints.CK_SV_LIFECYCLE: 0:8

**Issue 325: MySQL does not support check constraints. Emulating triggers created**

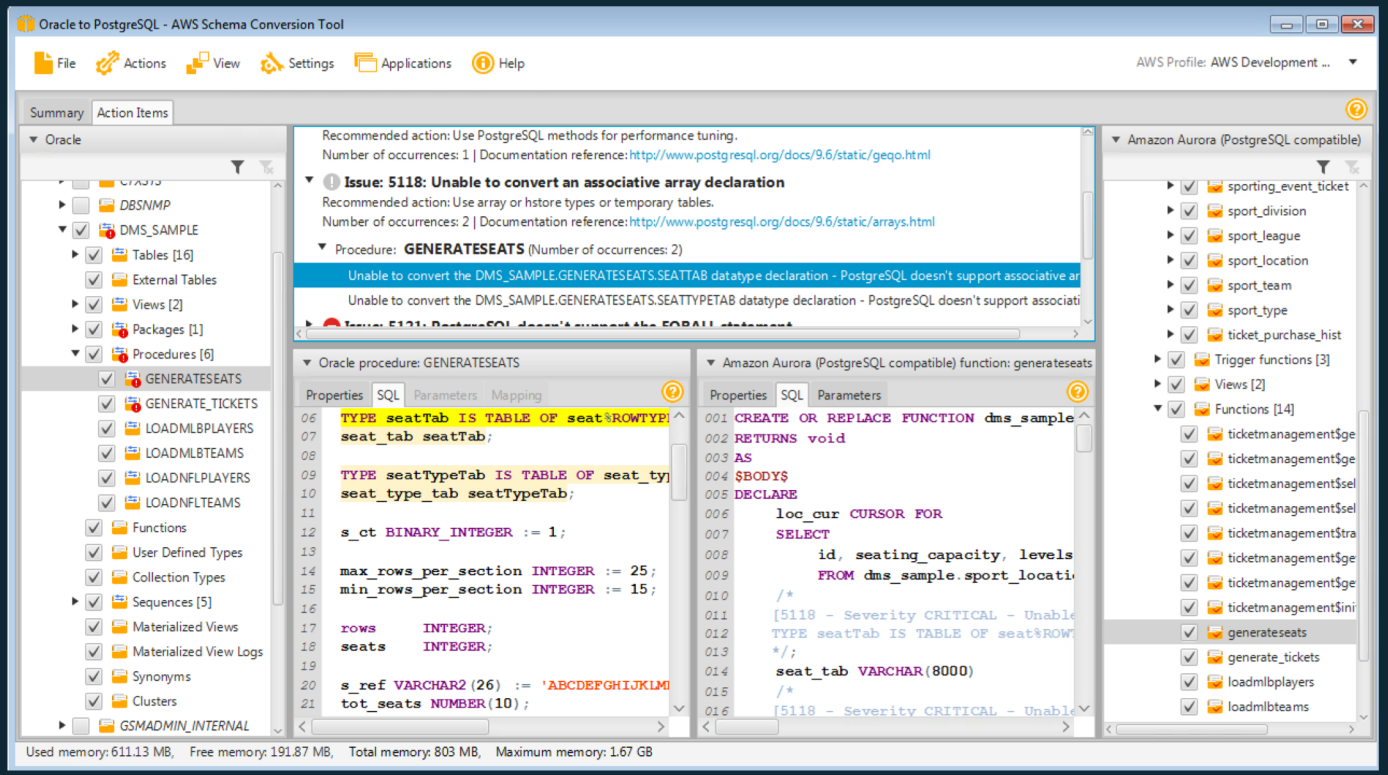
Recommended Action: Please revise generated code and modify it if is necessary.

Issue Code: 325 | No. of Occurrences: 283 | Estimated Complexity: Simple

Documentation References: https://dev.mysql.com/doc/refman/5.6/en/create-table.html

AWS Schema Conversion Tool Version 1.0.202 Page 2 of 4

# Convert schema and code objects

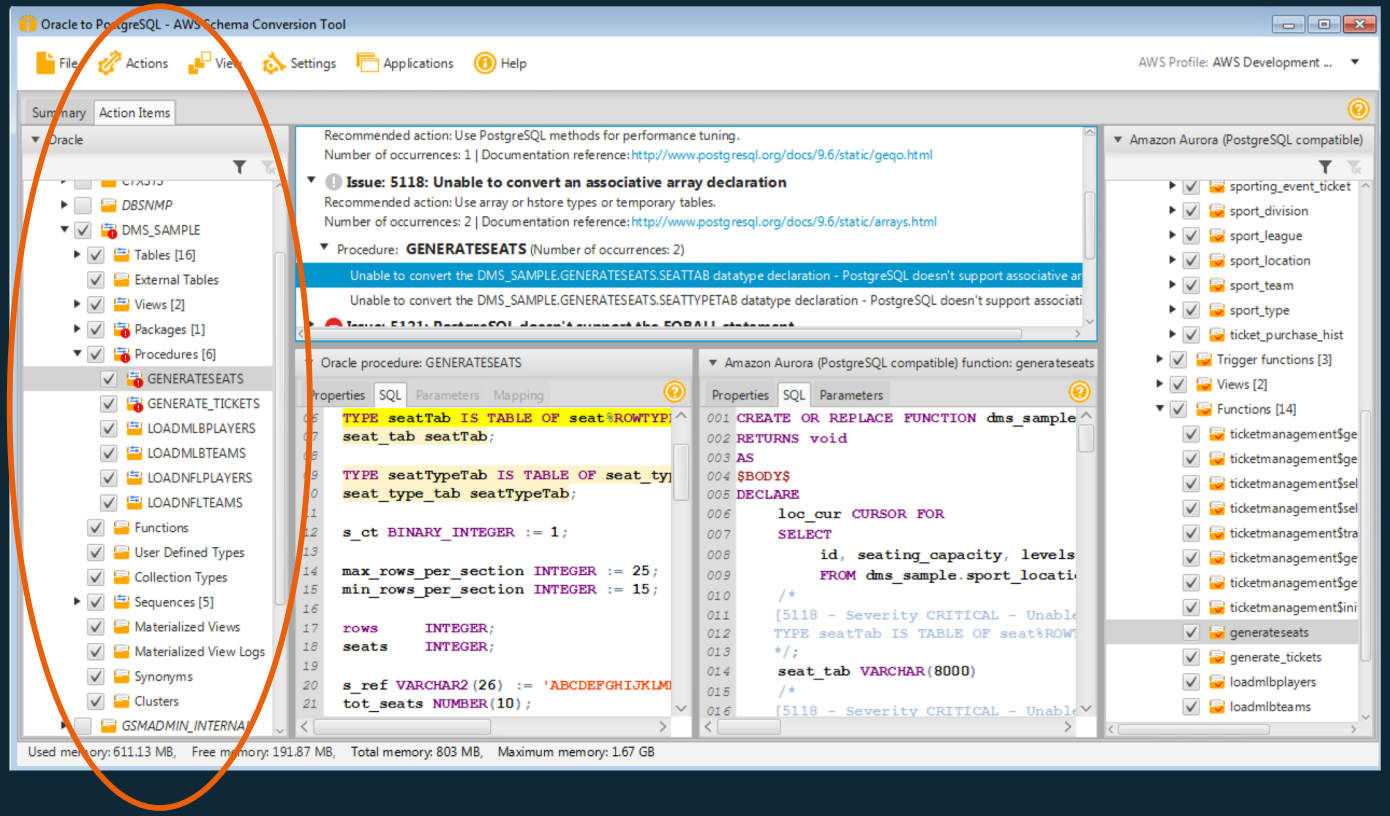



- Functions
- Indexes
- Packages
- Tables
- Triggers
- Schemas
- Sequences
- Stored Procedures
- Synonyms
- User Defined Types
- Views



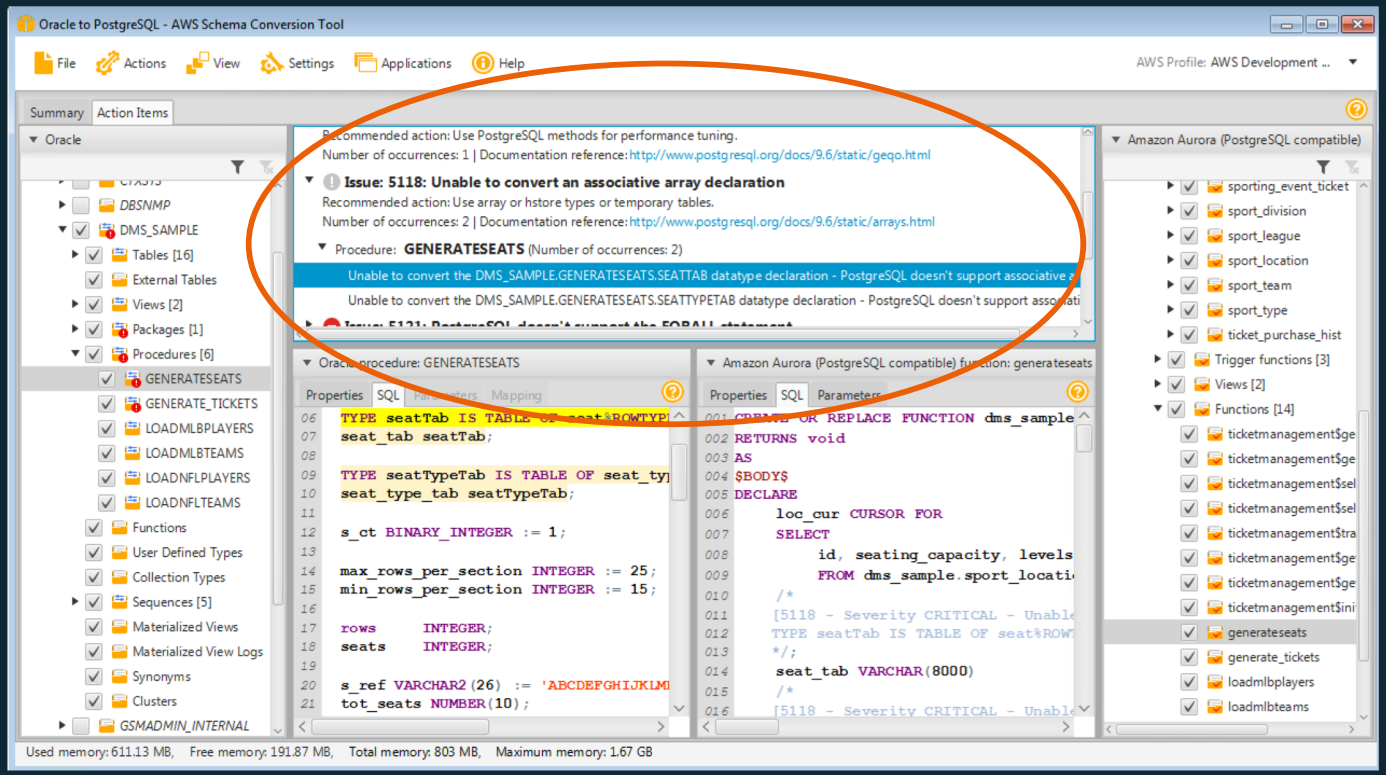
aws

# Tree view of objects



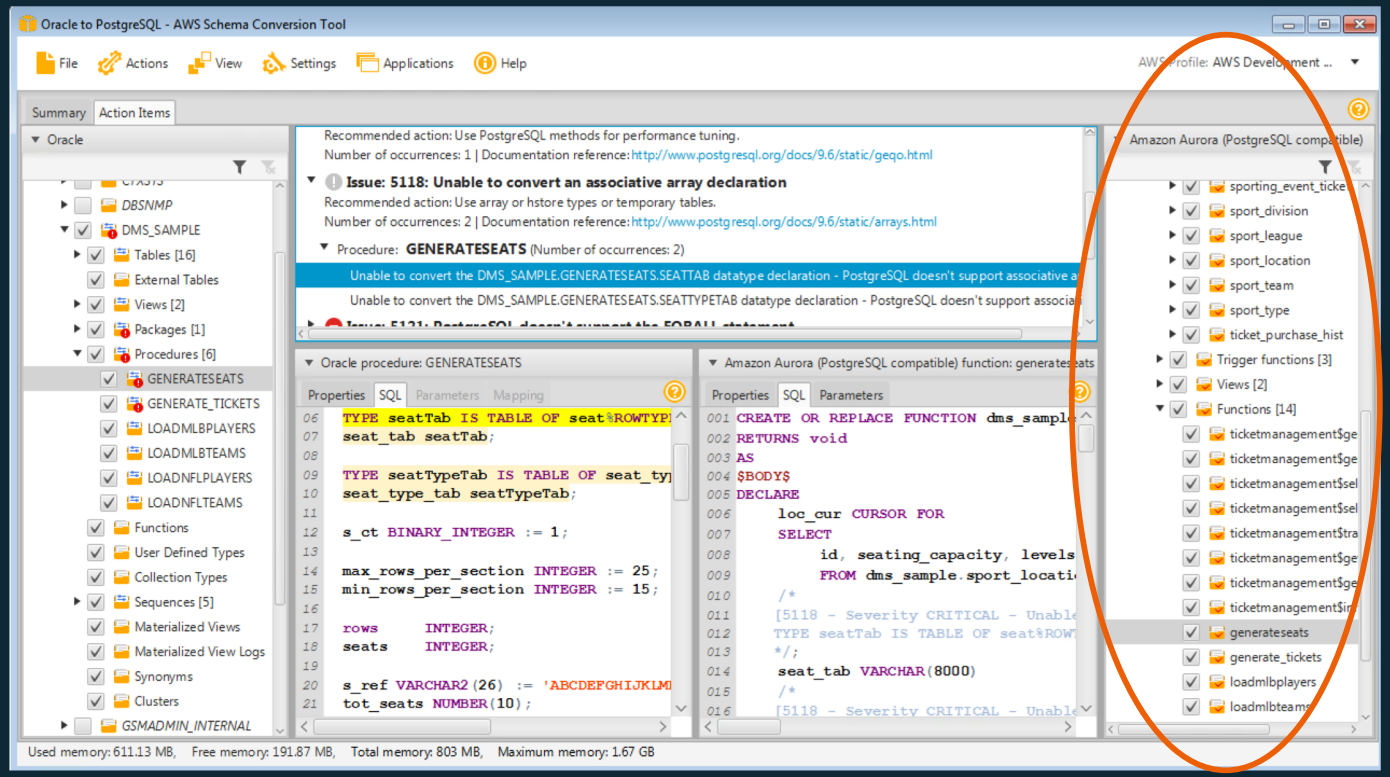

- Tree view of source
- Lazy loaded objects

# Actions at a glance



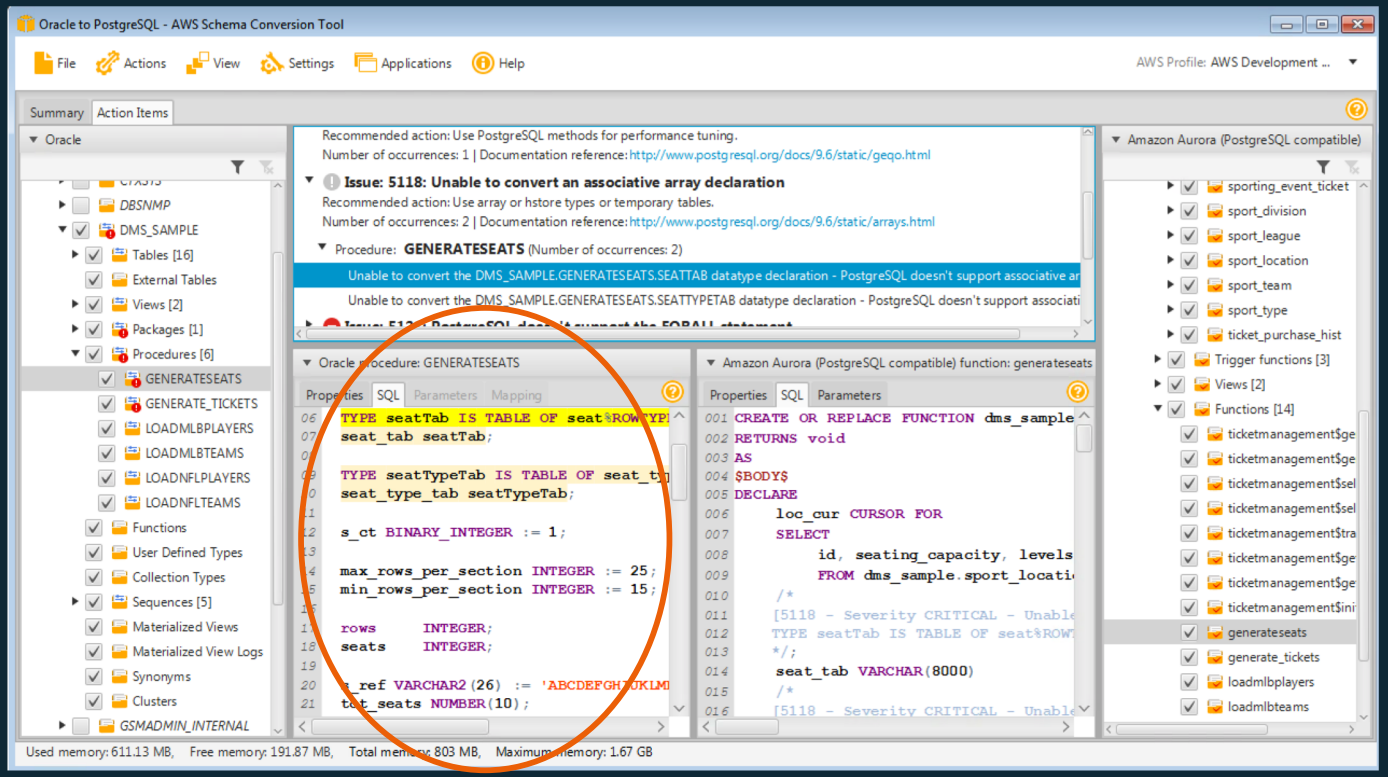

- Action items to help with manual conversion

# Target database schema



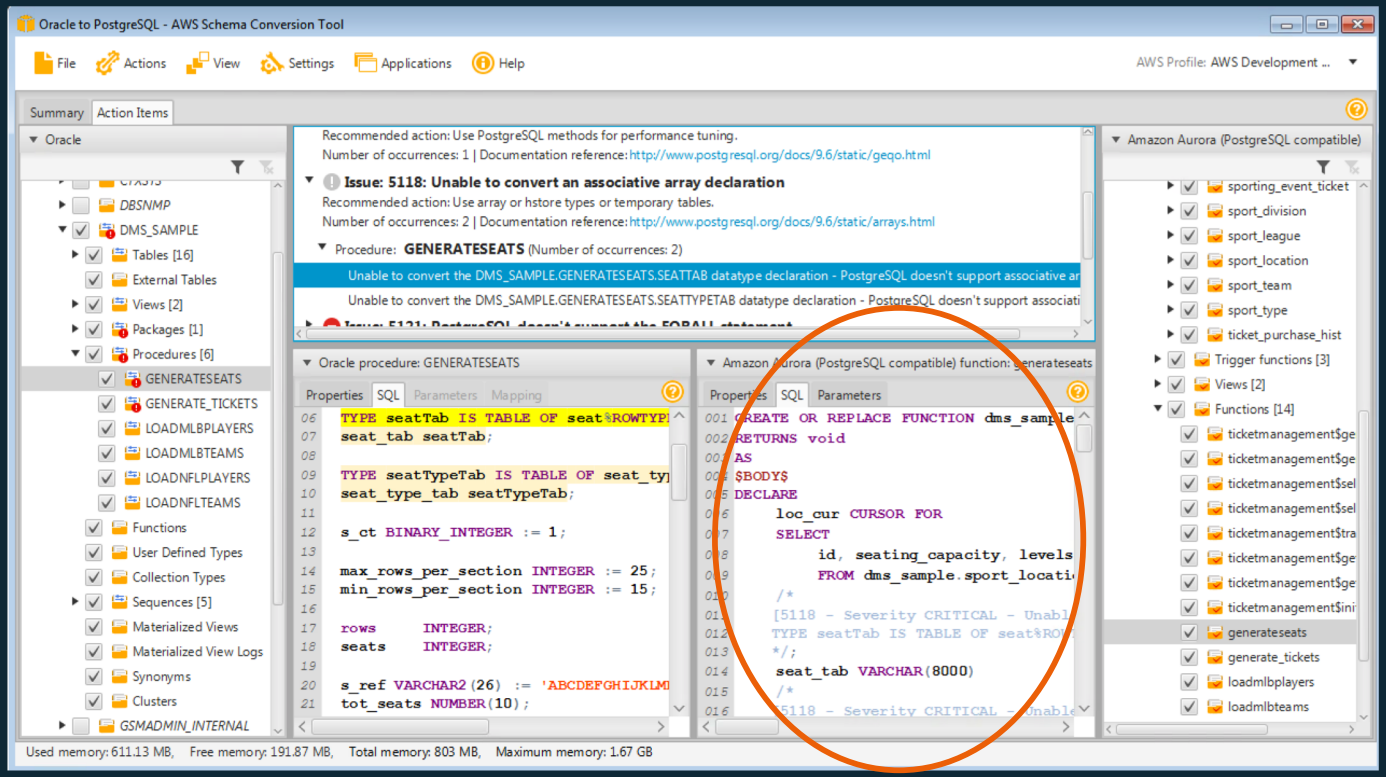

- Tree view of target
- Lazy loaded objects

# Source object code




- SQL code to create chosen source database object
- Properties of chosen source database object

# Target object code

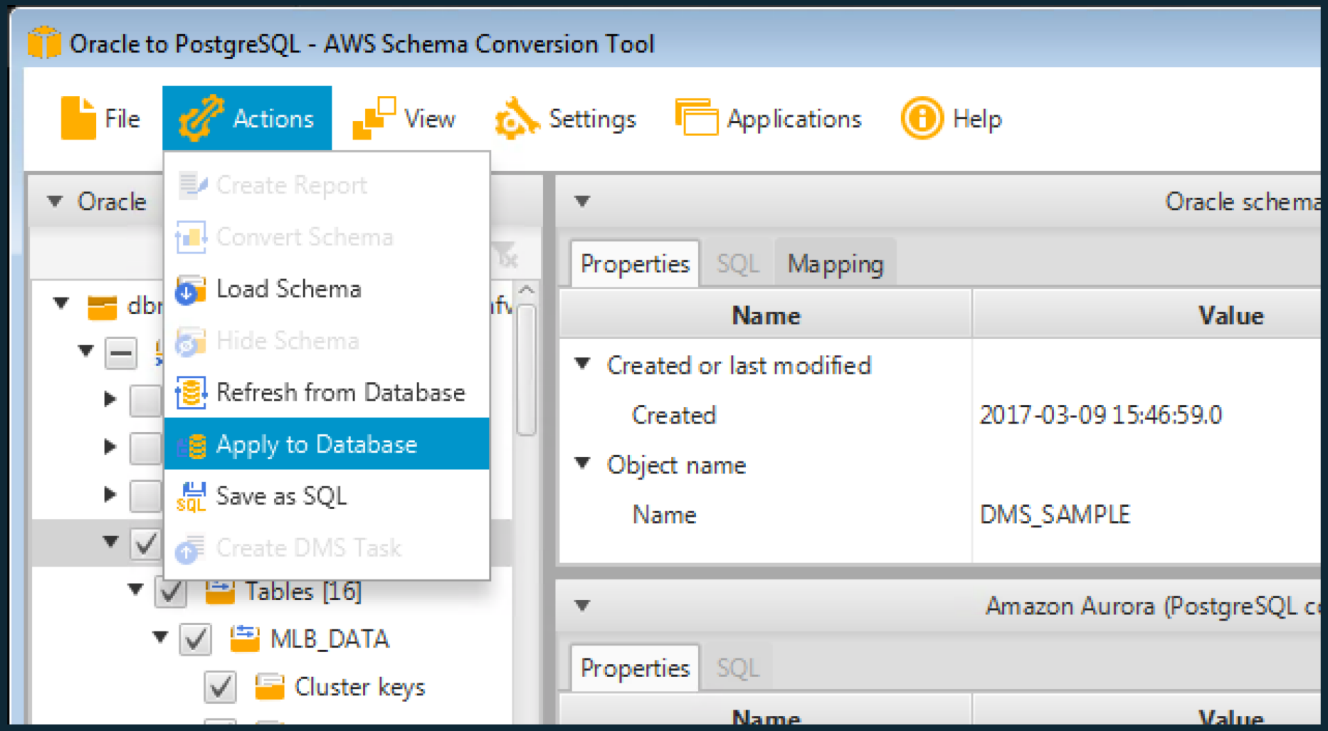



- Editable SQL code to create chosen object in the target database
- Properties of chosen target database object



aws

# Apply to target database




Select Apply to Database from the Action menu

*or*

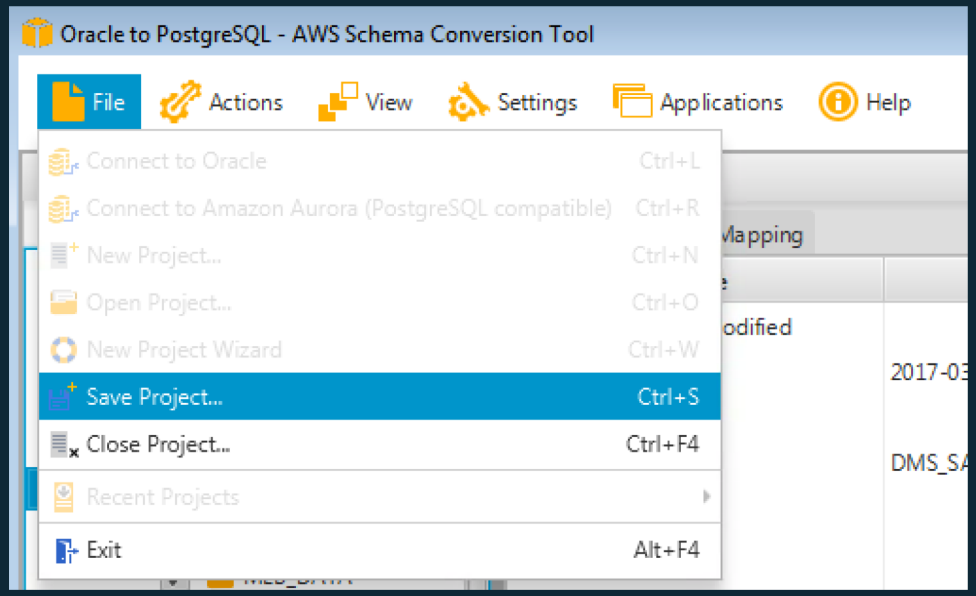
Right click on the target schema to apply



aws

# Menu navigation

# Work with projects



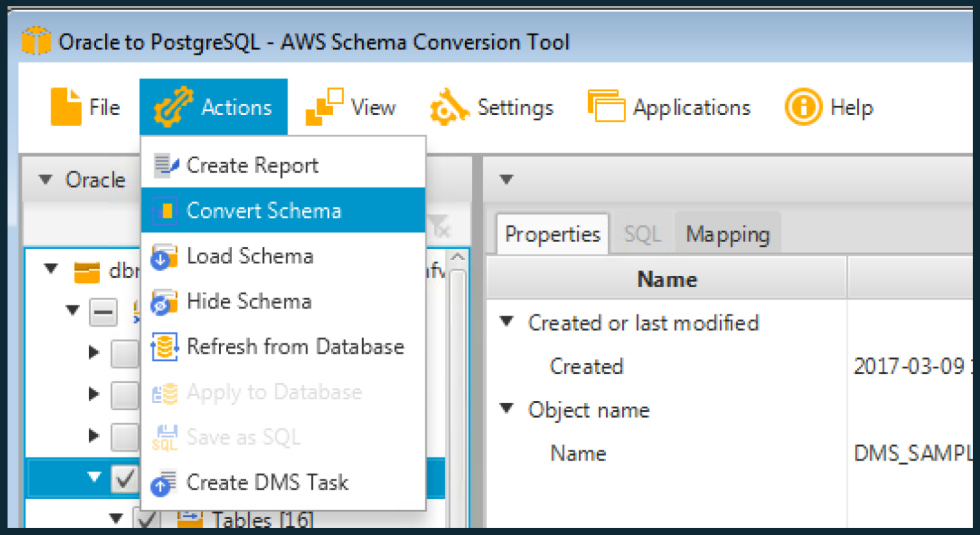

Save your work

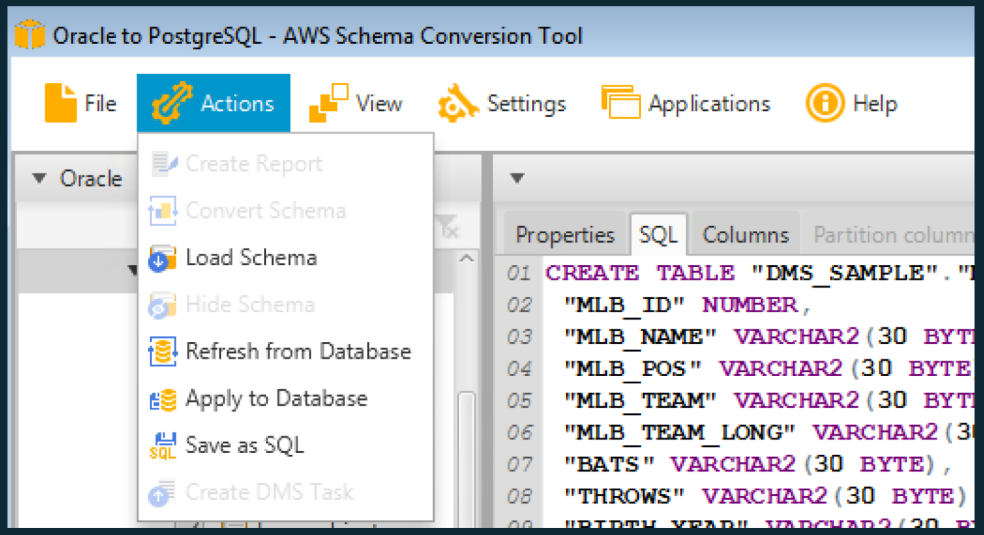
Open an existing project from a list of recent ones

Connect to a source or target

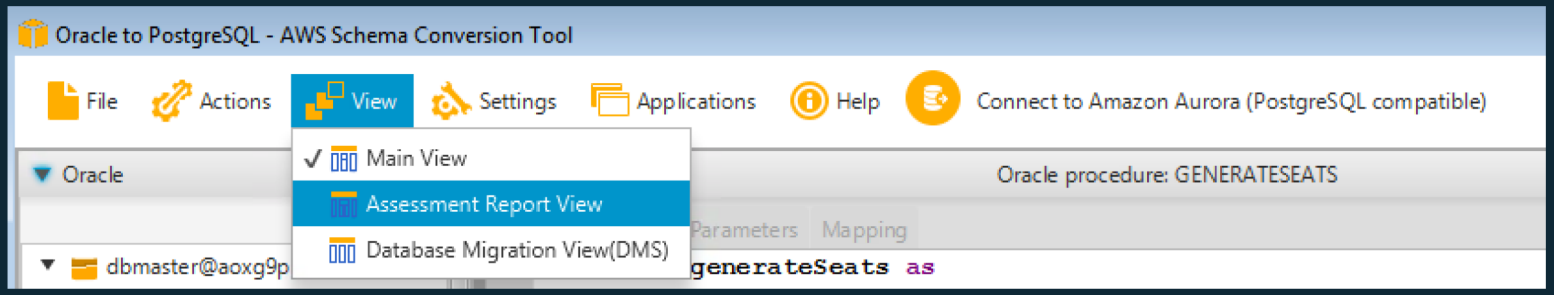
# Context sensitive actions

Menu options change when you work with the source versus the target



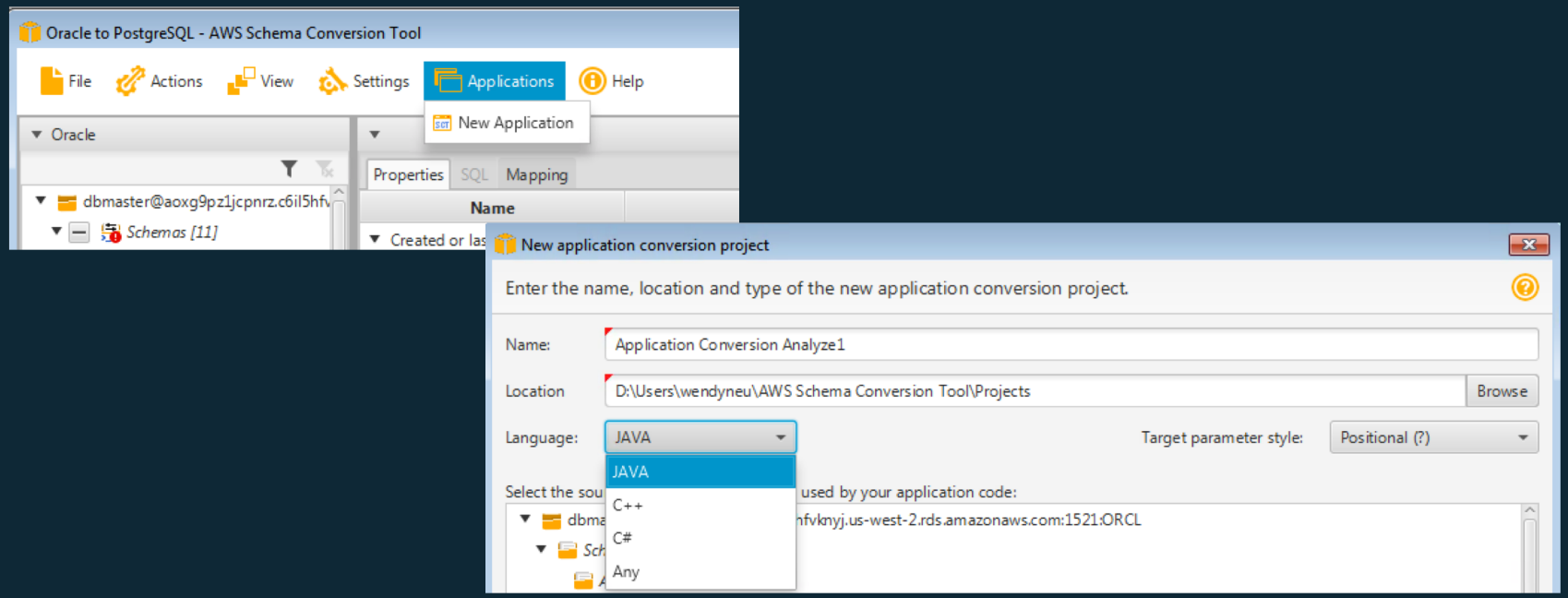



# Switch between views



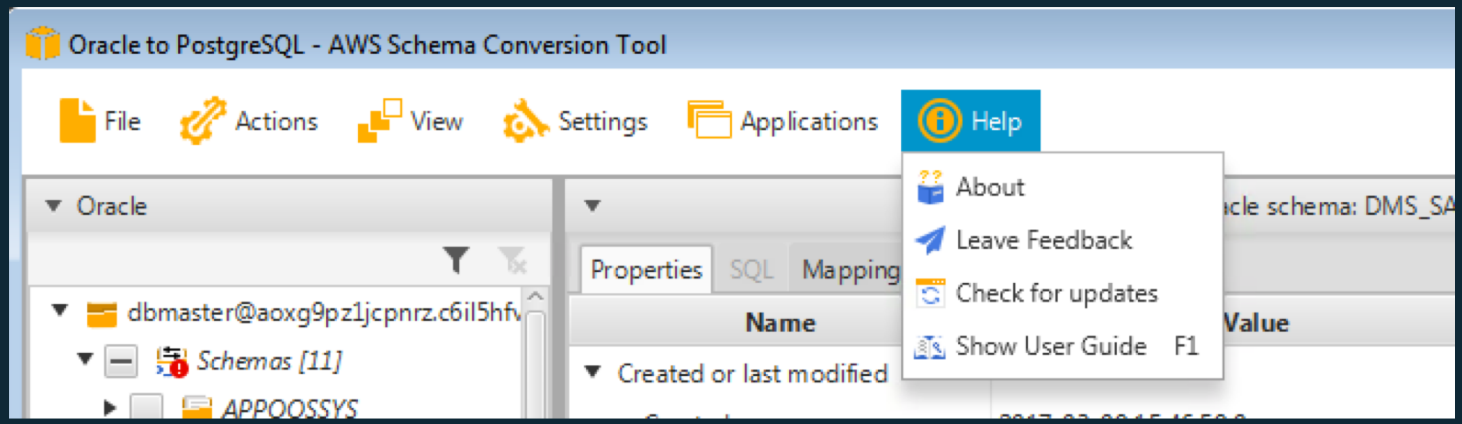

- Main view shows the working window for code conversion
- Assessment report view shows the summary and action item detail
- Database migration view for orchestrating your migration from within SCT

# Convert an application




Use the AWS Schema Conversion Tool to convert your application code

# Get help

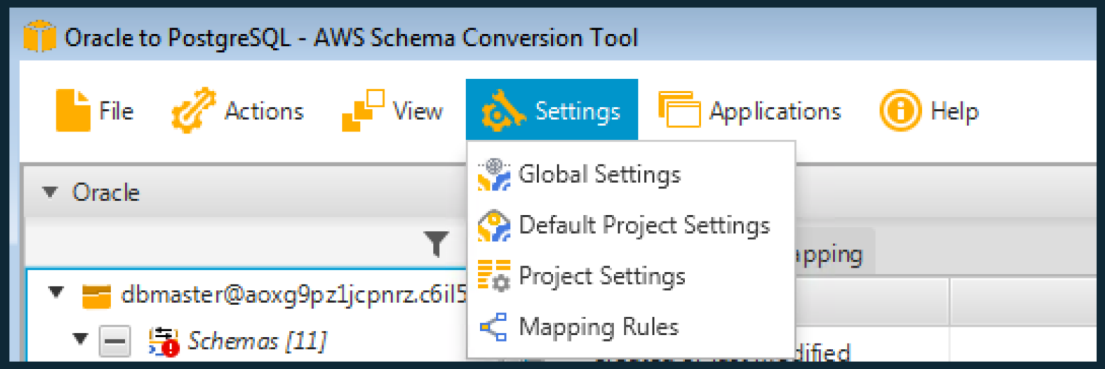



Update AWS SCT

User Guide

# Closer look at Settings

# Global settings



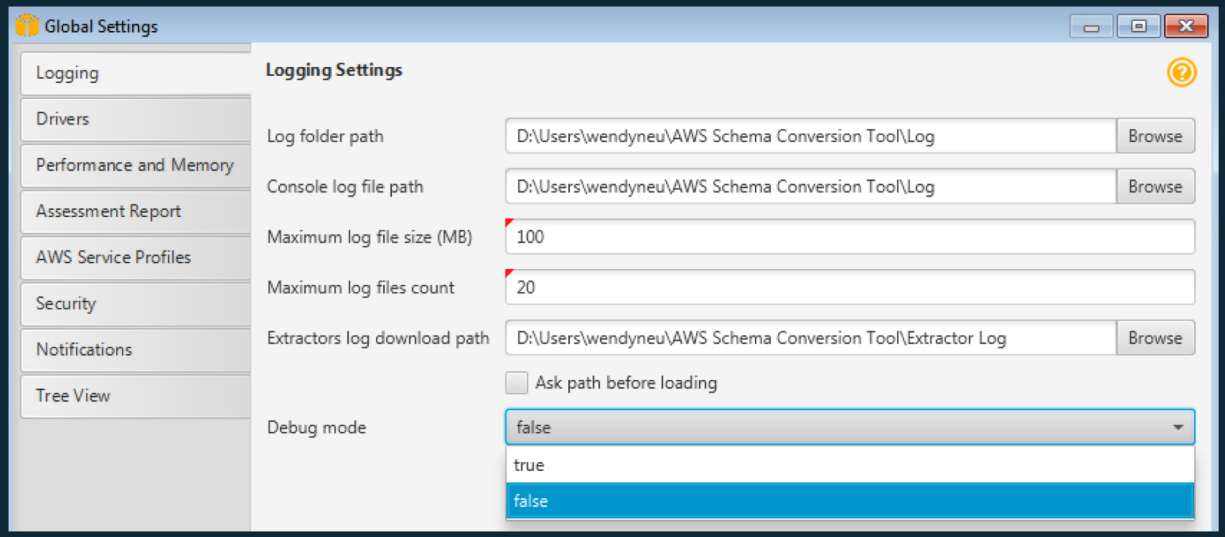

Applies to your installation of SCT

Control

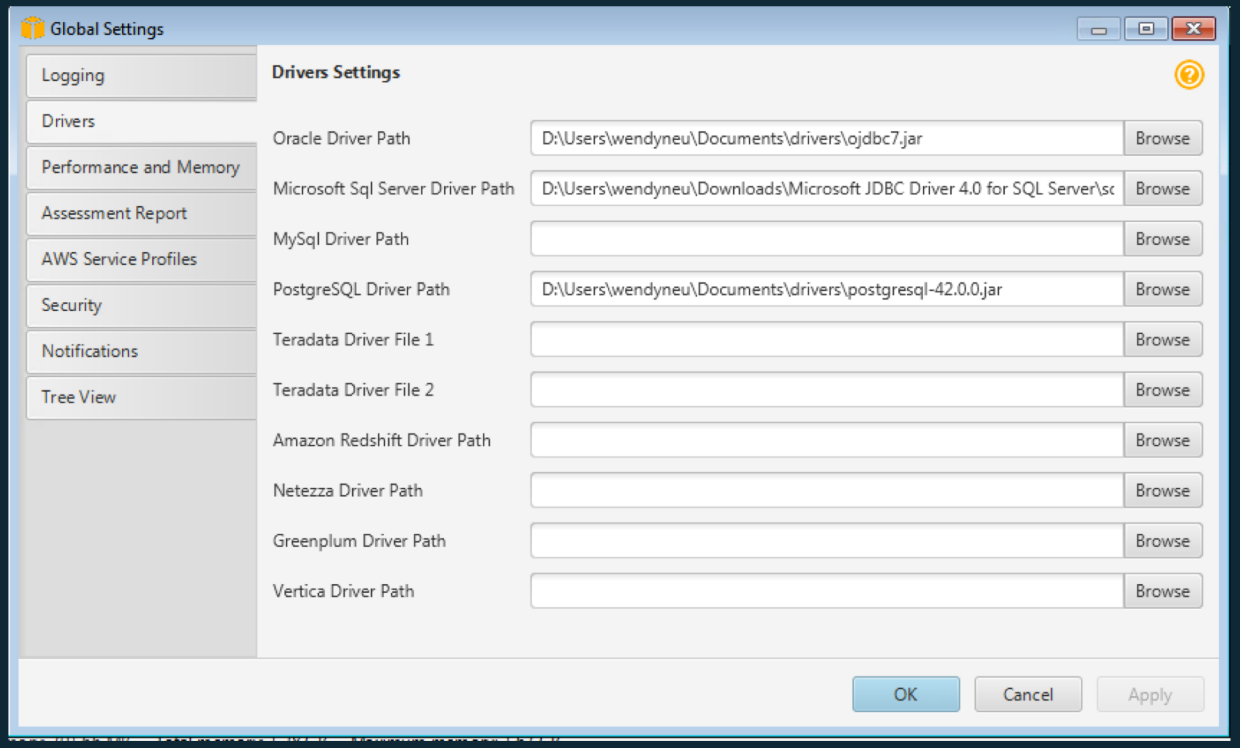
- Logging
- Drivers
- Performance and Memory
- Assessment Report
- AWS Service Profiles
- Security
- Notifications
- Tree View

# Global settings - logging

- Application activity is logged and file size is retained to 100 MB up to 20 files by default
- Increase the information in the log by setting Debug mode to true



# Drivers

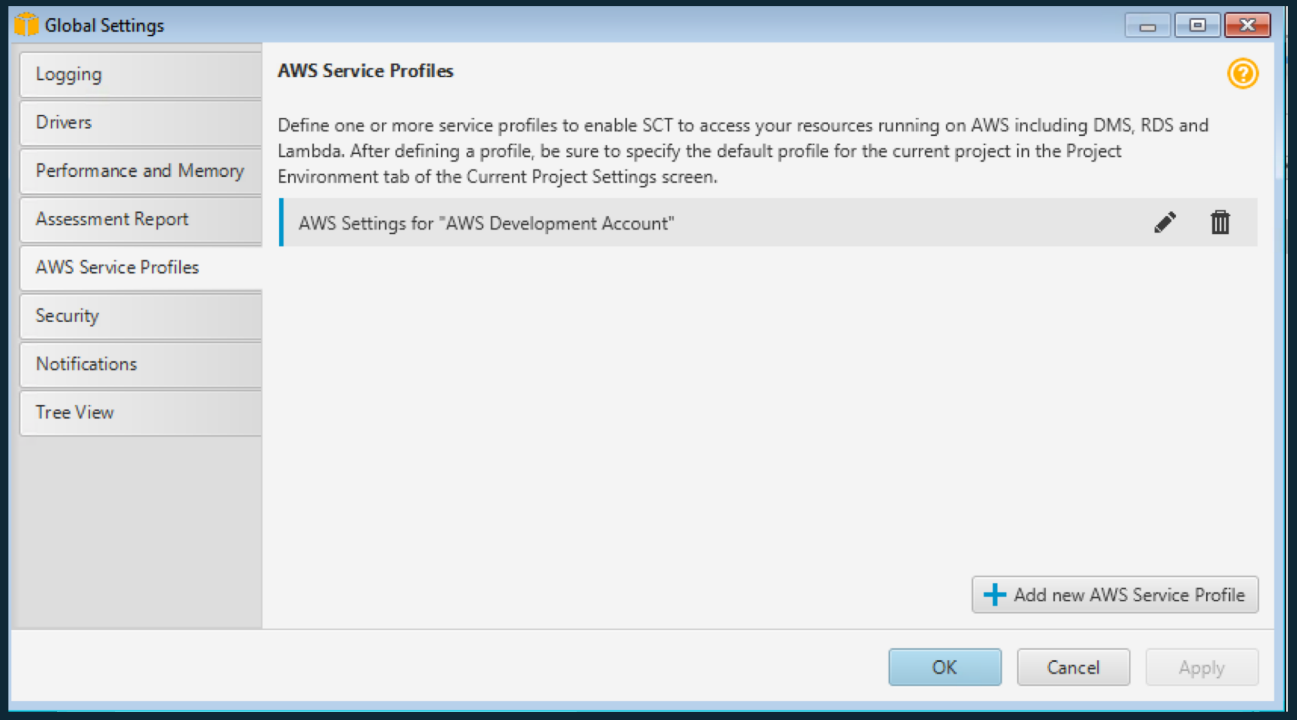



- Add or update drivers for your databases
- Drivers are used to connect to the source and target
- Use the recommended drivers in the user guide



aws

# AWS Service Profiles

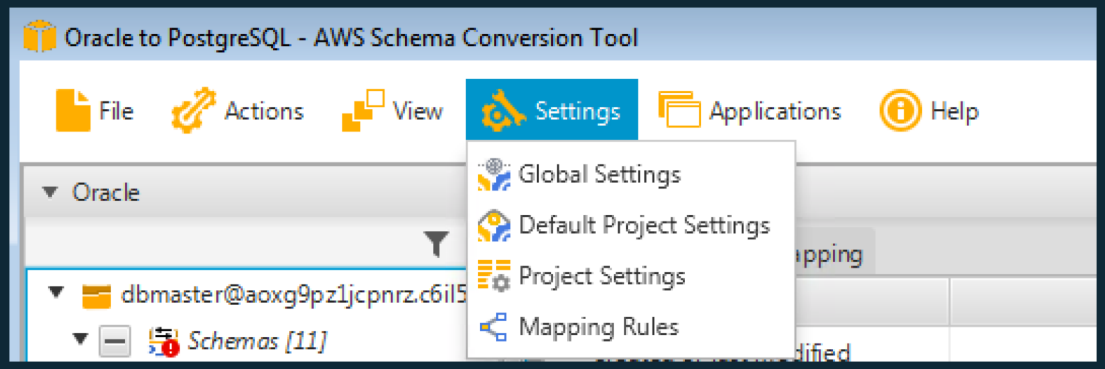



Add your AWS Credentials to initiate an AWS DMS migration from within AWS SCT

Enter your AWS Access Key and AWS Secret Key

Test the connection to check permissions

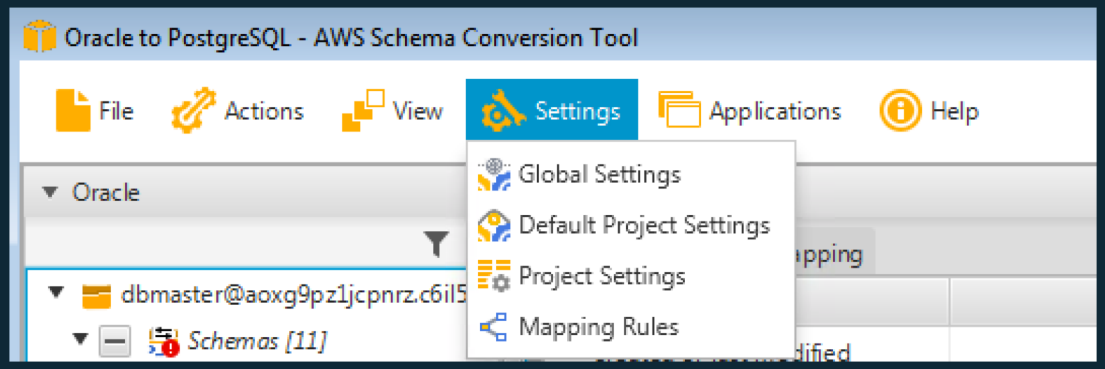
# Control your application and project settings




Default Project Settings

Indicates where your project will be stored by default

# Control your application and project settings



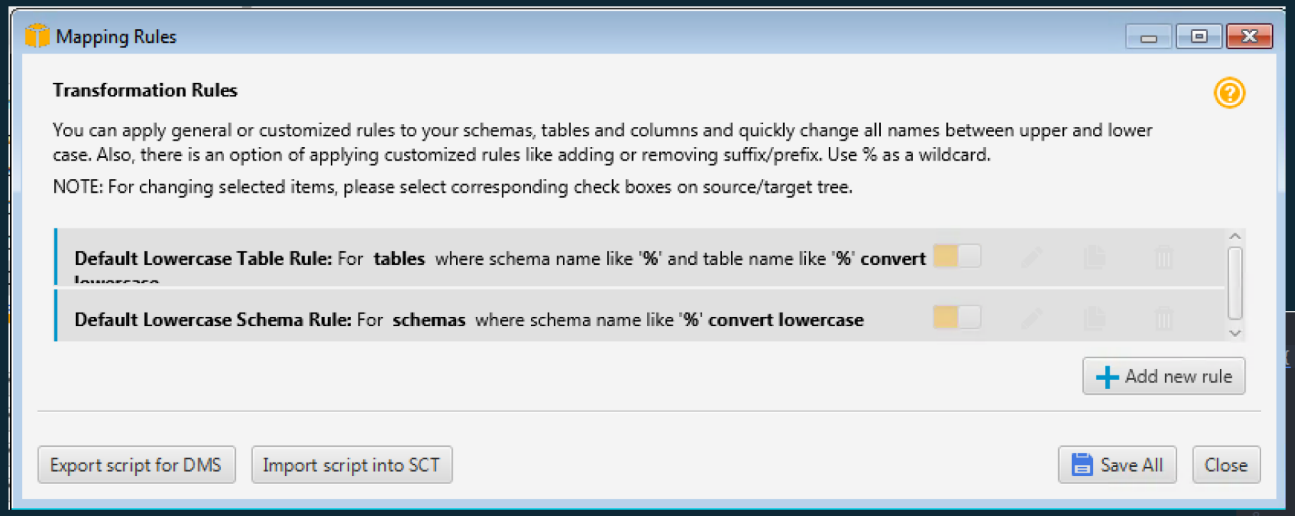

Project Settings

Settings specific to the open project

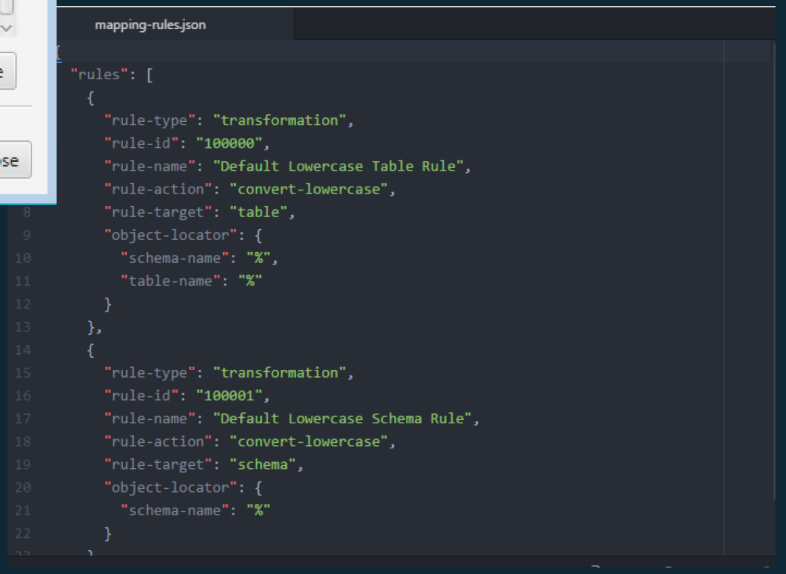
Control

- Conversion settings
- Project environment
- SQL Scripting
- Tree View

# Modify or add mapping rules




Import or export transformation rules as JSON






aws

# AWS Schema Conversion Tool Lab