

How to Generate ERD from Redshift Database

Written Date : April 08, 2016

You can reverse engineer <u>Entity Relationship Diagram (ERD)</u> from an <u>Amazon Redshift</u> database. Doing so provides you with a database design diagram that can be used for analysis of database structure, for communication with teammates and for archiving. In this tutorial, you will form an ERD from a simple online bike shop database.

Reverse engineer ERD from database in Redshift

In order to walk through this tutorial, please setup a new database in Redshift first. In this tutorial, we will interact with a database named *Tutorial01*. You may use any name you like. Once the database is created, execute the following scripts to create database tables in database. In the next section we will form an ERD from these tables.

CREATE TABLE Item (ID INT NOT NULL IDENTITY, name varchar(255), price float4, BrandID int4 NOT NULL, CatagoryID int4 NOT NULL, PRIMARY KEY (ID));

CREATE TABLE ItemVariant (ID INT NOT NULL IDENTITY, detail varchar(255), color varchar(30), "size" varchar(50), qty int4, ItemID int4 NOT NULL, PRIMARY KEY (ID));

CREATE TABLE Brand (ID INT NOT NULL IDENTITY, name varchar(255), PRIMARY KEY (ID)); CREATE TABLE Catagory (ID INT NOT NULL IDENTITY, name varchar(255), PRIMARY KEY (ID)); CREATE TABLE Customer (ID INT NOT NULL IDENTITY, firstName varchar(50), lastName varchar(50), email varchar(50), address varchar(255), phone varchar(50), PRIMARY KEY (ID)); CREATE TABLE "Order" (ID INT NOT NULL IDENTITY, CustomerID int4 NOT NULL, orderDate date, PRIMARY KEY (ID));

CREATE TABLE OrderLine (OrderID int4 NOT NULL, ItemVariantID int4 NOT NULL, qty int4, PRIMARY KEY (OrderID, ItemVariantID));

ALTER TABLE ItemVariant ADD CONSTRAINT FKItemVarian766691 FOREIGN KEY (ItemID) REFERENCES Item (ID);

ALTER TABLE Item ADD CONSTRAINT FKItem119940 FOREIGN KEY (BrandID) REFERENCES Brand (ID);

ALTER TABLE Item ADD CONSTRAINT FKItem265973 FOREIGN KEY (CatagoryID) REFERENCES Catagory (ID);

ALTER TABLE "Order" ADD CONSTRAINT FKOrder556711 FOREIGN KEY (CustomerID) REFERENCES Customer (ID);

ALTER TABLE OrderLine ADD CONSTRAINT FKOrderLine150838 FOREIGN KEY (OrderID) REFERENCES "Order" (ID);

ALTER TABLE OrderLine ADD CONSTRAINT FKOrderLine292294 FOREIGN KEY (ItemVariantID) REFERENCES ItemVariant (ID);

Reverse engineer ERD from database in Redshift

To reverse database:

- 1. Select **Tools > DB > Reverse Database** from the application toolbar.
- 2. In the **Database to Data Model** window, click **Next**.

&	Database to Data Model	×
Visual Pa Select Lar	radigm Database Reverse nguage	*
Language :	Java	~
Result :	Popup entities tree (can drag entities to diagram on demand)	~
Reverse	Table	
Include :	synonyms	
Reverse	Stored Procedure	
Reverse	Trigger	
Group st	tored procedures and triggers in one shape	
Place rever	sed entities to model: <root></root>	
* Table reco	rds are not included in reverse engineering of database	
100101000		lelp
		сiр

3. In the **Database Configuration** screen, select **Redshift** to be the database driver.

\$		Database to Data Model	×
Visual Paradig Database Conf	m Database Reverse		*
Driver :	HSQLDB (In-process)		✓ <>
Driver file :	FrontBase PostgreSQL		^ 🛃
Connection URL :	Ingres OpenEdae		2
Database	InterSystems Cache SOLite		
	H2		
User :	Redshift sa	Password :	
✓ Set as defaul	t	Test C	onnection
		< Back Next > Cancel	Help

4. Provide the JDBC **Driver File**. You may <u>click here to download the Amazon Redshift JDBC</u> <u>driver</u>

\$				Database	to Data Mode	el		×
Visua Datal	l Paradig base Conf	IM Databa iguration	ase Reverse					°*
Driver :		Redshift						v 💙
Driver f	ile :	C: (MyDriv	vers\RedshiftJDB	BC41-1.1.13.1013	3.jar			v 🕹
Connec	tion URL :						P	roduction 🗸 <table-cell></table-cell>
۲	Hostnam	e:						:
	Database	e name :						
0	jdbc:red	shift:// <ho< td=""><td>st_name>:<por< td=""><td>t_number>/<data< td=""><td>abase_name></td><td></td><td></td><td></td></data<></td></por<></td></ho<>	st_name>: <por< td=""><td>t_number>/<data< td=""><td>abase_name></td><td></td><td></td><td></td></data<></td></por<>	t_number>/ <data< td=""><td>abase_name></td><td></td><td></td><td></td></data<>	abase_name>			
User :		sa Password :						
🖌 Set	as defaul	t						Test Connection
					< Back	Next >	Cancel	Help

5. Fill in the hostname, port, username and password of your Redshift database. Again, in this tutorial we are using a database named *tutorial01*.

\$	Database to Data Model
Visual Paradig Database Confi	m Database Reverse guration
Driver :	Redshift 🗸 🗸
Driver file :	C:\MyDrivers\RedshiftJDBC41-1.1.13.1013.jar
Connection URL :	Production 🗸 ?
Hostname	
Database	name: tutorial01
0	
User :	Password :
Set as default	Test Connection
	< Back Next > Cancel Help

- 6. Click **Next**.
- 7. Narrow down the scope of reversal by selecting the *public* schema.

Database to Data Model	×
Visual Paradigm Database Reverse Selecting Schemas	*
Select Schema : 🕜 All Schemas 💿 Selected Schema	
information_schema pg_catalog pg_internal ✔ publid	
Check All Un-check all Cancel H	Help

8. Click Next.

9. Keep the tables selected in the **Selecting Tables** screen. The tables are the result of executing the SQL statements in the previous section.

\$	Database to Data Model	×
Visual Paradigm Database Reverse Selecting Tables		*
No. of table(s) found: 7 Available Tables: Filter Table (wildcard = *)	Selected Tables: public.brand public.catagory public.item public.item public.itemvariant public.order public.orderline > >> <<	
	< Back Next > Cancel	Help

10. Click Next.

11. Click Finish in the Reverse Database Preview screen.

Database to Data Model	×
Reverse Database Preview	*
Image: Solum Solution Image: Solum (Add) Image: Solum (Add) <t< td=""><td>*</td></t<>	*
< Back Finish Cancel	Help

12. A blank ERD will be created, will the **Reversed Entities** window popped out. The entities formed from the Redshift database are listed in the window. Now, select the entities and drag them onto the diagram. You can perform a multiple selection of entity by pressing the **Ctrl** or **Shift** key.



When you release your mouse button you will see the ERD formed from the entities. Tidy up the diagram content. Your diagram should look like the one below.



Related Links

- What is Entity Relationship Diagram (ERD)?
- Generating Redshift Database from ERD
- How to Produce Database Specification
- Provide Default Data for Database Design

Trademark Disclaimer Amazon Redshift is a trademark of Amazon Web Services



Visual Paradigm home page (https://www.visual-paradigm.com/)

Visual Paradigm tutorials (https://www.visual-paradigm.com/tutorials/)