Practical Tips for Migrating your IBM Netezza Data Warehouse to the Cloud



Matt Scaer: Principal DW Specialist SA



Amazon Redshift is a cloud data warehouse that's...









Amazon Redshift

is the most popular cloud data warehouse

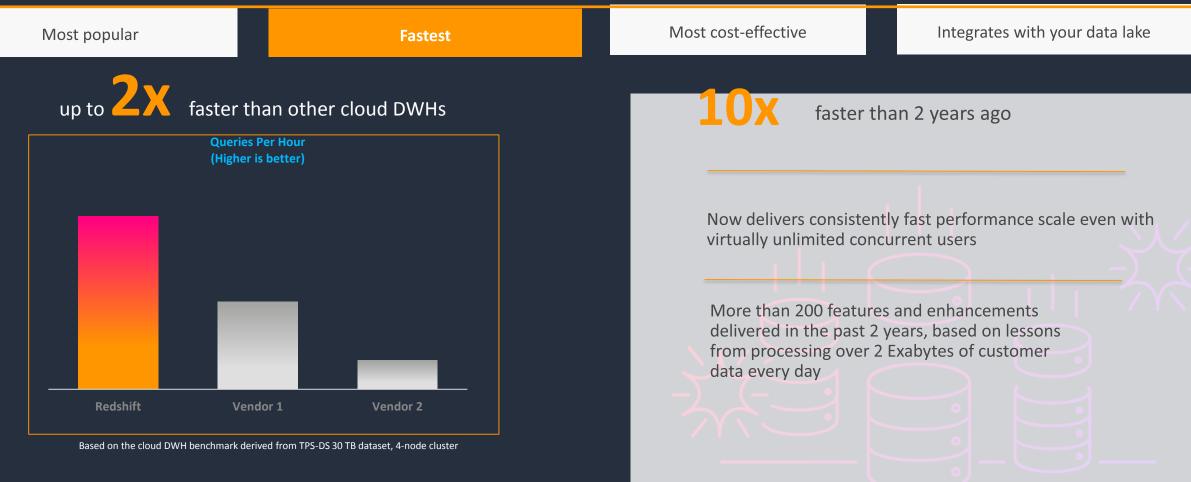




Amazon Redshift is the fastest cloud data warehouse



It reduces time to insights and gives you the **performance** you'll need as your business grows



3, 10 and 30TB:

https://github.com/awslabs/amazon-redshift-utils/tree/master/src/CloudDataWarehouseBenchmark/Cloud-DWB-Derived-from-TPCDSaWS

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Amazon Redshift is the most cost-effective



Moving to Redshift enables you run a 24x7 data warehouse cost-effectively

Most popular	Fastest	Most cost-effective		Integrates with your data lake
You can star	t small for just \$0.25 per hour and scale one-tenth the cost of o		les	s than

Use RI to reduce price up to 75% with Reserved Instances (RIs)





Amazon Redshift integrates with your data lake



Your Amazon Redshift Data Warehouse and Amazon S3 Data Lake enable all your workloads



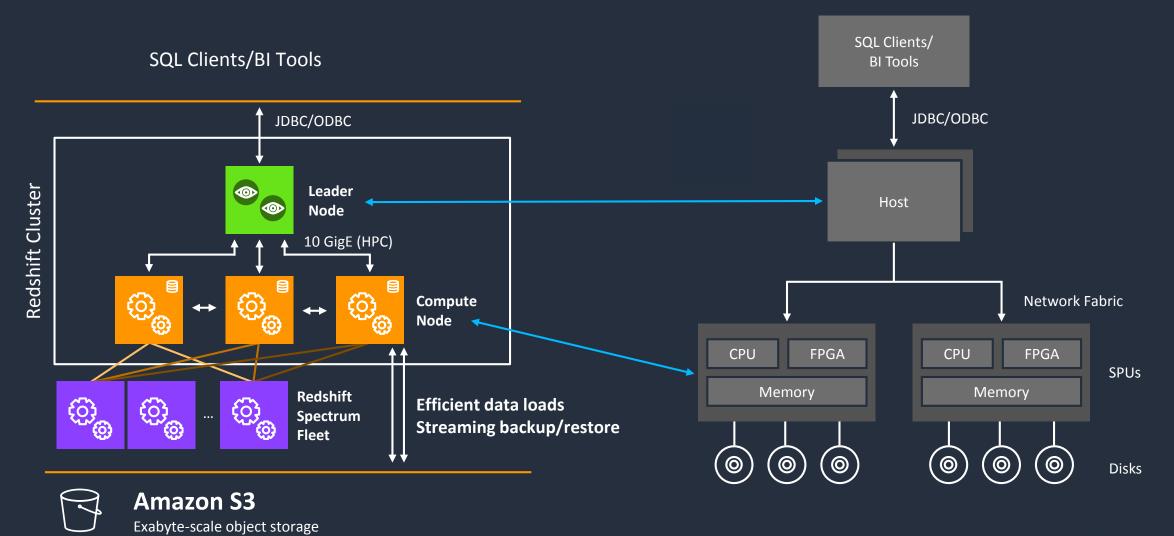


Redshift and Netezza: Similarities Easing Migration to the Cloud

- 1. Shared PostgreSQL lineage
- 2. Many similar concepts should be familiar to Netezza experts.
- 3. Architecture has parallels to a pre-Spectrum Redshift.



Redshift & Netezza Architectures



aws

Amazon Redshift Spectrum

Extend the data warehouse to exabytes of data in Amazon S3 Data Lake

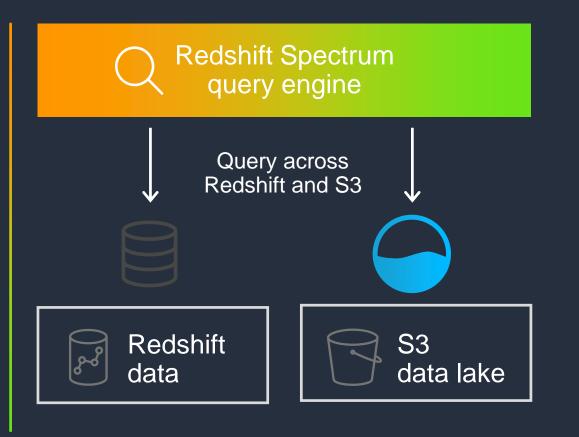
No data loading required

Scale compute and storage separately

Directly query data stored in Amazon S3

Parquet, ORC, Avro, JSON, and CSV data formats

Spectrum Request Accelerator

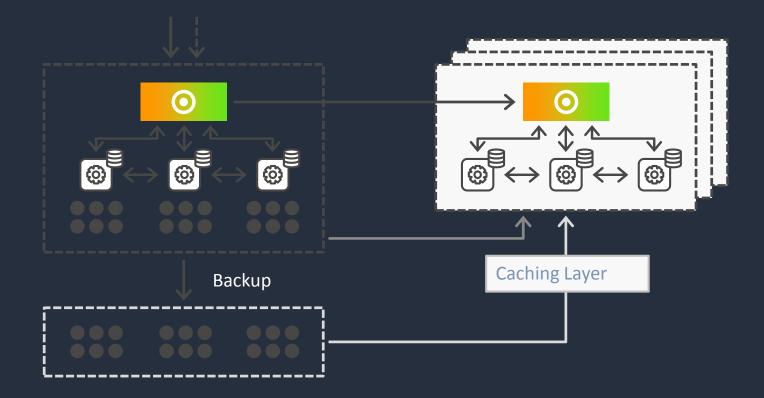






Concurrency Scaling

Redshift automatically adds transient clusters, in seconds, to serve sudden spike in concurrent requests with consistently fast performance. No hydration required.



For every 24 hours that your main of cluster is in use, you accrue a one hour credit for Concurrency Scaling. This means that Concurrency Scaling is **free** for > 97% of customers.

How it works:

- All queries go to the leader node, user only sees less wait for queries.
- When queries in designated WLM queue begin queuing, Redshift automatically routes them to the new clusters, enabling Concurrency Scaling automatically.
- Redshift automatically spins up a new cluster, processes waiting queries and automatically shuts down the Concurrency Scaling cluster.





Concurrency Scaling is a powerful new feature of Amazon Redshift. We were really impressed by the performance of the feature, especially with its ability to instantly add transient capacity with nothing to manage on our end. As Redshift administrators at Yelp, we think that Concurrency Scaling will keep our many users happy, even under peak load. We're excited that Concurrency Scaling provides the flexibility to handle significant variance in our workloads over the course of a day.

> -**Shahid Chohan**, Software Engineer, Yelp



© 2019, Amazon Web Services, Inc. or its affiliate

PEOPLE

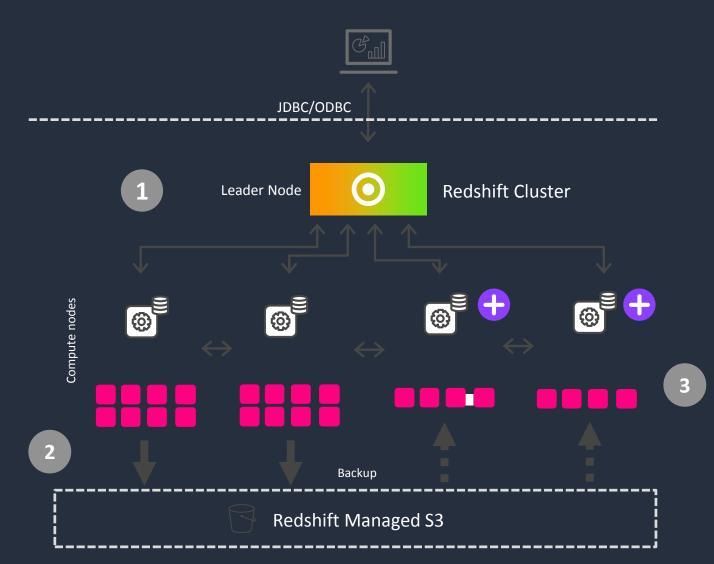
Redshift Elastic Resize



Scale your Redshift clusters up and down in minutes to get the optimal performance.

How it works:

- Redshift updates the snapshot on S3 with the most recent data.
- New nodes are added (for scaling up) or removed (for scaling down) during this period.
- The cluster is fully available for read and write queries. Queries that were being held are queued for execution automatically.





Leverage Redshift Ease of Use



- Auto Analyze automatically collects table statistics to deliver enhanced query performance
- Auto Data Distribution automatically selects table distribution style based on table size
- Auto Vacuum Delete automatically re-sorts and reclaims space from deleted rows, improving performance and space utilization
- Auto Workload Management Dynamic Concurrency with Query Priorities Redshift manages query traffic and executes based on your priority. Scale compute resources based on priority
- Snapshot Scheduler Enhancements more control over automated snapshot schedule and allows setting snapshot expiration date, bulk removal of expired manual snapshots
- Stored Procedures in Redshift makes migration to Redshift easier, you can bring your existing Stored Procedures



Architecture Comparison, Common Ground



Physical level

Netezza	Redshift
Host	Leader Node
Snippet Processing Unit (SPU) on an S-Blade	Compute Node
SPU contains CPU, FPGA, RAM and points to disk.	Compute Node contains CPU, Memory and Disk.
Users don't interact directly with the SPUs	Users don't interact directly with the Compute Nodes

Architecture Comparison, Common Ground



Database level

Netezza	Redshift
Relies on compression and zone maps to minimize I/O	Relies on compression and zone maps to minimize I/O
Compression is at the column level	Compression is at the column level
Data is distributed by a distribution key across the SPUs	Data is distributed by a distribution key across the Compute Nodes
Data can be distributed by the key, or RANDOM	Data can be distributed by the key, EVEN, ALL or Auto
	External data can be queried, but may be processed externally
No indexes	No indexes
No partitioning	No partitioning
No enforcement of PKs	No enforcement of PKs
Employs caching strategies	Employs caching strategies



Other Areas of Common Ground



Database level (Continued)

Netezza	Redshift
Allows a column alias to be referenced in the same SELECT clause supported	Allows a column alias to be referenced in the same SELECT clause supported (called lateral column alias reference in the Redshift documentation).
Use "groom" to physically delete logically delete rows	Use "vacuum delete" to physically delete logically delete rows. Now automated .
"sequence" column data type	"identity" column data
Stored procedure support	Stored procedure support (and SCT conversion from Netezza)
WLM: Prioritized Query Execution (PQE)	WLM: Auto WLM with Query Priorities
WLM: Short Query Bias	WLM: Short Query Acceleration

Redshift Functionality for Netezza Experts to Leverage



Categories	Redshift Managed Service
Operations	Automated patches and upgrades
	Automated backups, and cross region replication
	Hardware and networking on the cluster is AWS's responsibility
Easy to scale	Changing node count and types
	Elastic Resize in matters of minutes
	Redshift Spectrum – storage and compute layer
	Concurrency Scaling to handle additional concurrent spiky workload
Automated Administration	Automation of analyze, vacuum (delete), dist key recommendations
	<u>Redshift Advisor</u>
Data Lake Integration	Ease of operations on open formats within the data lake
Pace of Innovation	Rate of change and roadmap



AWS Ecosystem for Netezza Experts to Leverage



Categories	Redshift
Schema Conversion	Schema Conversion Tool for schema/stored proc conversion
Data Movement	Schema Conversion Tool Data Extractors
Near Real Time / Streaming	Kinesis Firehose and/or Redshift Spectrum
	Redshift as a data source for machine learning
	Redshift as a destination for DynamoDB
For Data Lake Integration + column level security	Lake Formation
Data Integration	AWS Glue + AWS EMR
Data Visualization	QuickSight
Monitoring and alerting	Cloud Watch
Launching Resources	Cloud Formation
Security	IAM



ETL/BI Tool Support



- 1. Note: Tools that support a connection via JDBC/ODBC will connect to Redshift.
- 2. Strong relationships with leading ETL/BI tool vendors.
- 3. This relationship has helped the vendor to generate efficient SQL against Redshift
- 4. Examples:
 - 1. Tableau's whitepaper "Optimizing your Amazon Redshift and Tableau Software Deployment for Better Performance"
 - 2. MicroStrategy's whitepaper "Amazon Redshift Best Practices for Performance"
- 5. Redshift Quick Starts: https://aws.amazon.com/quickstart/architecture/amazon-redshift/

Migration using Amazon Redshift Partners Products alooma DATOMETRY denodo snapLogic[®] MATILLION **Amazon Redshift FlyData** Segment ATTUNITY 0 CloudBeam xplenty talen ntegration at any sca **etl**eap Informatica https://aws.amazon.com/redshift/partners/



مح

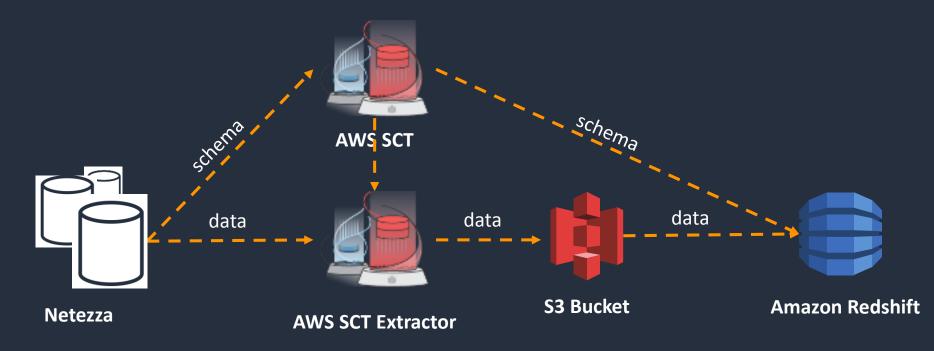
Migration using AWS Schema Conversion Tool

Run AWS Schema Conversion Tool

- To evaluate complexity of migration
- To identify supported/unsupported objects
- To migrate your schema

Use AWS SCT Data Extraction Agents

- To migrate your data
- Parallelize, scale out and scale up for performance
- Encrypt command and data streams for security





()مگ

aws

AWS profile: soo-profile -

😣 🗖 🗊 Netezza - Redshift 🛛 AWS Schema Conversion Tool

© 2019,

📙 File 🛷 Actions 📲 View 💩 Settings 🔚 Applications 🔞 Help

Used memory: 352.46 MB, Free memory: 718.54 MB, Total memory: 1.05 GB, Maximum memory: 3.46 GB

• W Dudbases (7) > PE NTEGER, • W DDA_SCAPE > PAYLOAD CARACTEE VARYING (400), • W DDA_SCAPE > PAYLOAD CARACTEE VARYING (400), • W DDA_SCAPE > DISTREMUTE ON (PK); • W DDASCON Ø DISTSTYPE REV •	File 👸 Actions 📑 View 🐼 S	ettings • Applications 😈 help	Aws prome. soo-prome +
■ and multiplications Soft Submitted Soft Soft Soft Soft Soft Soft Soft Soft	▼ Netezza	▼ Netezza table: TEST4	 Amazon Redshift
• W Dudubases (7) > P XTISER, • W DDA_SCHUTS > PYLOBA CREAKTER VARYING (400), • W DDA_SCHUTS > PYLOBA CREAKTER VARYING (400), • W DDA_SCHUTS > D STRIBUTE ON (PK); • W DDA_ST_TNV, MATSCH > D STRIBUTE ON (PK); • W DDA_ST_TNV, MATSCH > D STRIBUTE ON (PK); • W DDA_ST_TNV, MATSCH > D STRIBUTE ON (STRIBUTE ON (S	T Te	Properties SQL Columns Distribution key columns Organize columns Related converted objects Statistics	T 16
W → RALADA SCRUTS J → MALADA CHARACTER VANTING (400), W → ECARE J ST STRUTSTAPP W → ECARE J G STRUTS (G) W → Tables (B) J J W → AFS, TRNU, MKTSCN J AAST, TRNU, MKTSCN W → AFS, TRNU, MKTSCN J AAST, TRNU, MKTSCN W → AFS, TRNU, MKTSCN J AAST, TRNU, MKTSCN W → AFS, TRNU, MKTSCN J Amazon Redshift table: testA W → TEST3 V Amazon Redshift table: testA W → TEST3 G Control Gints (G) G G G G G G G G G G G G G G G G G G G	▼ <mark>≡</mark> admin@172.16.118.128:5480	1 CREATE TABLE "ADMIN".TEST4(▼ 🖌 🥃 awsdbe@test-for-soo.csanf2qfmbmi.us-east-1.redshif
Image: Control Image: Control Image: Control Image: Co	🔻 🗹 💾 Databases [7]		🔻 🗹 😼 Schemas [22]
Image: Market Set Set Set Set Set Set Set Set Set S	► 🖌 😑 DBA_SCRIPTS		V 🖕 _control
 ↓ UMM NUTEZZA_CUSTOMER_5 ↓ ↓ WWSION ↓	🕨 🔽 😑 ECARE		Image: Arrow and Arrow
 W WASSON W TEST W TEST W TASS RAW, MATSCN W AFS, TRW, MATSCN W TEST1 W TEST2 W TEST3 W TEST3 W TEST5 W TEST5<td>► 🔽 😑 EDW</td><td>6 DISTRIBUTE ON (PK);</td><td>▶ 🖌 😑 aws_oracle_ext</td>	► 🔽 😑 EDW	6 DISTRIBUTE ON (PK);	▶ 🖌 😑 aws_oracle_ext
• ♥ ● TEST • ♥ ● aws_vertice_ect • ♥ ● ADMN • ♥ ● aws_vertice_ect • ♥ ● ADMN • ♥ ● aws_vertice_ect • ♥ ● Tables (9) • ♥ ● aws_vertice_ect • ♥ ● AFS_TRVV_MKTSCN • ♥ ● aws_vertice_ect • ♥ ● TEST3 • ♥ ● aws_vertice_ect • ♥ ● TEST4 • ● Customs Distribution key columns Sort key columns Key management • ♥ ● aws_vertice_ect • ♥ ● Rectines Tables • ● pws_taster • ♥ ● aws_vertice_ect • ♥ ● Procedures Tables • ● pws_taster • ♥ ● aws_taster <td>► 🔽 😑 IBM_NETEZZA_CUSTOMER_S</td> <td></td> <td>► 🖌 😑 aws_sqlserver_ext</td>	► 🔽 😑 IBM_NETEZZA_CUSTOMER_S		► 🖌 😑 aws_sqlserver_ext
<pre> Schemas [1]</pre>	🕨 🔽 🧮 INVISION		▶ 🖌 😑 aws_teradata_ext
▼ ▲ ADMIN ▼ ↓ Tables (9) ▶ ↓ AFS_TRNU_MINTSCN ▶ ↓ AFS_TRNU_MINTSCN ▶ ↓ AFS_TRNU_MINTSCN ▶ ↓ AFS_TRNU_MINTSCN > ↓ AFSTAN > ↓ AFSTAN > ↓ AFSTAN > ↓ ↓ IFGER, > ↓ ↓ IFGER > ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	🔻 🖌 💾 TEST		► 🖌 😑 aws_vertica_ext
 ✓ Tables [9] ✓ AFS_TRVU_MCTSCN ✓ AFS_TRVU_MCTSCN ✓ AFS_TRVU_MCTSCN ✓ AFS_TRVU_MCTSCN ✓ TEST1 ✓ TEST2 ✓ TEST3 ✓ TEST4 Ø CREATE TABLE IF NOT EXISTS test_admin.test4(Ø 2 pk INTEGER, Ø 2 pvload (ARACTER VARYING(400) ENCODE ZSTD, Ø T tothes [1] Ø T TISTANP WITHOUT TIME ZONE ENCODE ZSTD, Ø T tothes [1] Ø T TOTSTANP WITHOUT TIME ZONE ENCODE ZSTD, Ø T tothes [1] Ø Procedures [1] Ø pk Ø	🔻 🖌 🚢 Schemas [1]		🕨 🔽 🤤 control
 AFS_TRVU_MKTSCN Cursor position: 0 Modified: true Amazon Redshift table: test4 Amazon R	🔻 🖌 🚢 ADMIN		▶ 🖌 😑 crashdumps
 AFS_TRVU_MKTSCN AFS_TRVU_MKTSCN AFS_TRVU_MKTSCN Cursor position: 0 Modified: true Properties 50. Columns Distribution key columns Sort key columns Key management 01 CREATE TABLE IF NOT EXISTS test_admin.test4(01 CREATE TABLE IF NOT EXISTS test_admin.test4(03 payload CHARACTER VARYING(400) ENCODE ZSTD, 04 Materialized Views 05) 07 Procedures [1] 07 Evendures [1] 07 Evendures [1] 07 Sequences [1] 08 Sequences [1] 09 pk 10): 	🔻 🗹 🚢 Tables [9]		▶ 🖌 🔚 dbc
 ArS_TRVU_MKTSCN ArS_TRVU_MKTSCN ArS_TRVU_MKTSCN Cursor position: 0 Modified: true Argon Redshift table: test4 Argon	► 🗹 🔚 AFS_TRVN_MKTSCN		▶ 🗸 😑 dbmaster
> ✓ AFS_TRVN_MKTSCN > ✓ a TEST1 > ✓ a TEST2 > ✓ a TEST3 > ✓ a TEST4 > ✓ a TEST3 > Ø apsload CHARACTER VARYING(400) ENCODE ZSTD, 4 to S INTFLE KAY Ø apsload CHARACTER VARYING(400) ENCODE ZSTD, 4 to S INTSKEY Ø apsload CHARACTER VARYING(400) ENCODE ZSTD, 4 to S INTKEY Ø apsload CHARACTER VARYING(400) ENCODE ZSTD, Ø apsload CHARACTER VARYING (400) ENCODE ZSTD, Ø apsload CHARACTER VARYING (400	► 🗹 😑 AFS_TRVN_MKTSCN		▶ 🗸 📮 dbo
▶ ♥ ■ TEST1 Cursor position: 0 Modified: true > ♥ ■ public > ♥ ■ TEST2 ▼ Amazon Redshift table: test4 > ♥ ■ public > ♥ ■ TEST3 Properties SQL Columns Distribution key columns Sort key columns Key management ● ♥ ■ Seg_dm > ♥ ■ TEST4 01 CREATE TABLE IF NOT EXISTS test_admin.test4(● ♥ ■ TEST5 02 pk INTEGER, ● ♥ ■ Views 03 payload CHARACTER VARYING(400) ENCODE ZSTD, ● ♥ ■ Views ● ♥ ■ TEST5 ● ♥ ■ Notechard 05) ● DISTSTYLE KEY ● DISTSTYLE KEY ● ♥ ■ Procedures [1] ● ♥ ■ K ● ♥ ■ Recense [1] ● ♥ ■ Sequences [1] ● ♥ ■ Applications ● 0 pk ■ External_2 ● ♥ ■ test_schema_1 ● ♥ ■ Applications ● ■ Note IN INTER ● ■ Note IN INTER ● ■ Note IN INTER	► 🗹 😑 AFS_TRVN_MKTSCN		▶ 🗸 😑 dms_sample
Image: Solid Columns Amazon Redshift table: test4 Image: Solid Columns From Extrast Image: Solid Columns Solid Key columns	► 🗹 😑 AFS_TRVN_MKTSCN		▶ 🗸 🧧 online_sales
▶ ♥ ■ TEST3 Properties SQL Columns Distribution key columns Key management ♥ I = tsi.dmin ▶ ♥ ■ TEST4 01 CREATE TABLE IF NOT EXISTS test_admin.test4(♥ I = tsi.dmin ♥ I = tsi.dmin ♥ ■ TEST3 02 CREATE TABLE IF NOT EXISTS test_admin.test4(02 № I = tsi.dmin ♥ I = tsi.dmin Ø # External tables 03 payload CHARACTER VARYING(400) ENCODE ZSTD, IIIII = tsi.dmin ♥ I = tsi.dmin Ø # NUTEGER, 03 payload CHARACTER VARYING(400) ENCODE ZSTD, IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	🕨 🗹 🧧 TEST1	Cursor position: 0 Modified: true	▶ 🔽 😑 public
Image: Projecties of Coordinate Softward Coordinate So	► 🗹 🧁 TEST2	Amazon Redshift table: test4	▶ 🗸 😑 sbg_dm
▶ 1 TEST4 01 CREATE TABLE IF NOT EXISTS test_admin.test4(▶ 2 TEST5 01 payload (HARACTER VARYING(400) ENCODE ZSTD, > 1 Views 03 payload (HARACTER VARYING(400) ENCODE ZSTD, > 1 Views 05 j > 1 Functions 06 j > 1 Functions 07 DISTSTYLE KEY 08 (09 pk > 2 CUSTOMER 09 pk > 2 Sequences [1] 09 pk > 2 mappications 00 pk > 2 mappications 0 pk	► 🗹 🧁 TEST3	Properties SQL Columns Distribution key columns Sort key columns Key management	▼ 🗸 🥃 test_admin
Image: Procedures [1] 02 pk INTEGER, Image: Procedures [1] 09 pk Image: Procedures [1] 0 pc Image: Procedures [1] pc pc p	🕨 🗹 🚆 TEST4		
W External tables 04 ts TIMESTAMP WITHOUT TIME ZONE ENCODE ZSTD W </td <td>► 🗹 🧁 TEST5</td> <td>-</td> <td>► 🗸 😼 test4</td>	► 🗹 🧁 TEST5	-	► 🗸 😼 test4
✓ ✓ Views 05) ✓ Materialized Views 06 DISTSTYLE KEY Ø DISTSKEY 06 DISTSKEY Ø Functions 07 DISTKEY Ø Procedures [1] 09 pk ✓ CUSTOMER 10); ✓ Sequences [1] ✓ Test_schema_0 ✓ Synonyms ✓ Test_schema_2 ✓ Applications ✓ Test_schema_2 ✓ Applications ✓ Test_schema_2 ✓ Generic ✓ Test_user	🗸 🥃 External tables		Views
Image: Custom 06 DISTSTYLE KEY Image: Custom Ima	🗸 🥃 Views		► ✓ 📮 Functions [1]
W = Functions 08 (V = Procedures [1] V = CUSTOMER I = 0); Sequences [1] V = Sequences [1] V = nzprod V = Applications V = Generic	🗸 🥃 Materialized Views		V Procedures
Image: Procedures [1] 09 pk Image: Discrete Procedures [1] Image: Discrete Procedures [1] <td>🗸 😑 Functions</td> <td></td> <td>► 🗸 🧧 test_dbo</td>	🗸 😑 Functions		► 🗸 🧧 test_dbo
Image: CUSTOMER Image: Dispersive Customer Image: Customer <td>🔻 🖌 😑 Procedures [1]</td> <td></td> <td>► 🗸 🧧 test schema</td>	🔻 🖌 😑 Procedures [1]		► 🗸 🧧 test schema
Image: Sequences [1] Image: Synonyms Image: Synonym	CUSTOMER		► V = test schema 0
✓ Synonyms ▶ ✓ anzprod ✓ Applications ✓ Generic	🕨 🔽 🥃 Sequences [1]		
▶ ✓ ■ nzprod ▶ ✓ ■ test_user ✓ ■ Applications ▶ ✓ ■ test_user ✓ ■ Generic ■ test_user			
 ✓ Generic 	🕨 🔽 🧧 nzprod		
	🧹 🥃 Generic		
Cursor position: 0 Modified: true	_		
< Cursor position: 0 Modified: true			
	Y <mark>< ></mark>	Cursor position: 0 Modified: true	<>

گی(

📙 File 🚀 Actions 📲 View 🔥 Settings 🦳 Applications 🔞 Help AWS profile: soo-profile - Amazon Redshift Netezza Agents Tasks Virtual partitions Snowball T T Table name Partition type admin@172.16.118.128:5480 V awsdbe@test-for-soo.csanf2qfmbi TEST.ADMIN.TEST4 Range 🔻 🗸 💾 Databases [7] ▼ 🗸 😼 Schemas [22] ► 🗸 😑 DBA SCRIPTS 🕨 🗸 🧧 control 🕨 🗸 😑 ECARE V = aws_netezza_ext 🕨 🖌 🧧 EDW ▶ 🗸 😑 aws oracle ext 🕨 🔽 🔚 IBM NETEZZA CUSTOMER SE V = aws_sqlserver_ext INVISION V = aws_teradata_ext 🔻 🗸 💾 TEST 🕨 🗸 🧧 aws vertica ext 🔻 🗸 💾 Schemas [1] 🕨 🗸 🧧 control Nodify 🕅 Delete 🔻 🗸 🗦 ADMIN 🕨 🗸 😑 crashdumps Partition type: Range 🔻 🗸 💾 Tables [9] 🕨 🗸 🧧 dbc ► 🗸 🔚 AFS_TRVN_MKTSCN_A Column name: PK 🕨 🗸 🧧 dbmaster Values: [2000000, 4000000, 6000000, 8000000, 10000000, 12000000, 14000000, 16000000, 18000000, 20000000, 2200000 AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dbo Partitions: PK < 2000000 AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dms sample PK >= 2000000 AND PK <= 4000000 🕨 🗸 🔚 AFS TRVN MKTSCN O ▶ 🗸 😑 online sales PK > 4000000 AND PK <= 6000000 🕨 🗸 😑 TEST1 🕨 🗸 🧧 public PK > 6000000 AND PK <= 8000000 ▶ 🗸 😑 TEST2 PK > 8000000 AND PK <= 10000000 🕨 🗸 🧧 sbg_dm 🕨 🗸 🧧 TEST3 PK > 10000000 AND PK <= 12000000 🔻 🗸 😼 test admin PK > 12000000 AND PK <= 14000000 🕨 🗸 📛 TEST4 🔻 🗸 😼 Tables [1] PK > 14000000 AND PK <= 16000000 TEST5 🕨 🗸 😼 test4 PK > 16000000 AND PK <= 18000000 🗸 😑 External tables 🗸 😑 Views PK > 18000000 AND PK <= 20000000 🗸 😑 Views PK > 20000000 AND PK <= 22000000 ▶ 🗸 😑 Functions [1] PK > 22000000 AND PK <= 24000000 \checkmark Materialized Views Procedures PK > 24000000 AND PK <= 26000000 V 🔤 Functions 🕨 🗸 🧧 test dbo PK > 26000000 AND PK <= 28000000 V Procedures [1] 🕨 🗸 🔚 test_schema PK > 28000000 AND PK <= 30000000 CUSTOMER PK > 30000000 AND PK <= 32000000 ▶ 🗸 😑 test schema 0 PK > 32000000 AND PK <= 34000000 Sequences [1] 🕨 🗸 🧧 test schema 1 PK > 34000000 V 🗧 Synonyms ▶ 🗸 😑 test_schema_2 PK IS NULL 🕨 🗸 🧧 nzprod 🕨 🗸 🧧 test user Applications 🗸 🧧 Generic © 2019. Ar > <

aws

😣 🗖 🗊 Netezza - Redshift -- AWS Schema Conversion Tool

()ىگ

aws

😣 🔵 Netezza - Redshift -- AWS Schema Conversion Tool

AWS profile: soo-profile -📙 File 🚀 Actions 📲 View 🔥 Settings 🦳 Applications 🔞 Help Netezza Amazon Redshift Agents Tasks Virtual partitions Snowball T T Description Version Host name Port Status Uptime Last update admin@172.16.118.128:5480 ▼ 🗸 😼 awsdbe@test-for-soo.csanf2qfmbi A Extractor 1.0.629 127.0.0.1 8192 0m 10-07-2019 11:21 ▼ 🗸 🍍 Databases [7] ▼ 🗸 😼 Schemas [22] ▶ 🗸 😑 DBA SCRIPTS 🕨 🗸 🧧 control 🕨 🗸 🧧 ECARE ▶ 🗸 🧧 aws netezza ext ▶ 🗸 😑 EDW 🕨 🗸 😑 aws oracle ext 🕨 🔽 🔚 IBM NETEZZA CUSTOMER SE ▶ 🗸 😑 aws sqlserver ext INVISION V = aws_teradata_ext 🔻 🗸 🎬 TEST V aws_vertica_ext 🔻 🗸 Schemas [1] 🕨 🗸 🔚 control 🔻 🗸 🚔 ADMIN ▶ 🗸 😑 crashdumps Tables [9] 🕨 🗸 🧧 dbc 🕨 🔽 🔚 AFS TRVN MKTSCN A 🕨 🗸 🧧 dbmaster 🕨 🔽 😑 AFS TRVN MKTSCN C 🕨 🗸 🧧 dbo 🕨 🔽 🔚 AFS TRVN MKTSCN C V = dms_sample ► 🗸 🔚 AFS_TRVN_MKTSCN_O 🕨 🗸 😑 online_sales 🕨 🗸 😑 TEST1 🕨 🗸 🧧 public Modify C Refresh all C Refresh selected + Register - Unregister ▶ 🗸 😑 TEST2 🕨 🗸 🗧 sbg_dm 🕨 🗸 🧧 TEST3 🔻 🗸 😼 test_admin ID: 96f9e7824d7744c5b0d9606c1e484e22 ► 🗸 💾 TEST4 🔻 🗸 😼 Tables [1] 3a29dad000c244ce9324eeaaf25ab092 Agent name: TEST5 🕨 🗸 😼 test4 **Description:** Extractor 🗸 😑 External tables 🗸 😑 Views Version: 1.0.629 🗸 😑 Views ▶ 🗸 😑 Functions [1] Host name: 127.0.0.1 V 🔚 Materialized Views Procedures Port: 8192 🗸 🧧 Functions 🕨 🗸 🧧 test dbo ACTIVE Status: V Procedures [1] 🕨 🗸 🧧 test schema Uptime: 0m V 😑 CUSTOMER ▶ 🗸 😑 test schema 0 Last update: 10-07-2019 11:21 Sequences [1] 🕨 🗸 🧧 test schema 1 SSL: false Synonyms 🕨 🗸 😑 test_schema_2 Registration date: 10-07-2019 11:21 🕨 🗸 🔚 nzprod 🕨 🗸 🧧 test user Applications 🗸 🧧 Generic © 2019. Ar



😣 😑 🗉 Netezza - Redshift 🛛 -- AWS Schema Conversion Tool

 Netezza 	Agents Tasks Virt	ual partitions Snow	ball			0	▼ Amazon Redshift
T Te	Task N		Status	Complete %		Elapsed Time	T
▼ = admin@172.16.118.128:5480	▼ TEST4						🔻 🖌 🍍 awsdbe@test-for-soo.csanf2qfr
 Databases [7] 	TEST4(Extract)		•		21% 0h:00m:4	l6s	🔻 🗸 🍍 Schemas [22]
► ✓ GBA_SCRIPTS	TEST4(Upload)		···		0%		► 🗸 😑 _control
ECARE	TEST4(Copy)				0%		V = aws_netezza_ext
► 🔽 🔚 EDW			Ŭ				► ✓ = aws_oracle_ext
► ✓ □ IBM_NETEZZA_CUSTOMER_SE							► ✓ = aws_sqlserver_ext
							► ✓ = aws_teradata_ext
▼ ✓ 🖆 TEST							► 🗸 🥃 aws_vertica_ext
▼ 🖌 🖆 Schemas [1] ▼ 🗸 🖆 ADMIN							► ✓ ⊆ control
▼ ✓ 🗳 ADMIN ▼ ✓ 🗳 Tables [9]							Image: Second
 Fables [9] Fables [9] AFS_TRVN_MKTSCN_A 							► ✓ 📮 dbc
► ✓ = AFS_TRVN_MKTSCN_A							► 🗸 🖨 dbmaster
AFS_TRVN_MKTSCN_C							► ✓ 📮 dbo
► ✓ = AFS_TRVN_MKTSCN_C							► ✓ = dms_sample
► ✓ = TEST1							🕨 🕨 🧧 online_sales
			Start Sto	n 🕨 Restart 👘 D	elete 🦪 Refresh	all (² Refresh selected	
TEST2			Start	op 🕪 Restart 📋 D	elete 🦪 Refresh	n all 🛛 e Refresh selected	► ✓
► ✓ = TEST2	Task details Subta	sks	Start Sto	op 🕪 Restart 🍈 D	elete 🦪 Refresh	n all 🤆 Refresh selected	► ✓ ⊑ sbg_dm
► 🗹 🧮 TEST3	Task details Subta	sks Parent agent	Start Stor	pp I Restart D	elete 🧭 Refresh	n all CRefresh selected	 ▶ ♥
► ✓ = TEST3 ▼ ✓ = TEST4		Parent agent					 ✓ ⊆ sbg_dm ✓ ≝ test_admin ✓ ≝ Tables [1]
► 🗹 🧮 TEST3	ID	Parent agent Extractor	Source name	Target name	Status	Complete simpl	 ✓ ⊆ sbg_dm ✓ ≝ test_admin ✓ ≝ Tables [1] ✓ ✓ ≝ test4
 ► ▼ = TEST3 ▼ ▼ = TEST4 ▼ ■ Constraints 	ID 424ebf27c1fa4178	Parent agent Extractor Extractor	Source name ADMIN.TEST4	Target name test_admin.test4	Status RUNNING	Complete simpl	 ✓ ⊆ sbg_dm ✓ ⊆ test_admin ✓ ⊆ Tables [1]
 TEST3 TEST4 Constraints TEST5 	ID 424ebf27c1fa4178 aaf243ed60a84ba	Parent agent Extractor Extractor Extractor	Source name ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4	Status RUNNING CREATED	Complete simpl 40%	 ✓ ⊆ sbg_dm ✓ ≝ test_admin ✓ ≝ Tables [1] ✓ ≝ test4 ✓ ⊆ Constraints
 TEST3 TEST4 Constraints TEST5 External tables 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437	Parent agent Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING	Complete simpl 40% 0% 47%	 ✓ ⊆ sbg_dm ✓ ⊆ test_admin ✓ ⊆ Tables [1] ✓ ⊆ test4 ✓ ⊆ Constraints ✓ ⊆ Views
 TEST3 TEST4 Constraints TEST5 External tables Views 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def	Parent agent Extractor Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING RUNNING	Complete simpl 40% 0% 47% 38%	 sbg_dm sbg_dm sbg_dmin sbg_dmi
 TEST3 TEST4 Constraints TEST5 External tables Views Naterialized Views 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def ecffea067ae1443b	Parent agent Extractor Extractor Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING	Complete simpl 40% 0% 47% 38% 50%	 Second system is a set of the set of the system is a set of the set of the set of
 TEST3 TEST4 Constraints TEST5 External tables Views Views Materialized Views Functions 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def ecffea067ae1443b 92b5b2f03c5d404	Parent agent Extractor Extractor Extractor Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING CREATED	Complete simpl 40% 0% 47% 38% 50% 0%	 Sbg_dm Sbg_dm Sbg_dm Sbg_dm Sbg_dm Tables [1] Sbg_dm Constraints Sbg_dm Tables [1] Sbg_dm Constraints Sbg_dm Tables [1] Sbg_dm Constraints Sbg_dm Tables [1] Sbg_dm Sb
 TEST3 TEST4 Constraints TEST5 External tables Views Views Materialized Views Functions Procedures [1] 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def ecffea067ae1443b 92b5b2f03c5d404 9a64c6007e6e42c 14fedc9687c9408	Parent agent Extractor Extractor Extractor Extractor Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4	Target name test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING CREATED RUNNING RUNNING RUNNING RUNNING RUNNING RUNNING	Complete simpl 40% 0% 47% 38% 50% 0% 0% 42%	 Sbg_dm Sbg_dm Sbg_dm Sbg_dmin Sbg_dmin Tables [1] Sbg_dmin Sbg_dmi
 TEST3 TEST4 Constraints TEST5 External tables Views Materialized Views Functions Procedures [1] CUSTOMER 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def ecffea067ae1443b 92b5b2f03c5d404 9a64c6007e6e42c 14fedc9687c9408 1f237031c4654d0	Parent agent Extractor Extractor Extractor Extractor Extractor Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4	Target name test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING RUNNING RUNNING RUNNING RUNNING CREATED RUNNING RUNNING CREATED RUNNING CREATED CREATED	Complete simpl 40% 0% 47% 38% 50% 0% 42% 42% 0%	 Second system is solved as a standard system is solve
 TEST3 TEST4 Constraints TEST5 External tables Views Views Functions Procedures [1] CUSTOMER Views [1] 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def ecffea067ae1443b 92b5b2f03c5d404 9a64c6007e6e42c 14fedc9687c9408 1f237031c4654d0 cc52fd54b93348e	Parent agent Extractor Extractor Extractor Extractor Extractor Extractor Extractor Extractor Extractor Extractor	Source name ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING RUNNING RUNNING RUNNING CREATED RUNNING CREATED CREATED CREATED CREATED CREATED CREATED CREATED	Complete simpl 40% 0% 47% 38% 50% 0% 42% 42% 42% 0% 0%	 Sbg_dm Sbg_d
 TEST3 TEST4 Constraints TEST5 External tables Views Views Functions Functions CUSTOMER Sequences [1] Sequences [1] Synonyms 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aa18437 4b8faa85f1fe4def ecffea067ae1443b 92b5b2f03c5d404 9a64c6007e6e42c 14fedc9687c9408 1f237031c4654d0 cc52fd54b93348e 4ad200d8edb34ec	Parent agent Extractor	Source name ADMIN.TEST4	Target name test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING RUNNING RUNNING RUNNING CREATED RUNNING CREATED CREATED CREATED CREATED CREATED CREATED CREATED	Complete simpl 40% 0% 47% 38% 50% 0% 42% 42% 42% 0% 0% 0%	 Second system is subsection. Second system is subsec
 TEST3 TEST4 Constraints TEST5 External tables Views Views Functions Functions Procedures [1] CUSTOMER Sequences [1] Sequences [1] Synonyms Naterial Synonyms 	ID 424ebf27c1fa4178 aaf243ed60a84ba d66a7c8aaa18437 4b8faa85f1fe4def ecffea067ae1443b 92b5b2f03c5d404 9a64c6007e6e42c 14fedc9687c9408 1f237031c4654d0 cc52fd54b93348e	Parent agent Extractor	Source name ADMIN.TEST4 ADMIN.TEST4	Target name test_admin.test4 test_admin.test4	Status RUNNING CREATED RUNNING RUNNING RUNNING RUNNING RUNNING RUNNING CREATED RUNNING CREATED CREATED CREATED CREATED CREATED CREATED CREATED	Complete simpl 40% 0% 47% 38% 50% 0% 42% 42% 42% 0% 0%	 Sbg_dm Sbg_d



Used memory: 448.2 MB, Free memory: 1023.3 MB, Total memory: 1.44 GB, Maximum memory: 3.46 GB

© 2019,

() چ ا

😰 🗖 🗊 Netezza - Redshift 🛛 AWS Schema Conversion Tool 📙 File 🚀 Actions 📲 View 🔥 Settings 🦳 Applications 🕕 Help AWS profile: soo-profile • Amazon Redshift Netezza Agents Tasks Virtual partitions Snowball T T Task Name Status **Complete** % Elapsed Time admin@172.16.118.128:5480 ▼ √ awsdbe@test-for-soo.csanf2qfmbi ▼ TEST4 🔻 🗸 💾 Databases [7] 🔻 🔽 🎽 Schemas [22] (\mathbf{b}) 55% 0h:00m:56s TEST4(Extract) ► 🗸 😑 DBA SCRIPTS 🕨 🗸 🧧 control TEST4(Upload) 32% 0h:00m:04s 🕨 🗸 🗧 ECARE ▶ 🗸 😑 aws_netezza_ext TEST4(Copy) (\mathbf{b}) 0% ▶ 🗸 🔚 EDW ▶ 🗸 🧧 aws oracle ext ▶ 🗸 😑 IBM NETEZZA CUSTOMER SE ▶ 🗸 😑 aws sqlserver ext INVISION 🕨 🗸 🧧 aws teradata ext 🔻 🗸 💾 TEST 🕨 🗸 🧧 aws vertica ext 🔻 🗸 📮 Schemas [1] 🕨 🗸 🧧 control 🔻 🗸 🗎 ADMIN 🕨 🗸 😑 crashdumps 🔻 🗸 Tables [9] 🕨 🗸 🔚 dbc 🕨 🗸 😑 AFS TRVN MKTSCN A 🕨 🗸 🧧 dbmaster AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dbo AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dms sample 🕨 🔽 😑 AFS TRVN MKTSCN O ▶ 🗸 😑 online sales 🕨 🗸 🧧 TEST1 Start Stop Restart Delete Refresh all Refresh selected V = public ▶ 🗸 😑 TEST2 🕨 🗸 🧧 sbg_dm Task details Subtasks 🕨 🗸 🧧 TEST3 🔻 🗸 🍍 test admin ID Parent agent Source name **Target name** Status Complete simpl... 🔻 🗸 🗦 TEST4 🔻 🗸 🍍 Tables [1] 424ebf27c1fa4178... ADMIN.TEST4 test admin.test4 RUNNING Extractor 0% 🗸 🧧 Constraints 🔻 🗸 💾 test4 ADMIN.TEST4 PENDING aaf243ed60a84ba... Extractor test admin.test4 0% TEST5 🗸 🧧 Constraints d66a7c8aaa18437... Extractor ADMIN.TEST4 test admin.test4 RUNNING 0% V 😑 External tables Views 🗸 🧧 Views 4b8faa85f1fe4def... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... ▶ ✓ ⊨ Functions [1] \checkmark Materialized Views ecffea067ae1443b... Extractor ADMIN.TEST4 COMPLETED 10... Procedures test_admin.test4 V E Functions ▶ 🗸 🧧 test dbo 92b5b2f03c5d404.. ADMIN.TEST4 test admin.test4 PENDING 0% Extractor V Procedures [1] 🕨 🗸 😑 test schema 9a64c6007e6e42c... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... V 😑 CUSTOMER ▶ 🗸 😑 test schema 0 14fedc9687c9408... ADMIN.TEST4 test admin.test4 COMPLETED 10... Extractor Sequences [1] V = test_schema_1 1f237031c4654d0... ADMIN.TEST4 test admin.test4 PENDING 0% Extractor 🗸 😑 Synonyms ▶ 🗸 😑 test schema 2 cc52fd54b93348e... ADMIN.TEST4 test admin.test4 PENDING 0% Extractor Image: A state of the state 🕨 🗸 🧧 test user 4ad200d8edb34ec... Extractor ADMIN.TEST4 test admin.test4 PENDING 0% Applications PENDING 0% 4b64bdf2c513484... Extractor ADMIN.TEST4 test_admin.test4 🗸 🧧 Generic 616d3a4c990643e... Extractor ADMIN.TEST4 test admin.test4 PENDING 0% © 2019. An 05a635f75e3d44b... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 🗸

Used memory: 485.61 MB, Free memory: 985.89 MB, Total memory: 1.44 GB, Maximum memory: 3.46 GB

aws



aws

🙆 🗖 🗊 Netezza - Redshift 🛛 AWS Schema Conversion Tool Actions 📲 View ል Settings 🦳 Applications 🔞 Help AWS profile: soo-profile 🔓 File Netezza Amazon Redshift Agents Tasks Virtual partitions Snowball T. T. Task Name Status Complete % Elapsed Time admin@172.16.118.128:5480 awsdbe@test-for-soo.csanf2afmbi ▼ TEST4 ▼ 🗸 💾 Databases [7] ▼ 🗸 🍍 Schemas [22] (\mathbf{b}) 98% 0h:00m:57s TEST4(Extract) ► ✓ 🗧 DBA SCRIPTS 🕨 🗸 🧧 control (\mathbf{b}) 74% 0h:00m:18s TEST4(Upload) 🕨 🗸 🧧 ECARE 🕨 🗸 🔚 aws netezza ext 21% 0h:00m:24s TEST4(Copy) Image: Second ▶ ✓ ⊨ aws oracle ext ► 🗸 🧧 IBM_NETEZZA_CUSTOMER_SE V = aws sqlserver ext 🕨 🗸 🧧 INVISION 🕨 🗸 😑 aws teradata ext 🔻 🗸 TEST V = aws_vertica_ext ▼ 🗸 💾 Schemas [1] 🕨 🗸 🧧 control 🔻 🗸 📛 ADMIN 🕨 🗸 😑 crashdumps ▼ 🗸 💾 Tables [9] 🕨 🗸 🔚 dbc 🕨 🖌 😑 AFS TRVN MKTSCN A 🕨 🗸 😑 dbmaster AFS TRVN MKTSCN C 🕨 🗸 🔚 dbo AFS_TRVN_MKTSCN_C 🕨 🗸 😑 dms sample 🕨 🔽 🔚 AFS TRVN MKTSCN O 🕨 🗸 😑 TEST1 Start Stop Restart 🗍 Delete Refresh all Refresh selected V = public ▶ 🗸 😑 TEST2 🕨 🗸 🗧 sbg_dm Task details Subtasks 🕨 🗸 😑 TEST3 🔻 🗸 💾 test admin ID Parent agent Source name **Target name** Status Complete simpl... ▼ ✓ 💾 TEST4 🔻 🗸 🍍 Tables [1] aa1243e000ao4pa... EXITACIO ADJVIDULE 514 test aumin.test4 PENDING U% 🔻 🗸 💾 test4 d66a7c8aaa18437... Extractor ADMIN.TEST4 test_admin.test4 PENDING 0% TEST5 🗸 😑 Constraints ADMIN.TEST4 4b8faa85f1fe4def... Extractor test admin.test4 PENDING 0% External tables 🗸 🧧 Views ecffea067ae1443b... Extractor ADMIN.TEST4 test admin.test4 RUNNING 50% Views ▶ 🗸 😑 Functions [1] COMPLETED 92b5b2f03c5d404... Extractor ADMIN.TEST4 test_admin.test4 10.. \checkmark Materialized Views Procedures ADMIN.TEST4 test admin.test4 9a64c6007e6e42c... Extractor RUNNING 0% Functions V = test dbo ADMIN.TEST4 RUNNING 14fedc9687c9408... Extractor test admin.test4 50% V Procedures [1] 🕨 🗸 🧧 test_schema ADMIN.TEST4 1f237031c4654d0... test admin.test4 PENDING 0% Extractor V 😑 CUSTOMER ▶ 🗸 😑 test schema 0 ▶ 🗸 😑 Sequences [1] cc52fd54b93348e... Extractor ADMIN.TEST4 test admin.test4 PENDING 0% V = test_schema_1 🗸 😑 Synonyms 4ad200d8edb34ec... Extractor ADMIN.TEST4 PENDING test_admin.test4 0% 🕨 🗸 😑 test schema 2 🕨 🗸 🧧 nzprod 4b64bdf2c513484... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10.. 🕨 🗸 🧧 test user Applications 616d3a4c990643e... Extractor ADMIN.TEST4 test admin.test4 PENDING 0% 🗸 🧧 Generic 05a635f75e3d44b... Extractor ADMIN.TEST4 test admin.test4 RUNNING 50% 3cf3bc6d455f46f8... Extractor ADMIN.TEST4 test admin.test4 PENDING 0%

© 2019, Am

Used memory: 543.74 MB, Free memory: 927.76 MB, Total memory: 1.44 GB, Maximum memory: 3.46 GB



😰 🗖 🗊 Netezza - Redshift 🛛 AWS Schema Conversion Tool ² Actions 🛛 💾 View 🔥 Settings 🛛 🔚 Applications 🛛 🔞 Help AWS profile: soo-profile 📙 File Amazon Redshift Netezza Agents Tasks Virtual partitions Snowball T T Task Name Status **Complete** % Elapsed Time admin@172.16.118.128:5480 ▼ √ awsdbe@test-for-soo.csanf2qfmbi ▼ TEST4 ▼ 🗸 🍍 Databases [7] 🔻 🔽 🎽 Schemas [22] \odot 100% 0h:00m:57s TEST4(Extract) ► 🗸 😑 DBA SCRIPTS 🕨 🗸 🧧 control \odot TEST4(Upload) 100% 0h:00m:18s 🕨 🗸 🗧 ECARE ▶ 🗸 😑 aws_netezza_ext (\mathbf{b}) 5% 0h:01m:38s ▶ 🗸 🔚 EDW ▶ 🗸 😑 aws oracle ext ▶ 🗸 😑 IBM NETEZZA CUSTOMER SE ▶ 🗸 😑 aws sqlserver ext ► 🗸 🧧 INVISION 🕨 🗸 🧧 aws teradata ext 🔻 🗸 💾 TEST 🕨 🗸 🧧 aws vertica ext 🔻 🗸 📮 Schemas [1] 🕨 🗸 🧧 control 🔻 🗸 🗎 ADMIN 🕨 🗸 😑 crashdumps 🔻 🗸 Tables [9] 🕨 🗸 🔚 dbc 🕨 🗸 😑 AFS TRVN MKTSCN A 🕨 🗸 🧧 dbmaster AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dbo AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dms sample 🕨 🔽 😑 AFS TRVN MKTSCN O ▶ 🗸 😑 online sales 🕨 🗸 🧧 TEST1 Start Stop Restart Delete Refresh all Refresh selected V = public ► 🗸 😑 TEST2 🕨 🗸 🧧 sbg_dm Task details Subtasks TEST3 🔻 🗸 🍍 test admin ID Parent agent Target name Status Complete simpl... Source name 🔻 🗸 🗦 TEST4 🔻 🗸 🍍 Tables [1] aa1243euovao4pa... EXITACIO ADMIN. LES 14 test_admin.test4 PENDING U 70 🗸 🧧 Constraints 🔻 🗸 💾 test4 d66a7c8aaa18437... Extractor ADMIN.TEST4 test admin.test4 RUNNING 50% TEST5 🗸 🧧 Constraints 4b8faa85f1fe4def... Extractor ADMIN.TEST4 test admin.test4 RUNNING 50% 🗸 🧧 External tables Views ecffea067ae1443b... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 🗸 🧧 Views ▶ ✓ ⊨ Functions [1] COMPLETED 92b5b2f03c5d404... Extractor ADMIN.TEST4 test_admin.test4 10... \checkmark Materialized Views Procedures 9a64c6007e6e42c... Extractor ADMIN.TEST4 test admin.test4 RUNNING 50% V E Functions ▶ 🗸 🧧 test dbo ADMIN.TEST4 COMPLETED 14fedc9687c9408... Extractor test admin.test4 10... V Procedures [1] 🕨 🗸 🧧 test schema 1f237031c4654d0.. ADMIN.TEST4 test admin.test4 RUNNING Extractor 50% V 😑 CUSTOMER ▶ 🗸 😑 test schema 0 Sequences [1] cc52fd54b93348e... Extractor ADMIN.TEST4 test admin.test4 RUNNING 50% Image: Section of the section of V Synonyms 4ad200d8edb34ec... Extractor ADMIN.TEST4 PENDING test_admin.test4 0% ▶ 🗸 😑 test schema 2 Image: A start of the start 4b64bdf2c513484... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10.. 🕨 🗸 🧧 test user Applications 616d3a4c990643e... Extractor ADMIN.TEST4 test admin.test4 PENDING 0% 🗸 🧧 Generic 05a635f75e3d44b... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 3cf3bc6d455f46f8... Extractor

ADMIN.TEST4

test admin.test4

RUNNING

50%

© 2019. An

Used memory: 632.75 MB, Free memory: 838.75 MB, Total memory: 1.44 GB, Maximum memory: 3.46 GB

aws



😰 🗖 🗊 Netezza - Redshift 🛛 AWS Schema Conversion Tool 🕈 Actions 🛛 💾 View 🔥 Settings 🛛 🔚 Applications 🛛 🔞 Help AWS profile: soo-profile 📙 File Amazon Redshift Netezza Agents Tasks Virtual partitions Snowball T T Task Name Status **Complete** % Elapsed Time admin@172.16.118.128:5480 ▼ √ awsdbe@test-for-soo.csanf2qfmbi ▼ TEST4 ▼ 🗸 🍍 Databases [7] 🔻 🔽 🎽 Schemas [22] \odot 100% 0h:00m:57s TEST4(Extract) ► 🗸 😑 DBA SCRIPTS 🕨 🗸 🧧 control \odot TEST4(Upload) 100% 0h:00m:18s 🕨 🗸 🗧 ECARE ▶ 🗸 😑 aws_netezza_ext 100% 0h:01m:54s ▶ 🗸 🔚 EDW ▶ 🗸 😑 aws oracle ext ▶ 🗸 😑 IBM NETEZZA CUSTOMER SE ▶ 🗸 😑 aws sqlserver ext ► 🗸 🧧 INVISION 🕨 🗸 🧧 aws teradata ext 🔻 🗸 💾 TEST 🕨 🗸 🧧 aws vertica ext 🔻 🗸 Schemas [1] 🕨 🗸 🧧 control 🔻 🗸 🗎 ADMIN 🕨 🗸 😑 crashdumps 🔻 🗸 Tables [9] 🕨 🗸 🔚 dbc 🕨 🗸 😑 AFS TRVN MKTSCN A 🕨 🗸 🧧 dbmaster AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dbo AFS_TRVN_MKTSCN_C 🕨 🗸 🧧 dms sample 🕨 🔽 😑 AFS TRVN MKTSCN O ▶ 🗸 😑 online sales 🕨 🗸 🧧 TEST1 Start Stop Restart Delete Refresh all Refresh selected V = public ► 🗸 😑 TEST2 🕨 🗸 🧧 sbg_dm Task details Subtasks TEST3 🔻 🗸 🍍 test admin ID Parent agent Target name Status Complete simpl... Source name 🔻 🗸 🗦 TEST4 🔻 🗸 🍍 Tables [1] aa1243euovao4pa... EXITACIO ADMIN. LES 14 test_admin.test4 COMPLETED 10... 🗸 🧧 Constraints 🔻 🗸 💾 test4 d66a7c8aaa18437... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... TEST5 🗸 🧧 Constraints 4b8faa85f1fe4def... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 🗸 🧧 External tables Views ecffea067ae1443b... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 🗸 🧧 Views ▶ ✓ ⊨ Functions [1] 92b5b2f03c5d404... Extractor ADMIN.TEST4 test_admin.test4 COMPLETED 10... \checkmark Materialized Views Procedures COMPLETED 9a64c6007e6e42c... Extractor ADMIN.TEST4 test admin.test4 10... V E Functions ▶ 🗸 🧧 test dbo ADMIN.TEST4 COMPLETED 14fedc9687c9408... Extractor test admin.test4 10... V Procedures [1] 🕨 🗸 🧧 test schema 1f237031c4654d0.. ADMIN.TEST4 test admin.test4 COMPLETED Extractor 10... V 😑 CUSTOMER ▶ 🗸 😑 test schema 0 Sequences [1] cc52fd54b93348e... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... Image: Section of the section of V Synonyms 4ad200d8edb34ec... Extractor ADMIN.TEST4 test_admin.test4 COMPLETED 10... ▶ 🗸 😑 test schema 2 Image: A start of the start 4b64bdf2c513484... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 🕨 🗸 🧧 test user Applications 616d3a4c990643e... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 🗸 🧧 Generic 05a635f75e3d44b... Extractor ADMIN.TEST4 test admin.test4 COMPLETED 10... 3cf3bc6d455f46f8... Extractor COMPLETED

ADMIN.TEST4

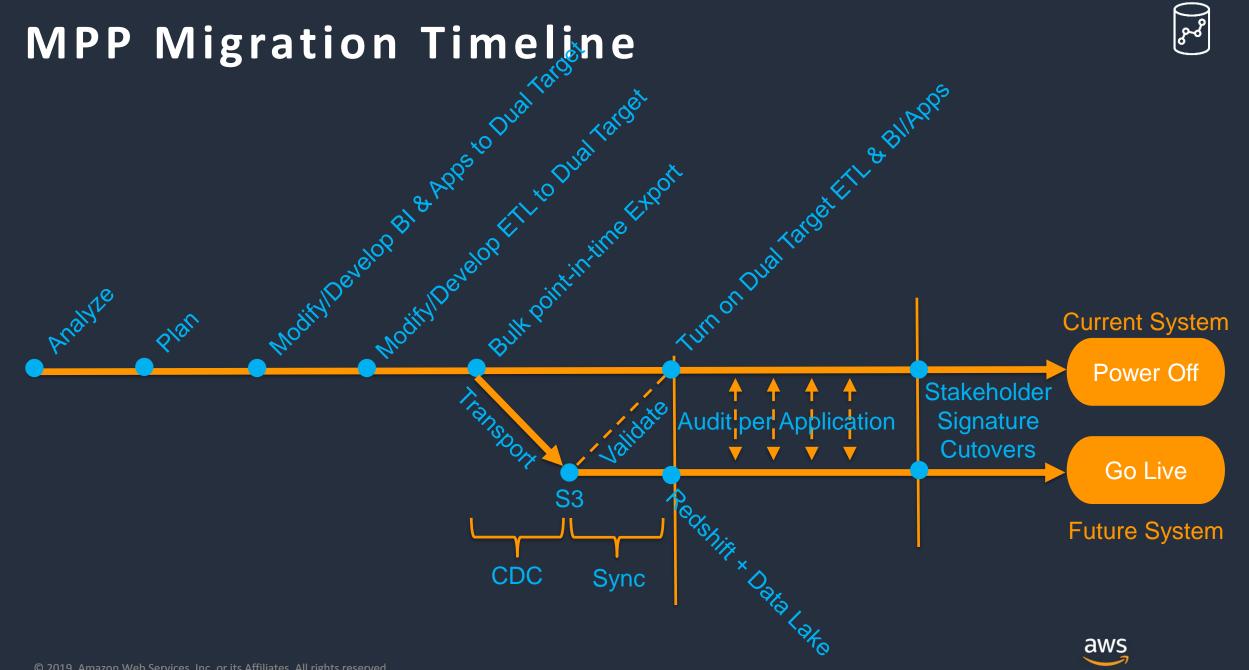
test admin.test4

10...

© 2019. An

Used memory: 687.33 MB, Free memory: 784.17 MB, Total memory: 1.44 GB, Maximum memory: 3.46 GB

aws



Resources



- Try it out for yourself: <u>https://aws.amazon.com/redshift/</u>
- Redshift Quick Start: https://aws.amazon.com/quickstart/architecture/amazon-redshift/
- Netezza to Redshift Migration without downtime:

https://aws.amazon.com/blogs/big-data/how-to-migrate-from-ibm-netezza-to-amazon-redshift-with-no-downtime/

• Data Warehouse Migration using ASWS Schema Conversion Tool:

<u>https://aws.amazon.com/blogs/database/how-to-migrate-your-data-warehouse-to-amazon-redshift-using-the-aws-schema-</u> <u>conversion-tool-data-extractors/</u>

- Github Lab on On-Premises DW to Amazon Redshift with AWS SCT: https://github.com/wrbaldwin/da-week/tree/master/Labs/DW-Migration-to-Redshift-using-SCT
- Use virtual partitioning in the AWS Schema Conversion Tool: <u>https://aws.amazon.com/blogs/database/use-virtual-partitioning-in-the-aws-schema-conversion</u>



Questions?







Thanks!

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

