ORACLE

What's new in Autonomous Database in 2023

Oct 2023

Sandesh Rao

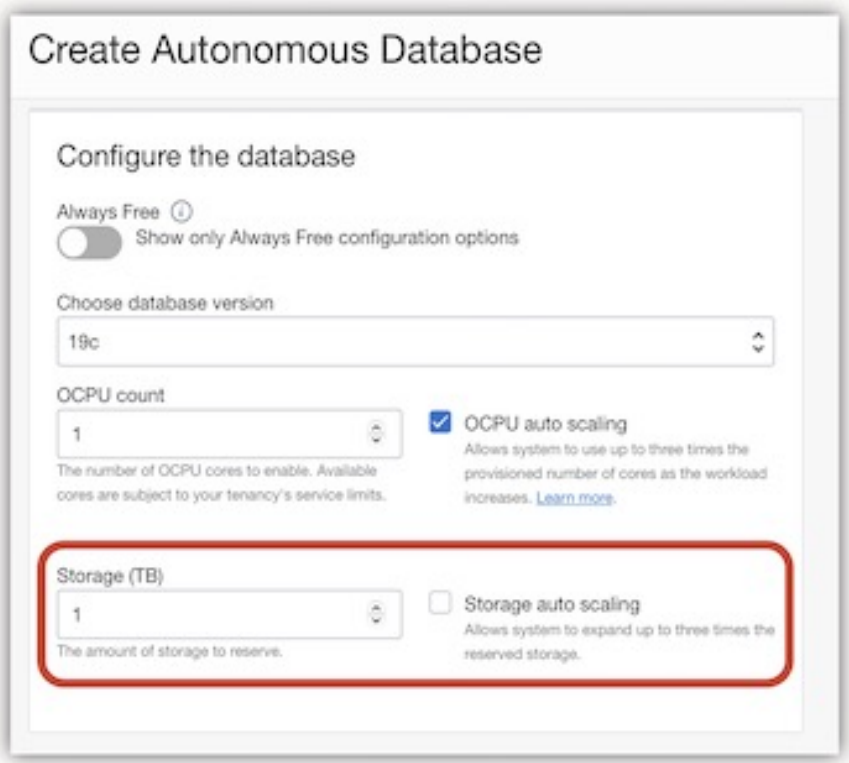
VP AIOps for the Autonomous

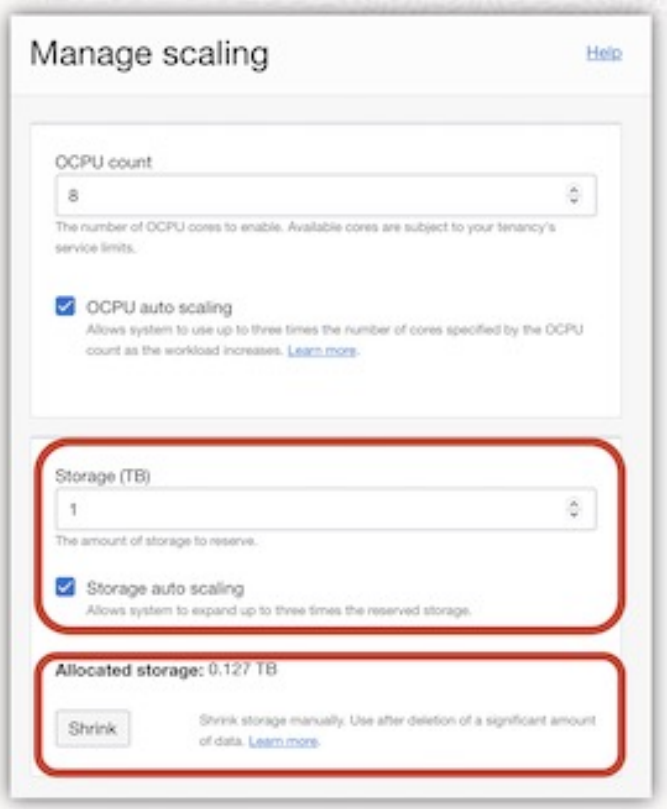
 @sandeshr

 https://www.linkedin.com/in/raosandesh/

 https://www.slideshare.net/SandeshRao4

Autonomous Database now provides automatic scaling for storage



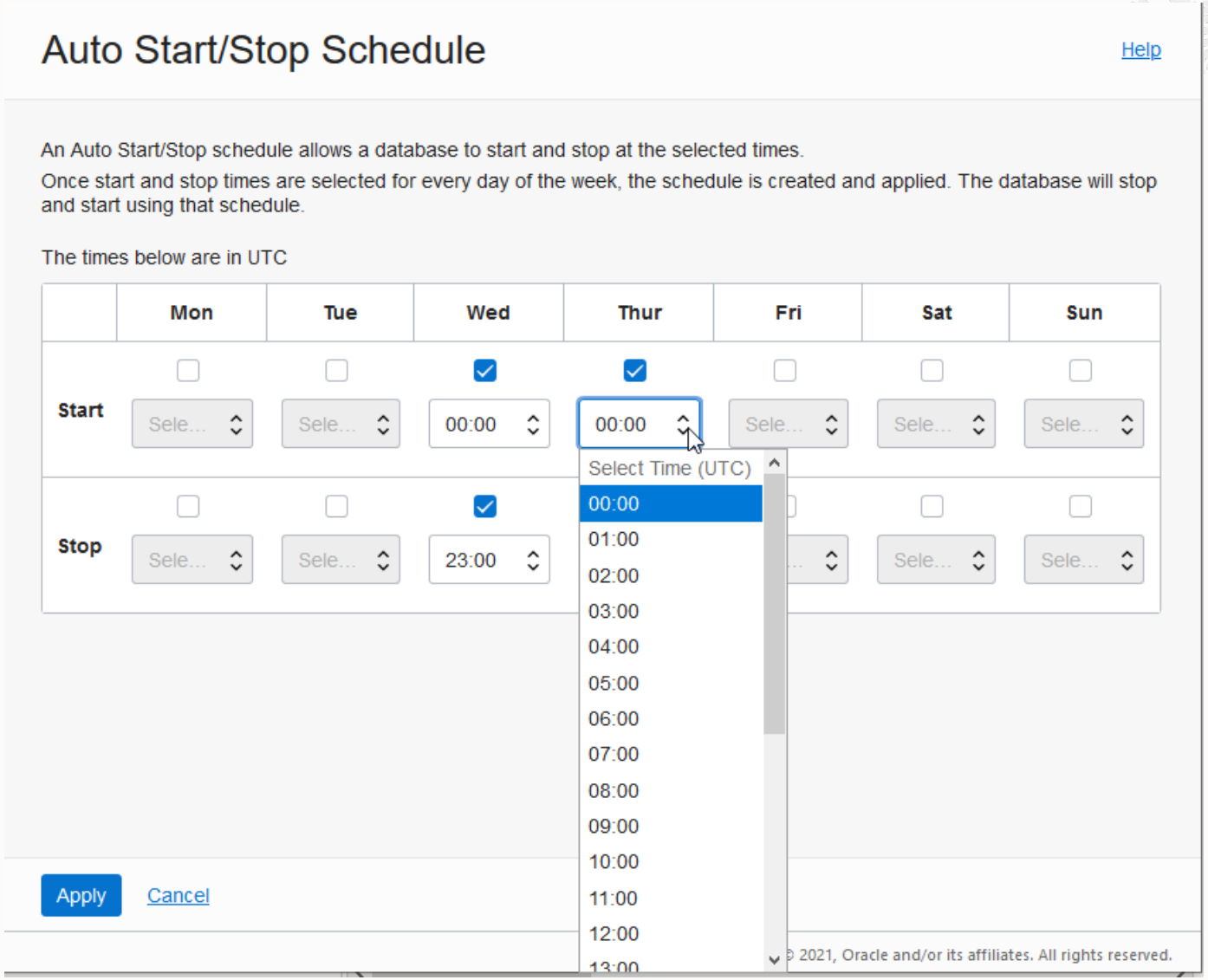


Autonomous Database now provides automatic scaling for storage

Works like auto-scaling for OCPUs

Will automatically expand the base reserved storage by up to 3x

Defining A Calendar Schedule to Stop and Start An Autonomous Database

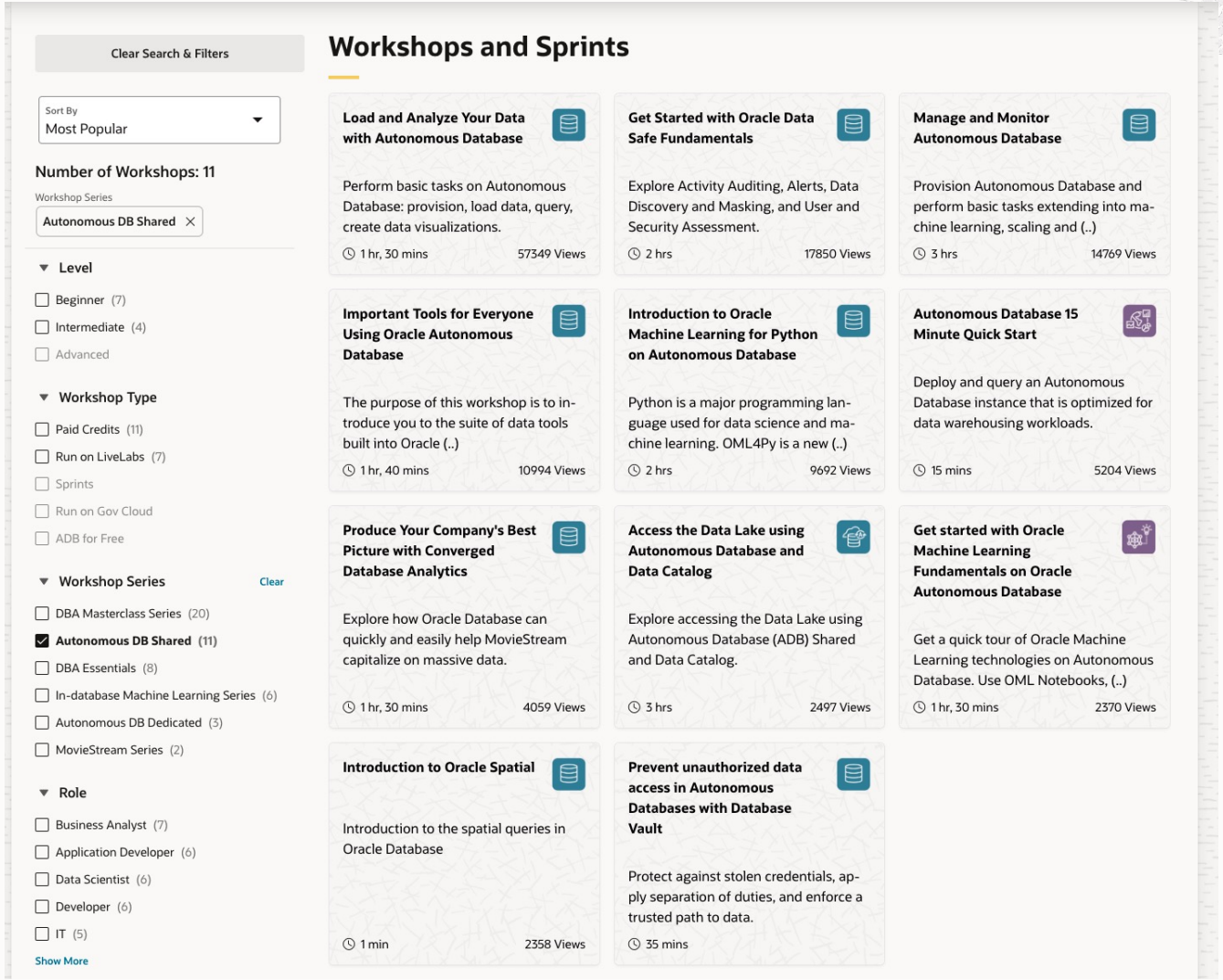


Schedule start and stop times for an Autonomous Database instance

Instance automatically starts and stops according to the specified schedule

Allows you to reduce costs by scheduling shutdown periods when a system is not in use

Getting Started workshops for Autonomous Database



https://bit.ly/get-started-with-adb

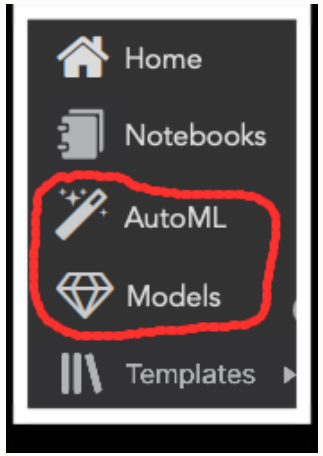
AutoML “Code-free” user interface supporting automated end-to-end machine learning

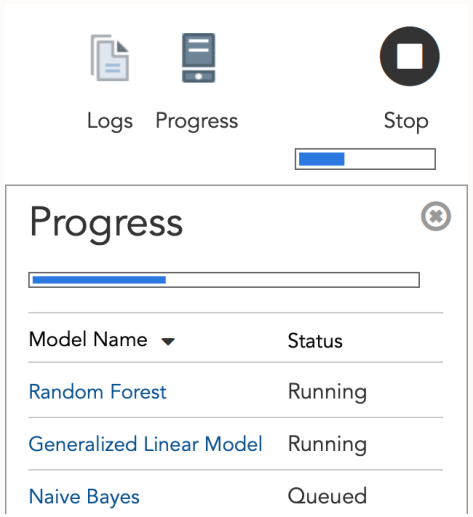
Automate production and deployment of ML models

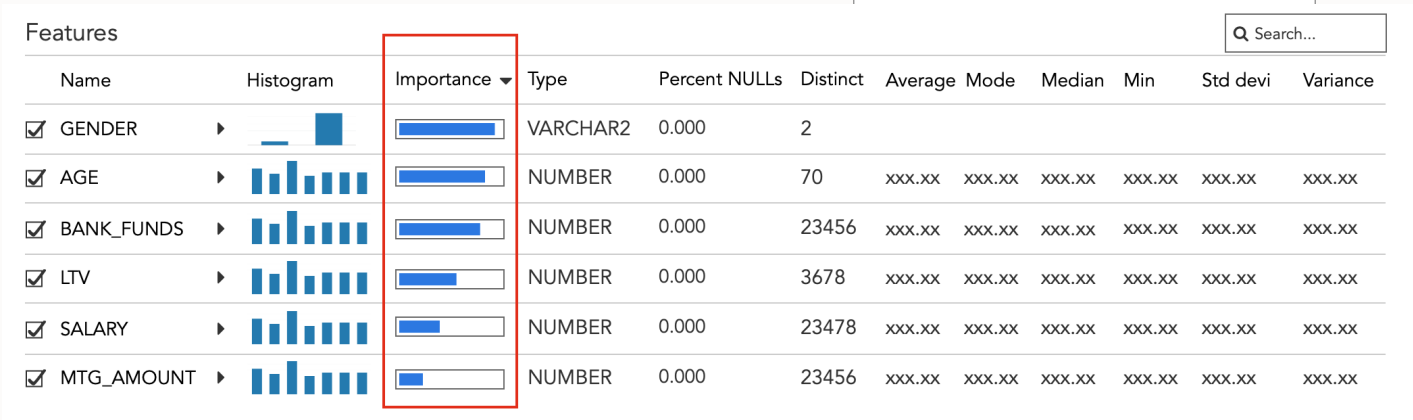
- Enhance Data Scientist productivity and user-experience
- Enable non-expert users to leverage ML
- Unify model deployment and monitoring
- Support model management

Features

- Minimal user input: data, target
- Model leaderboard
- Model deployment via REST
- Model monitoring
- Cognitive features for image and text

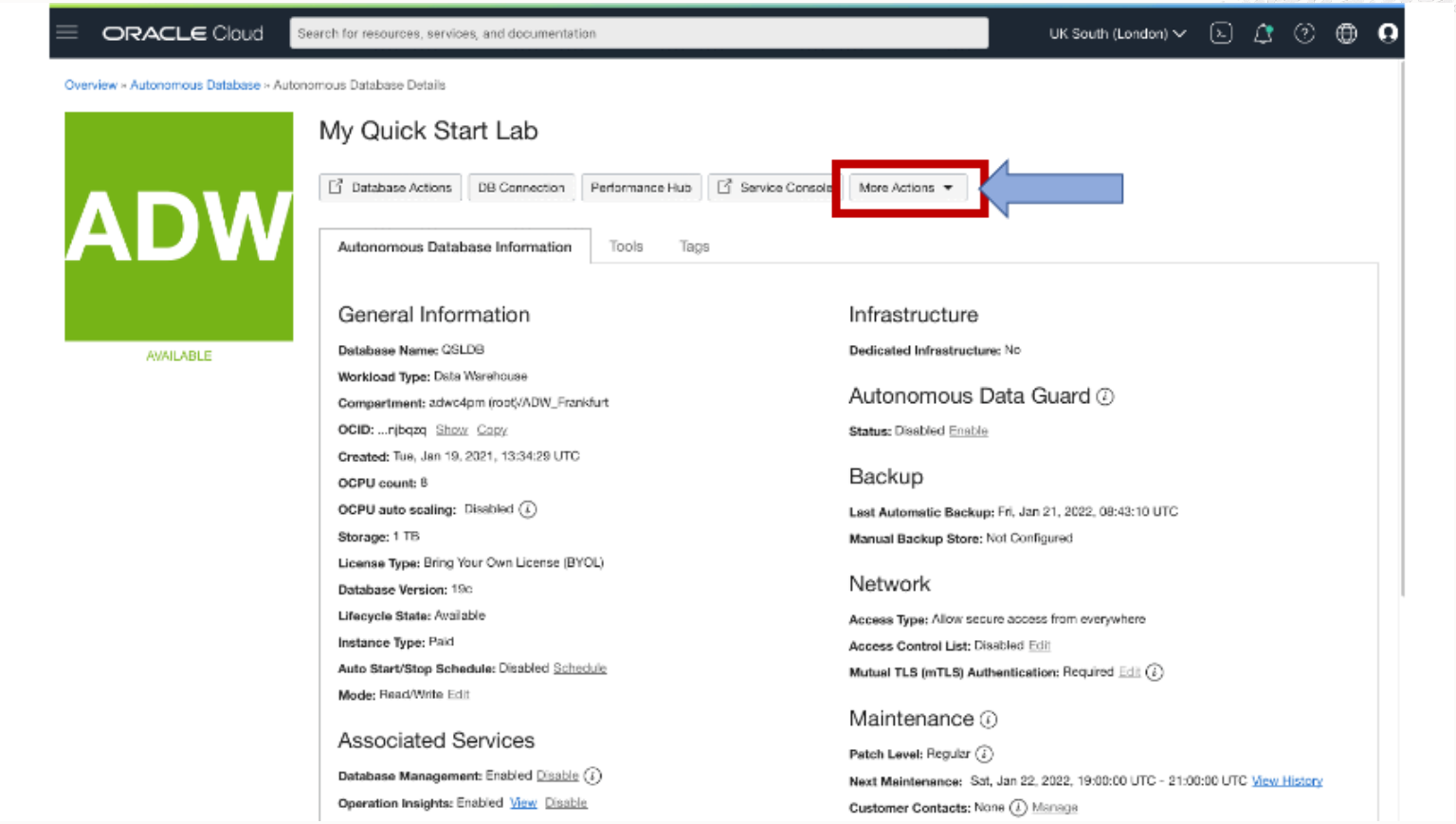






Copyright © 2020 Oracle and/or its affiliates.

Autonomous Database Now Supports Cross-Region Clones

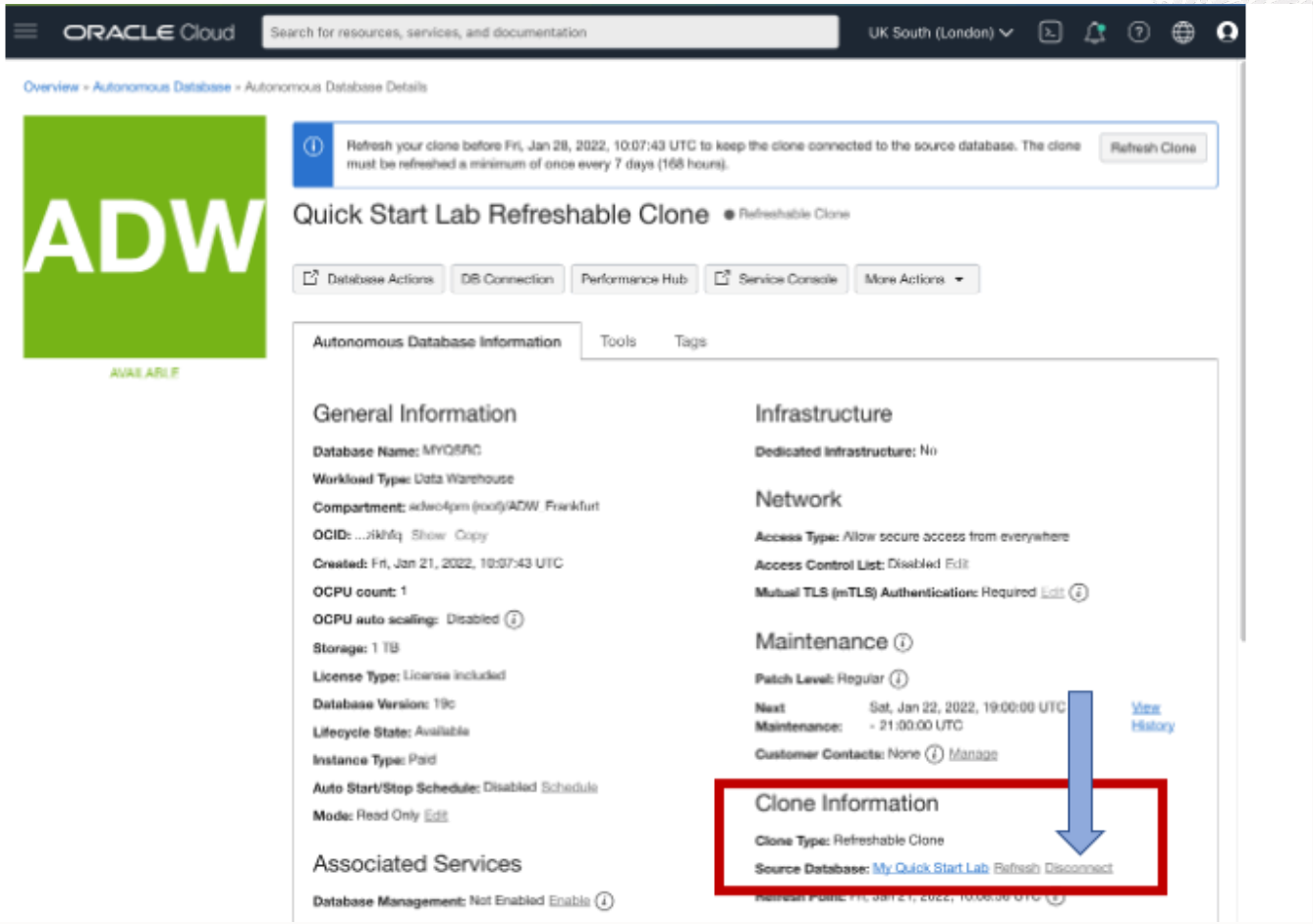


Clone an Autonomous Database instance & select your preferred region for the clone

Create the clone in the current region or a remote region

Available regions list limited to regions you are subscribed to

Reconnect a Disconnected Refreshable Clone



After you disconnect a refreshable clone, during the following 24-hour period you can now reconnect that refreshable clone to the source database

Allows you to use a refreshable clone as a test database, run DML, and make changes while the database is disconnected

When you are done with your testing, you can reconnect to the source database, which refreshes the clone to the point where you disconnected.

Data As JSON To Object Storage

OVERVIEW

Autonomous Database now has a procedure to export a query as JSON directly to Object Storage bucket.

The query can be an advanced query - includes joins or subqueries.

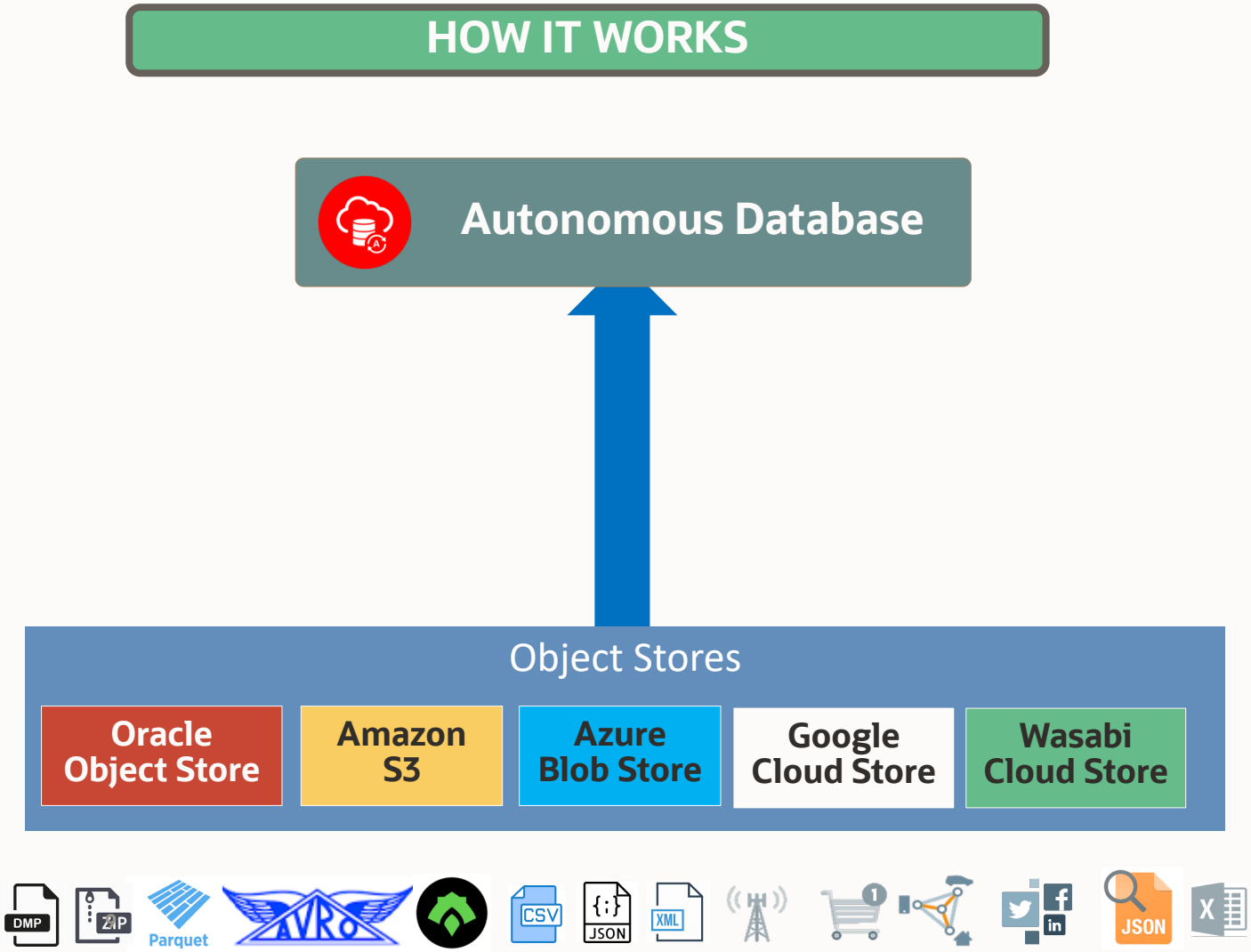
Specify format parameter with compression option to compress the output files.

Use DBMS_CLOUD.DELETE_OBJECT to delete the files

HOW IT WORKS

```
BEGIN

DBMS_CLOUD.EXPORT_DATA(
credential_name => 'DEF_CRED_NAME',
file_uri_list =>
'bucketname/filename',
query => 'SELECT * FROM DEPT',
format => JSON_OBJECT('type' value
'json'));

END; /
```

Autonomous Database – Accessing Data In Object Stores

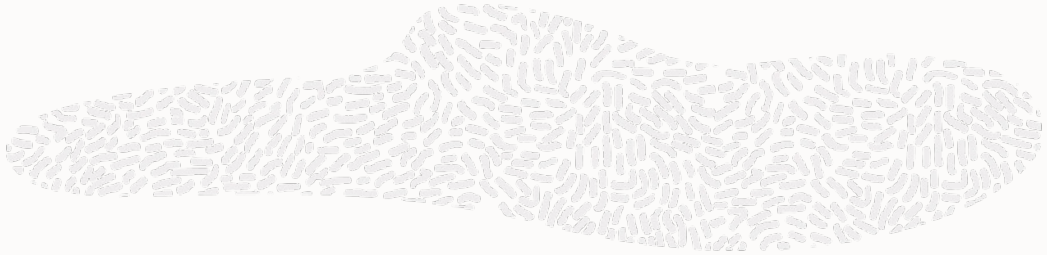


- ADB provides transparent access over data in Object Stores
- Easily join across data sets in the "*data lake*" and in-database data sets
- Leverage in-file metadata with Avro, ORC and Parquet to simplify creating tables

Copyright © 2021, Oracle and/or its affiliates

Hybrid Partitioned Tables

Note only use of DBMS_CLOUD syntax is supported



```
BEGIN DBMS_CLOUD.CREATE_HYBRID_PART_TABLE(
 table_name =>'HPT1',
 credential_name =>'OBJ_STORE_CRED',
 format => json_object('delimiter' value ',', '
         recorddelimiter' value 'newline', '
         characterset' value 'us7ascii'),
 column_list => 'col1 number, col2 number, col3 number'
 partitioning_clause => 'partition by range (col1)
    (partition p1 values less than (1000) external location (
         'https://swiftobjectstorage.us-ashburn-1 .../file_01.txt') ,
     partition p2 values less than (2000) external location (
         'https://swiftobjectstorage.us-ashburn-1 .../file_02.txt') ,
     partition p3 values less than (3000) ) )
END;
```

External tables with partitioning specified in source files

Partitioning is a well-established technique to improve the performance and manageability of database systems by dividing large objects into smaller partitions; any large data warehouse takes advantage of it

```
BEGIN
DBMS_CLOUD.CREATE_EXTERNAL_PART_TABLE(
  TABLE_NAME => 'sales_new_api',
  CREDENTIAL_NAME => 'CRED_OCI',
  FILE_URI_LIST => 'https://objectstorage.us-ashburn-
1.oraclecloud.com/n/my_namespace/b/moviestream_landing/o/sales_sample/*.parquet',
  FORMAT => '{"type":"parquet", "schema":
"first","partition_columns":[{"name":"month","type":"varchar2(100)"}]}'

);
END;

/
```

External tables with partitioning specified in source files

```
SQL> explain plan for SELECT * FROM sales_new_api where month='2019-11';

Explained.

SQL> select plan_table_output from table(dbms_xplan.display('plan_table',null,'basic'));

                                          PLAN_TABLE_OUTPUT
_________________________________________________________________
Plan hash value: 3087785210

_____________________________________________________________
| Id  | Operation                            | Name          |
_____________________________________________________________
|   0 | SELECT STATEMENT                     |               |
|   1 |  PARTITION LIST SINGLE               |               |
|   2 |   EXTERNAL TABLE ACCESS STORAGE FULL| SALES_NEW_API |
_____________________________________________________________

9 rows selected.

SQL> SELECT day_id, month from sales_new_api where rownum<3;

                            DAY_ID      MONTH
_____________________________________ __________
01-JAN-19 12.00.00.000000000 AM    2019-01
01-JAN-19 12.00.00.000000000 AM    2019-01
```

We now derive the column structure for self-describing table formats with partitioned external tables, just like with nonpartitioned external tables

External tables with partitioning specified in source files

If new files are added or removed in the underlying Object Store, you just run the new sync procedure like this:

```
BEGIN

DBMS_CLOUD.SYNC_EXTERNAL_PART_TABLE (table_name => 'sales_new_api');

END;

/
```




Set Patch Level When Creating A Clone

OVERVIEW

When you provision or clone an Autonomous Database instance you can select a patch level to apply upcoming patches.

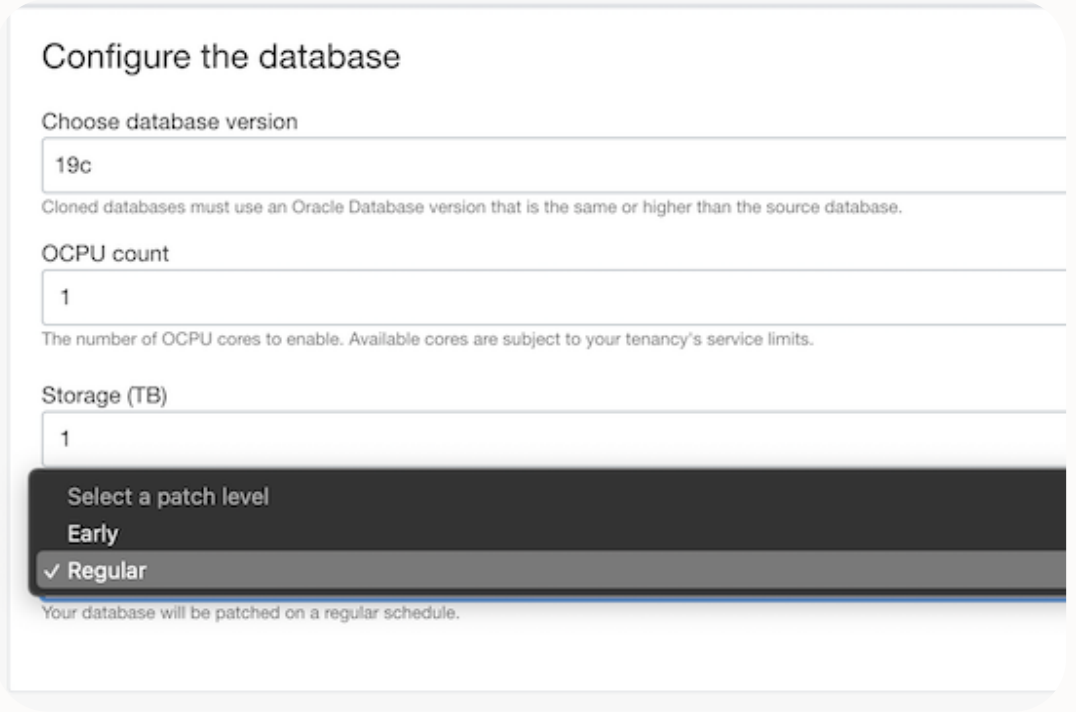
There are two patch level options: **Regular** and **Early**.

The Early patch level allows testing upcoming patches one week before they are applied as part of the regular patching program

The console shows the patch level setting with the section headed Maintenance.



HOW IT WORKS

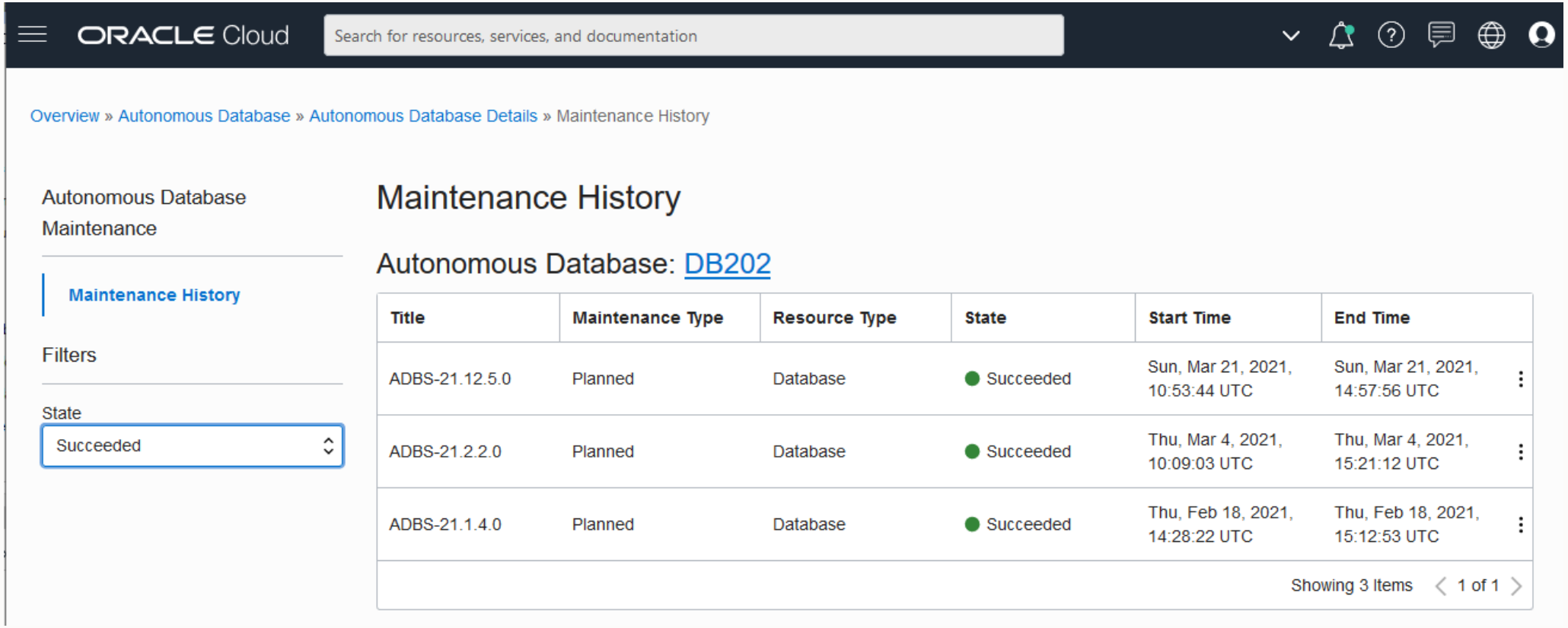


View Patch Details

OVERVIEW

View Autonomous Database maintenance event history to see details about past maintenance events

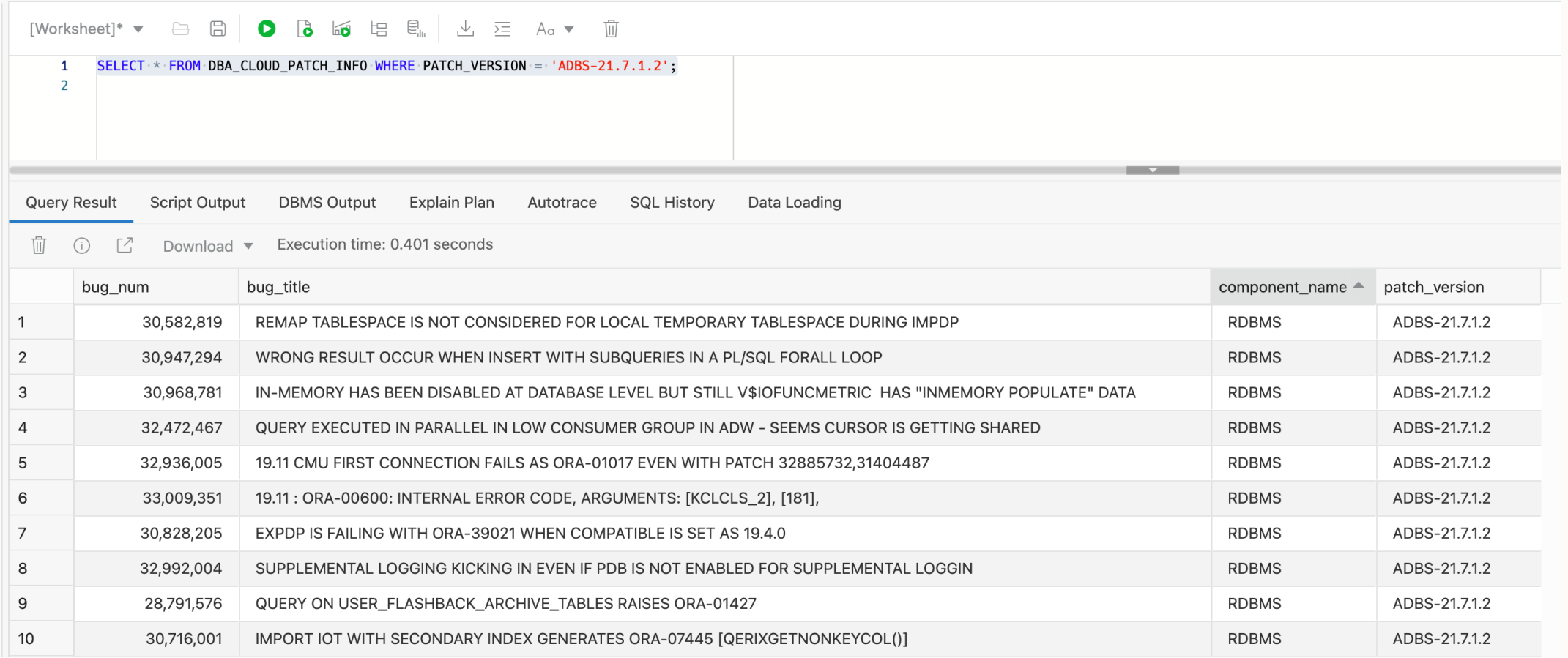
(requires ADMIN user)



View Patch Details

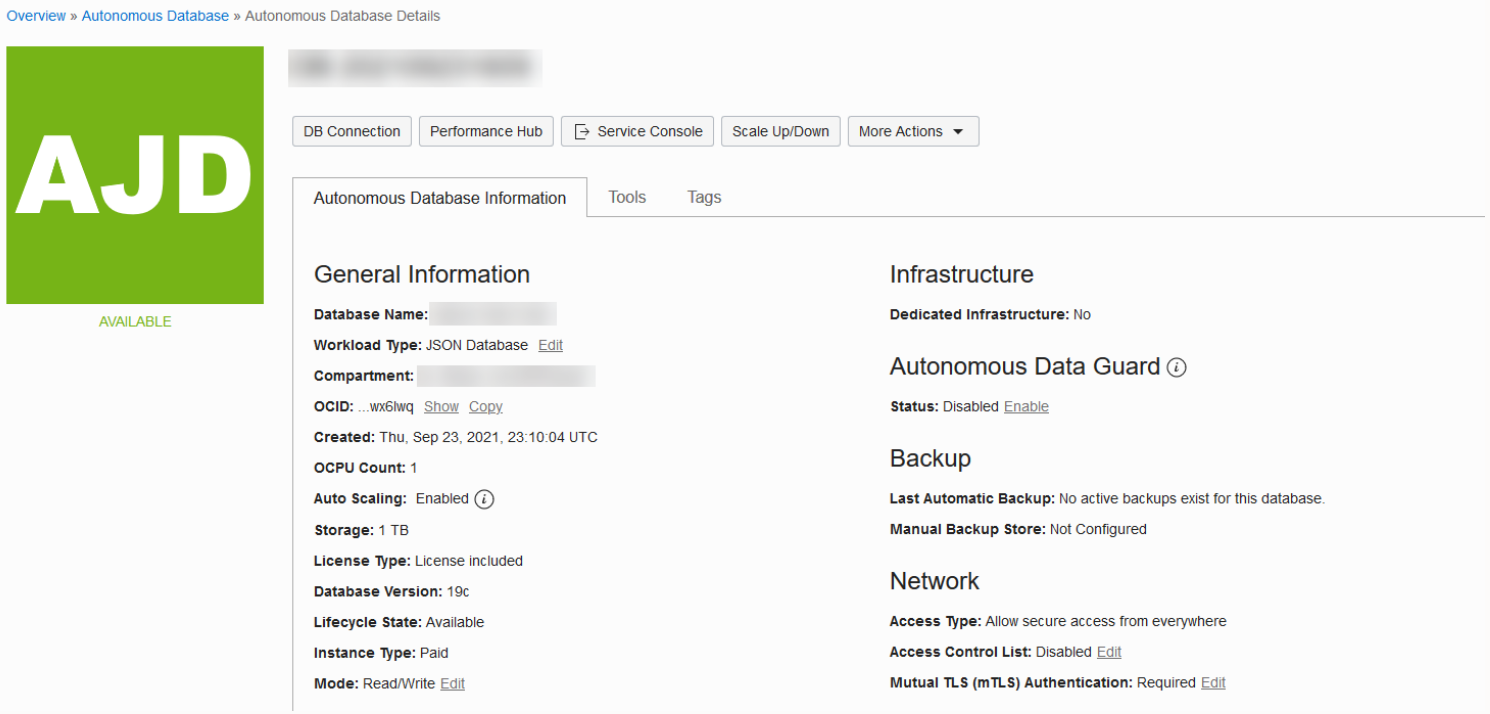
HOW IT WORKS

```
SELECT * FROM DBA_CLOUD_PATCH_INFO;

SELECT * FROM DBA_CLOUD_PATCH_INFO WHERE PATCH_VERSION = 'ADBS-21.7.1.2';
```



	bug_num	bug_title	component_name	patch_version
1	30,582,819	REMAP TABLESPACE IS NOT CONSIDERED FOR LOCAL TEMPORARY TABLESPACE DURING IMPDP	RDBMS	ADBS-21.7.1.2
2	30,947,294	WRONG RESULT OCCUR WHEN INSERT WITH SUBQUERIES IN A PL/SQL FORALL LOOP	RDBMS	ADBS-21.7.1.2
3	30,968,781	IN-MEMORY HAS BEEN DISABLED AT DATABASE LEVEL BUT STILL V$IOFUNCMETRIC HAS "INMEMORY POPULATE" DATA	RDBMS	ADBS-21.7.1.2
4	32,472,467	QUERY EXECUTED IN PARALLEL IN LOW CONSUMER GROUP IN ADW - SEEMS CURSOR IS GETTING SHARED	RDBMS	ADBS-21.7.1.2
5	32,936,005	19.11 CMU FIRST CONNECTION FAILS AS ORA-01017 EVEN WITH PATCH 32885732,31404487	RDBMS	ADBS-21.7.1.2
6	33,009,351	19.11 : ORA-00600: INTERNAL ERROR CODE, ARGUMENTS: [KCLCLS_2], [181],	RDBMS	ADBS-21.7.1.2
7	30,828,205	EXPDP IS FAILING WITH ORA-39021 WHEN COMPATIBLE IS SET AS 19.4.0	RDBMS	ADBS-21.7.1.2
8	32,992,004	SUPPLEMENTAL LOGGING KICKING IN EVEN IF PDB IS NOT ENABLED FOR SUPPLEMENTAL LOGGIN	RDBMS	ADBS-21.7.1.2
9	28,791,576	QUERY ON USER_FLASHBACK_ARCHIVE_TABLES RAISES ORA-01427	RDBMS	ADBS-21.7.1.2
10	30,716,001	IMPORT IOT WITH SECONDARY INDEX GENERATES ORA-07445 [QERIXGETNONKEYCOL()]	RDBMS	ADBS-21.7.1.2

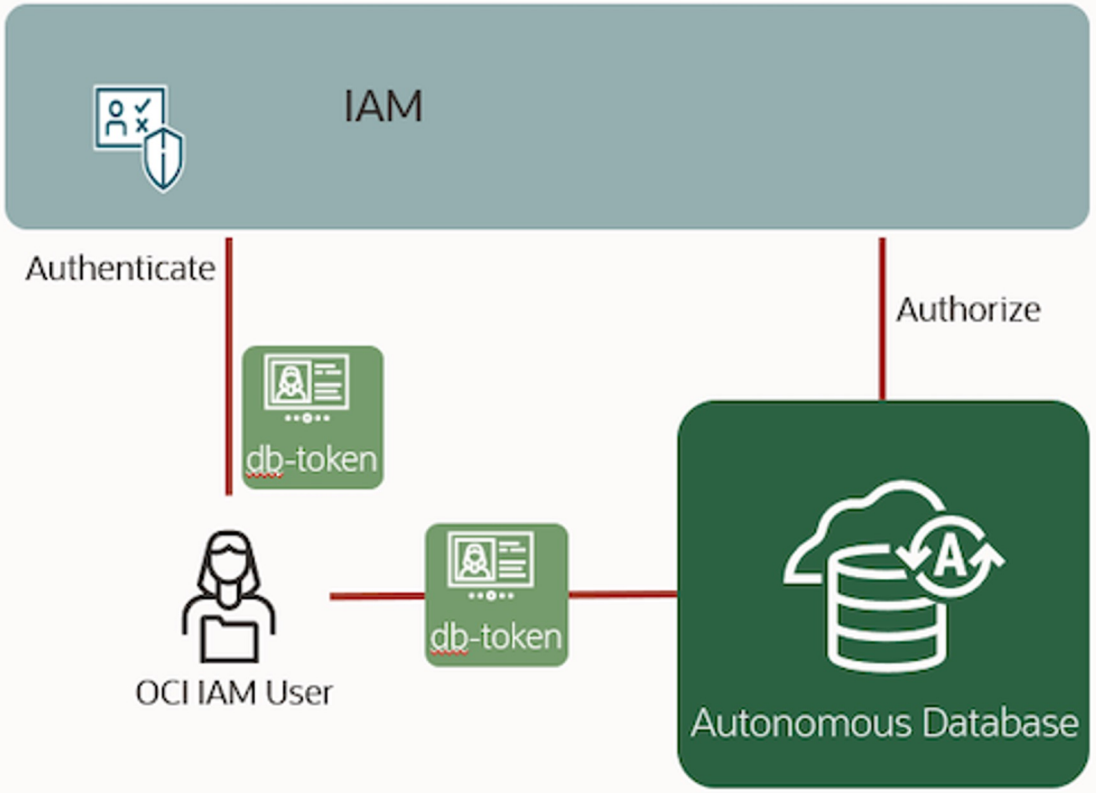
Oracle Database API for MongoDB



Oracle Database API for MongoDB enables developers to connect applications directly to Oracle Autonomous Database using MongoDB's own drivers and tools

Developers can combine existing MongoDB skills with full power of Autonomous Database

Integration With OCI Identity and Access Management (IAM) Authentication



OCI Identity and Access Management users can now authenticate and authorize to ADB-Shared.

Better security since user access to databases is managed centrally instead of locally in every database

Reduces zombie database user accounts

User management moves DBA tasks to the IAM administrator

SQL*Plus users can sign into Autonomous Database using their IAM username and IAM database password

Users can also use IAM SSO tokens with the latest JDBC-thin and OCI-C database clients to connect with ADB-Shared

Access Resources on Azure



Simplified access to resources on Azure

1. Enable Azure service principal authentication in Autonomous Database

Enable the Azure service principal authentication in Autonomous Database instance

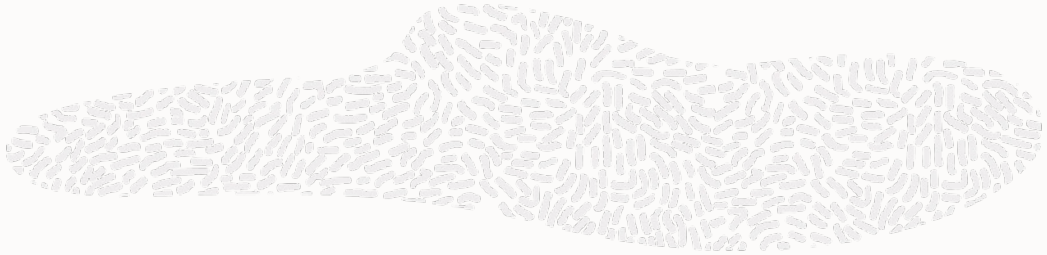
Needs Azure Active Directory (AD) tenant ID for this so that my ADB-S instance can be properly associated with my Azure tenancy and registered as an authorized application in

```
SQL> BEGIN
  2      DBMS_CLOUD_ADMIN.ENABLE_PRINCIPAL_AUTH(
  3          provider => 'AZURE',
  4          params   => JSON_OBJECT('azure_tenantid' value '74***-****-****-****-****67'));
  5  END;
  6  /

PL/SQL procedure successfully completed.
```

see How to find your Azure Active Directory tenant ID

Simplified access to resources on Azure

2. Provide Azure application consent



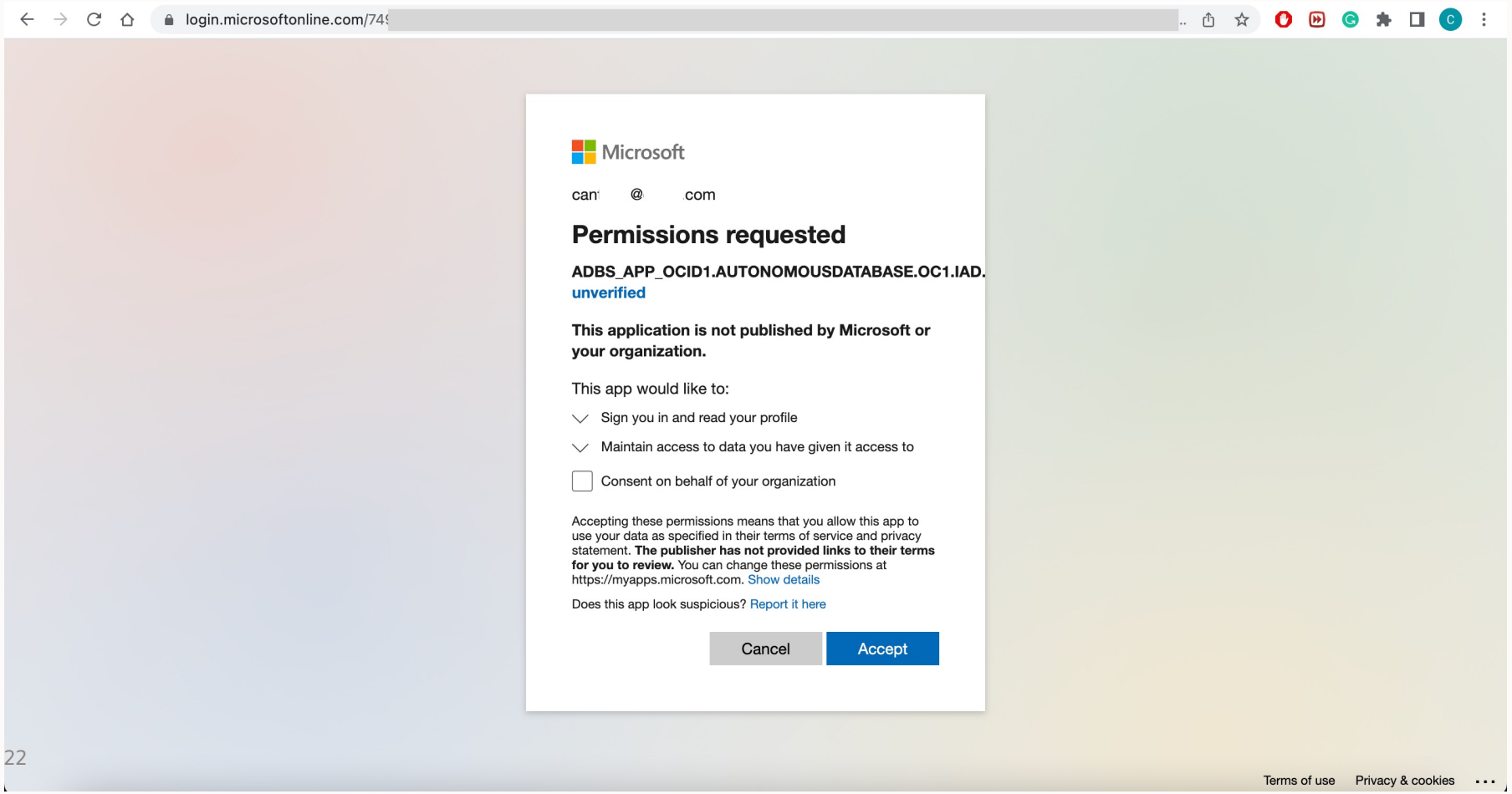
Query the CLOUD_INTEGRATIONS view to find the **azure_consent_url and azure_app_name**

```
SQL> SELECT param_name, param_value FROM CLOUD_INTEGRATIONS;

PARAM_NAME           PARAM_VALUE
-------------------  ------------------------------------------------------------
-------
azure_tenantid       74***-****-****-****-****67
azure_consent_url    https://login.microsoftonline.com/'74***-****-****-
****_****67'/oauth2/v2.0/***&scope=User.read
azure_app_name       ADBS_APP_OCID1.AUTONOMOUSDATABASE.OC1.IAD.******VFA
```

Simplified access to resources on Azure

3. Provide Azure application consent

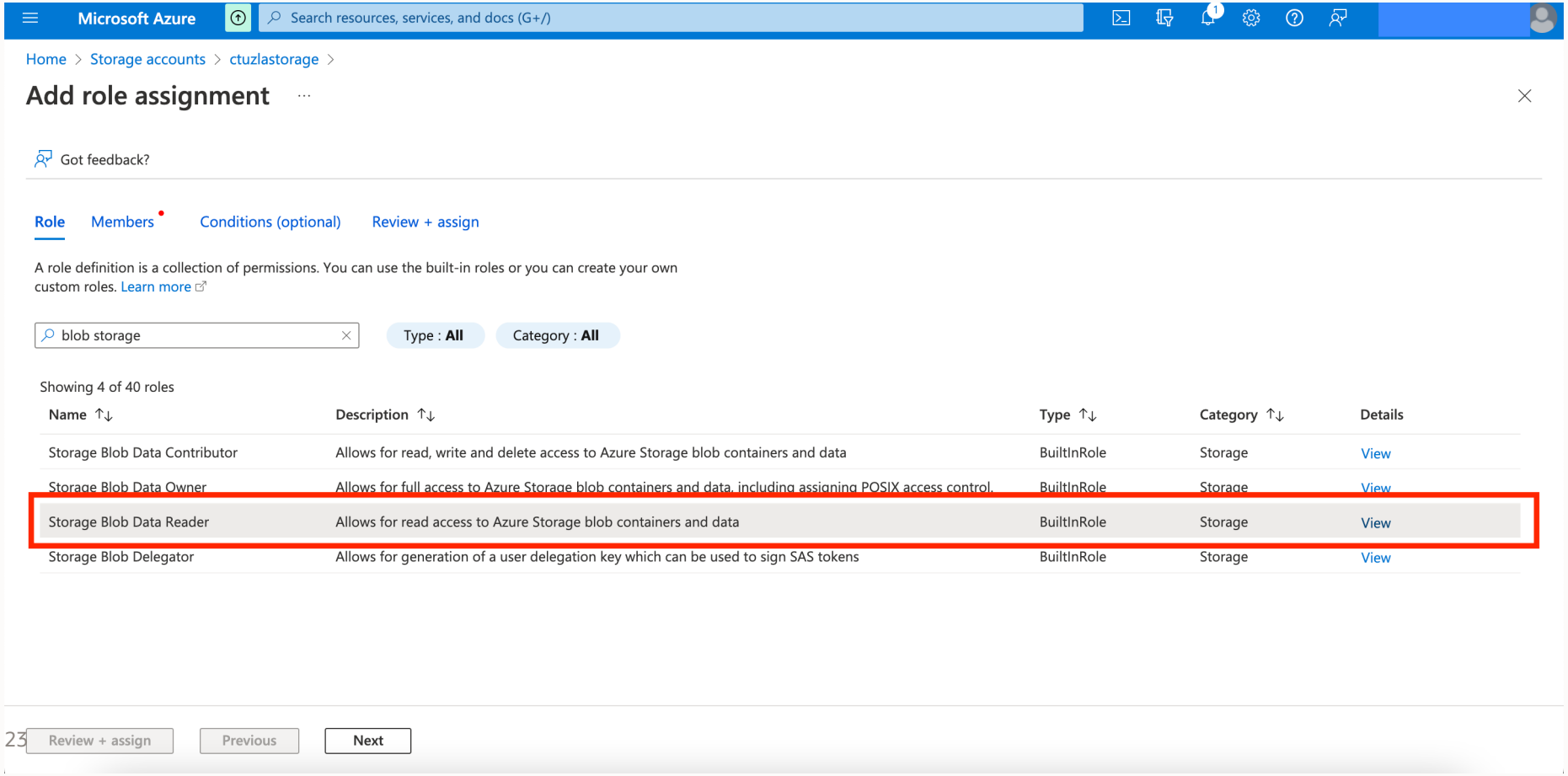
Visit the URL from azure_consent_url and accept the request:



Simplified access to resources on Azure

4. Assign necessary roles for the application to access the Azure resources

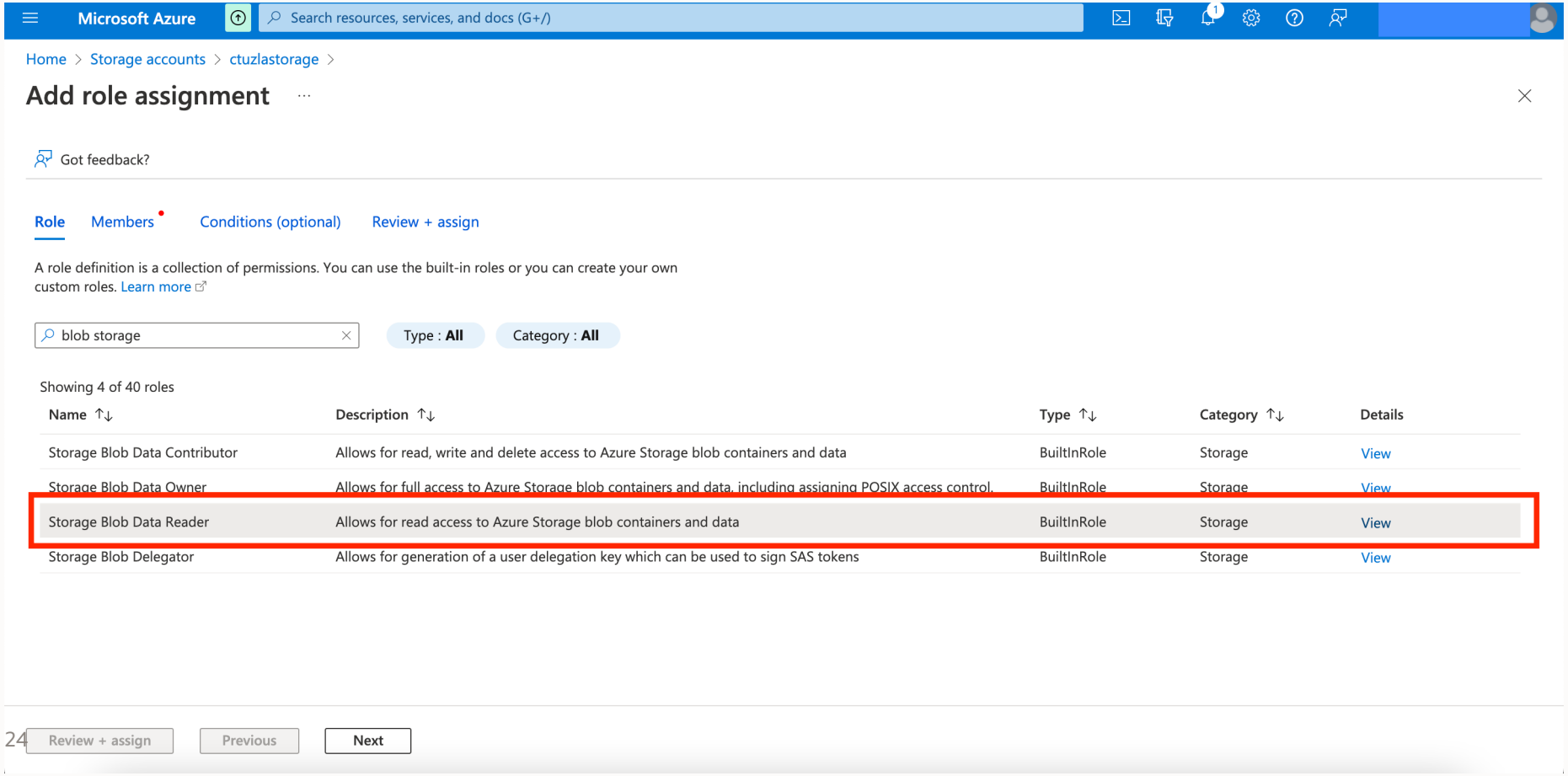
Navigate to *Azure console home -> Storage accounts -> <myStorageAccountname> -> Access Control (IAM) -> Add -> Add Role Assignment -> Role* and search for “blob storage”



Simplified access to resources on Azure

4. Assign necessary roles for the application to access the Azure resources

Navigate to *Members -> Select members* and copy/paste the the value of **azure_app_name**



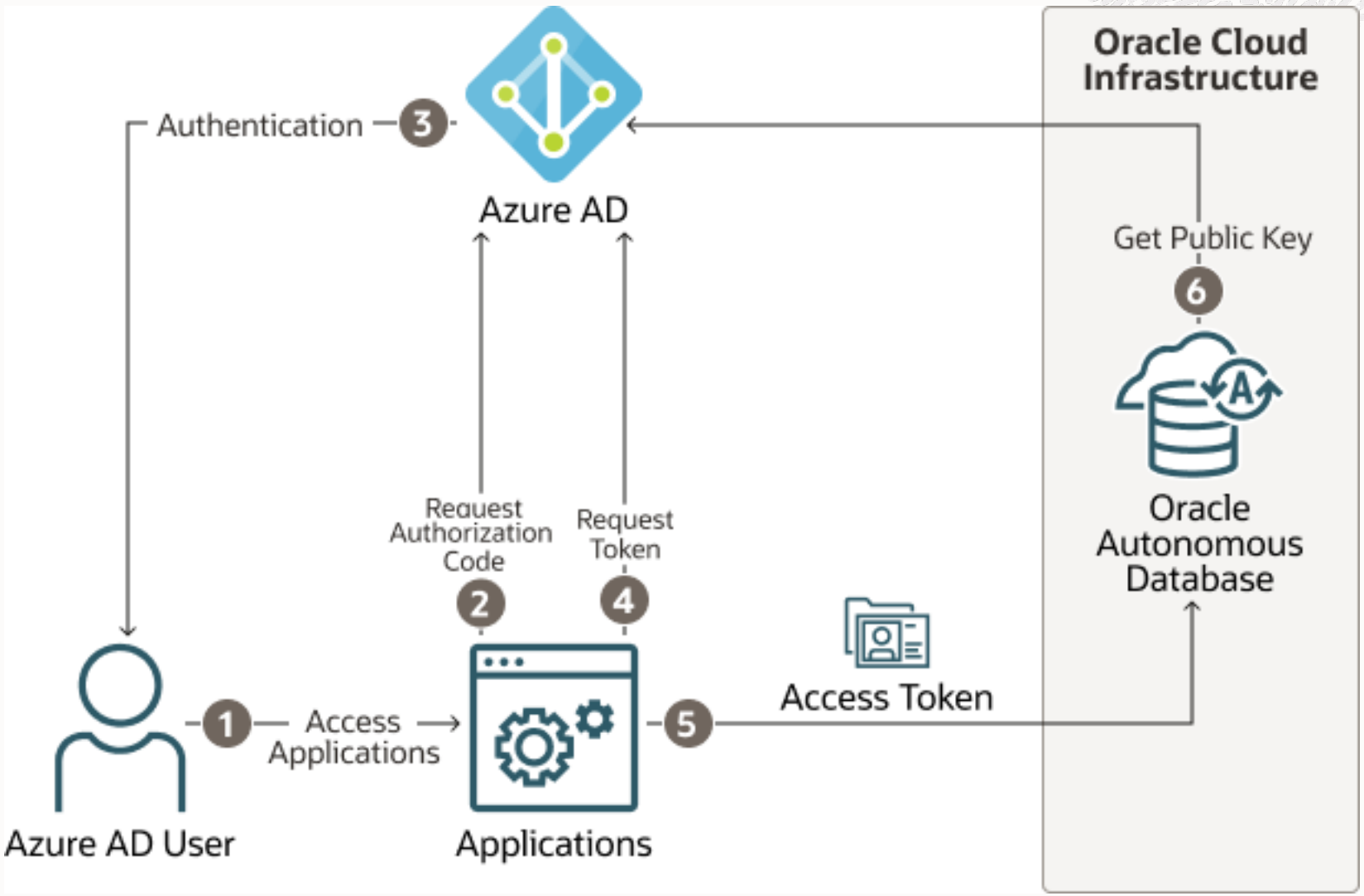
Simplified access to resources on Azure

5. Load data from Azure blob storage via Azure service principal

Load some data from my Azure blob storage
AZURE$PA is the credential_name in DBMS_CLOUD.COPY_DATA procedure:

```
SQL> CREATE TABLE CHANNELS
  2    (channel_id CHAR(1),
  3     channel_desc VARCHAR2(20),
  4     channel_class VARCHAR2(20)
  5     );
Table CHANNELS created.

SQL> BEGIN
  2    DBMS_CLOUD.COPY_DATA(
  3      table_name =>'CHANNELS',
  4      credential_name =>'AZURE$PA',
  5      file_uri_list =>'https://ctuzlastorage.blob.core.windows.net/ctuzlabucket/chan_v3.dat',
  6      format => json_object('ignoremissingcolumns' value 'true', 'removequotes' value 'true')
  7    );
  8  END;
  9  /

PL/SQL procedure successfully completed.

SQL> select count(*) from channels;


   COUNT(*)
___________
         10
```

Use Azure Active Directory to centrally manage database users



Autonomous Database users can be centrally managed in Microsoft Azure Active Directory (Azure AD) for stronger security instead of locally in every database

Azure AD users and applications can use Azure AD OAuth2 access tokens to access the Autonomous Database

Azure AD users don't have to remember separate database credentials; they use their existing Azure AD SSO credentials

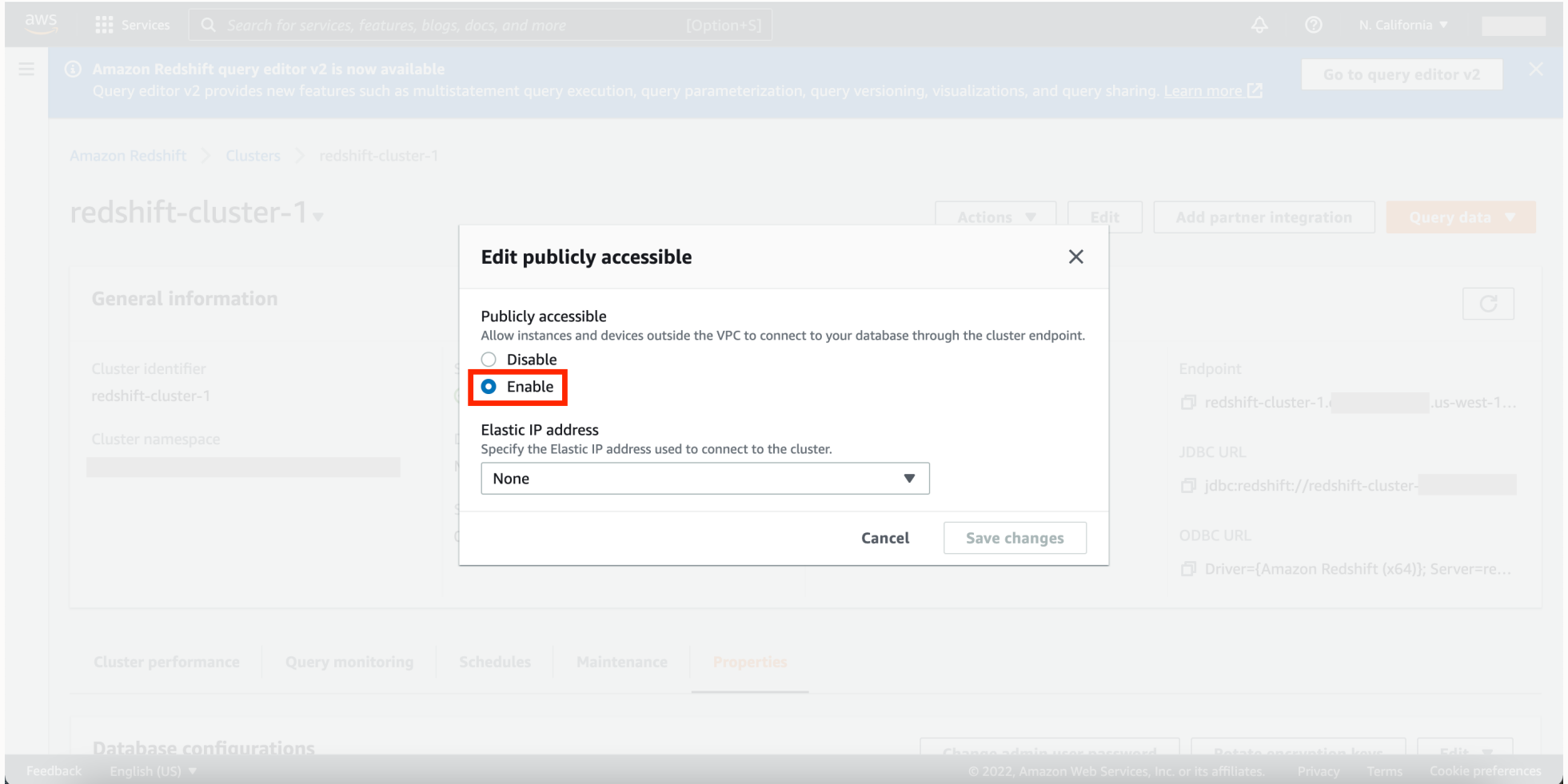
Connect to non-Oracle Databases



Connect to Non-Oracle databases using built-in heterogeneous connectivity

Step-1: Make Sure Redshift is Configured to Allow Public Access

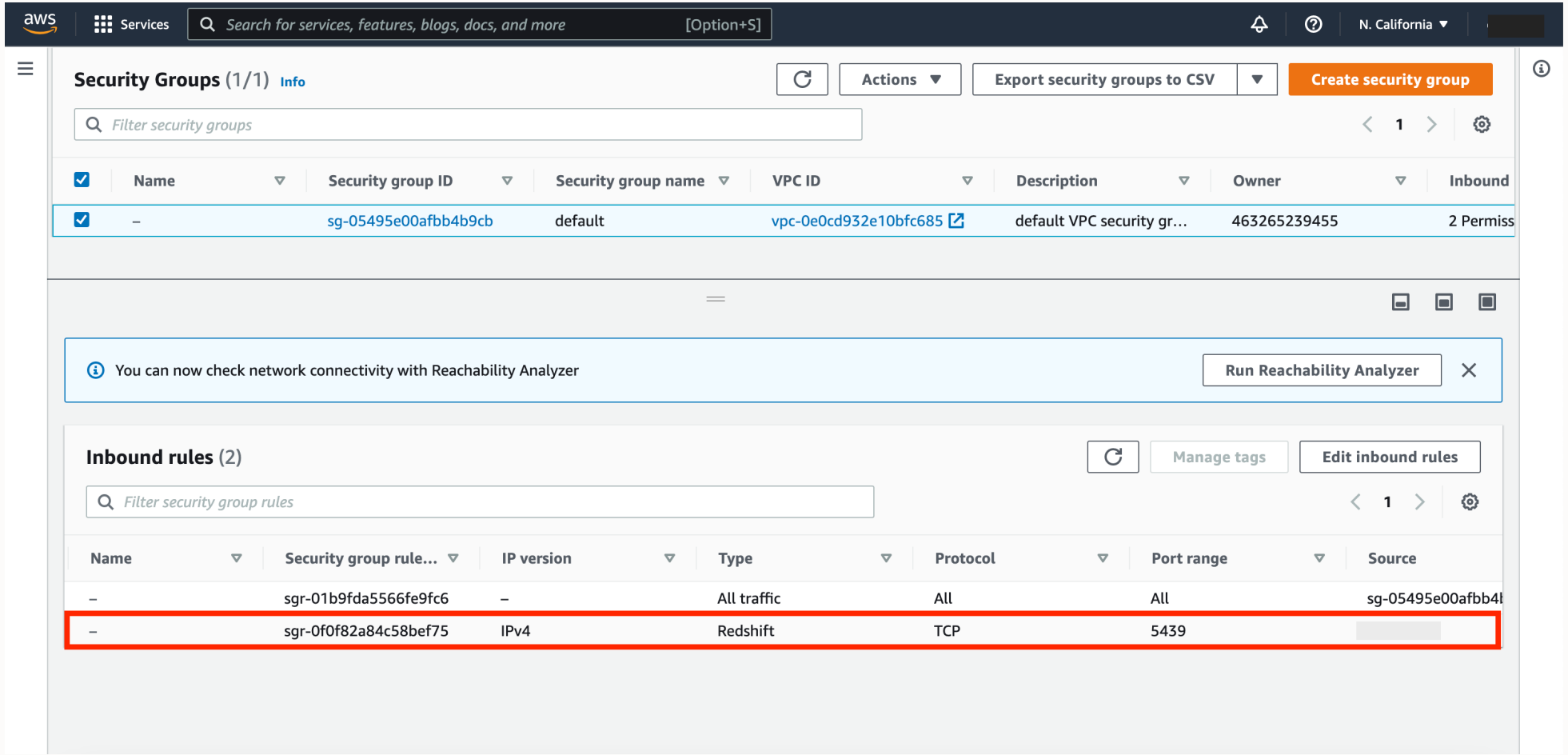
Under “Actions” menu of your Redshift cluster on AWS console, we need to select “Modify publicly accessible setting” to make sure our Redshift cluster is publicly accessible:



Connect to Non-Oracle databases using built-in heterogeneous connectivity

Step-1: Make Sure Redshift is Configured to Allow Public Access

We then navigate to the VPC security group that is assigned to our Redshift cluster and create an inbound rule to port 5439 from the source IP or CIDR range of our choice:



Connect to Non-Oracle databases using built-in heterogeneous connectivity

Step-2: Create a Database Link to your Redshift Instance

First, create a credential object with the credentials (username and password) of the target database:

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL(
   credential_name => 'REDSHIFT_CRED’,
   username => 'awsadmin’,
   password => '************’);
END;
/

PL/SQL procedure successfully completed.
```

Connect to Non-Oracle databases using built-in heterogeneous connectivity

Step-2: Create a Database Link to your Redshift Instance

Next, we will create a database link to our Redshift instance. As you can tell, it looks nearly identical to any other database link creation except for the gateway_params parameter for which we are passing 'awsredshift' as our database type:

```
BEGIN
     DBMS_CLOUD_ADMIN.CREATE_DATABASE_LINK(
          db_link_name => 'REDSHIFT_LINK',
          hostname => 'redshift-cluster-1.******.us-west-1.redshift.amazonaws.com',
          port => '5439',
          service_name => 'dev',
          credential_name => 'REDSHIFT_CRED',
          gateway_params => JSON_OBJECT('db_type' value 'awsredshift'),
          ssl_server_cert_dn => NULL);
END;
/

PL/SQL procedure successfully completed.
```

Connect to Non-Oracle databases using built-in heterogeneous connectivity

Step-3: Run a Query Over the Database Link

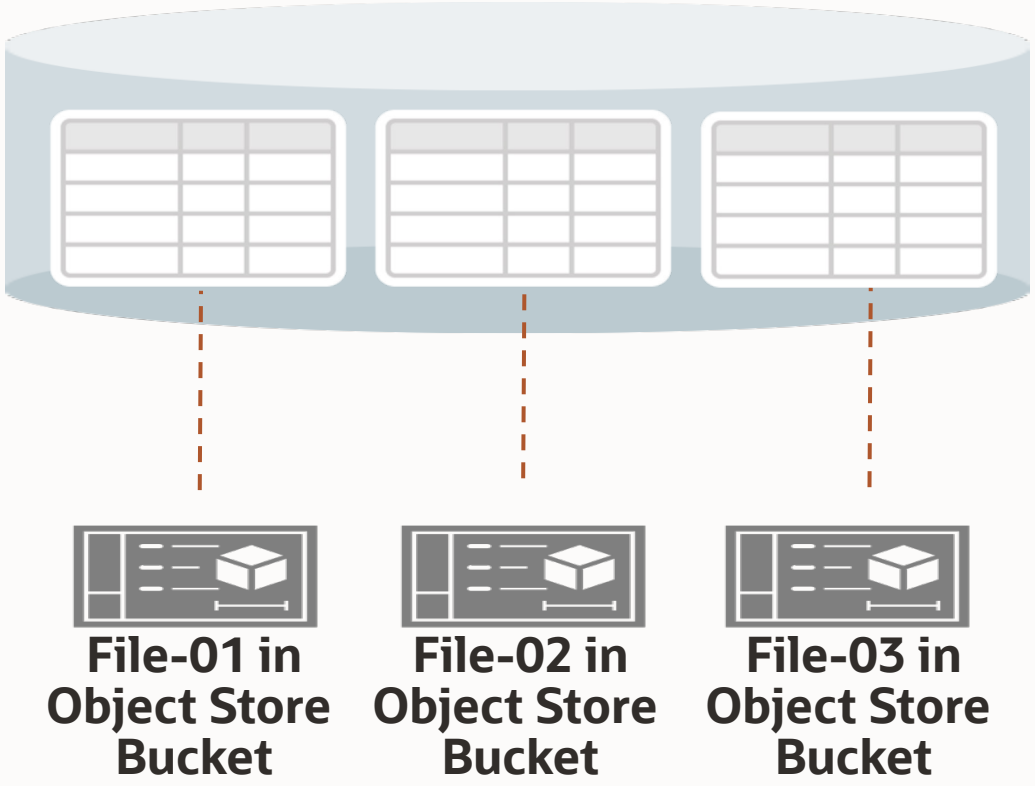
```
SELECT COUNT(*) FROM SALES@REDSHIFT_LINK;

COUNT(*)
-----------
172456
```




Load Data Using DBMS_CLOUD



Load data using DBMS_CLOUD



- For data loading from files in the Cloud
- Store your object storage credentials
- Use the procedure DBMS_CLOUD.COPY_DATA to load data
- The source file in this example is channels.txt

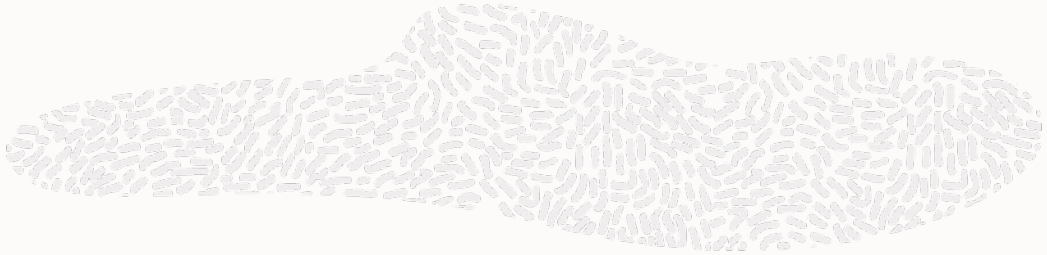
```
SET DEFINE OFF BEGIN
DBMS_CLOUD.CREATE_CREDENTIAL(
credential_name => 'DEF_CRED_NAME',
username => 'adwc_user@example.com',
password => 'password' ); END; /
```

Load data using DBMS_CLOUD

```
CREATE TABLE CHANNELS (channel_id CHAR(1), channel_desc VARCHAR2(20),
channel_class VARCHAR2(20) );

BEGIN DBMS_CLOUD.COPY_DATA( table_name =>'CHANNELS', credential_name
=>'DEF_CRED_NAME',
file_uri_list =>
'https://objectstorage.us-phoenix-1.oraclecloud.com/n/namespace-string/b/
bucketname/o/channels.txt', format => json_object('delimiter' value ',') );

END;
```

```
BEGIN DBMS_CLOUD.COPY_DATA( table_name =>'CHANNELS', credential_name
=>'DEF_CRED_NAME',
file_uri_list =>'https://objectstorage.us-phoenix-1.oraclecloud.com/n/namespace-
string/b/
bucketname/o/exp01.dmp,
https://objectstorage.us-phoenix-1.oraclecloud.com/n/namespace-string/b/
bucketname/o/exp02.dmp', format => json_object('type' value 'datapump') );

END;
```

Load data using DBMS_CLOUD



```
BEGIN DBMS_CLOUD.COPY_COLLECTION( collection_name => 'fruit', credential_name =>
'DEF_CRED_NAME', file_uri_list => 'https://objectstorage.us-ashburn-1.oraclecloud.com/n/
namespace-string/b/fruit_bucket/o/myCollection.json’,
format => JSON_OBJECT('recorddelimiter' value '''\n''') );

END;
```

```
BEGIN DBMS_CLOUD.COPY_COLLECTION( collection_name => 'fruit2', credential_name =>
'DEF_CRED_NAME', file_uri_list => 'https://objectstorage.us-ashburn-1.oraclecloud.com/n/
namespace-string/b/json/o/fruit_array.json’,
format => '{"recorddelimiter" : "0x''01''", "unpackarrays" : TRUE}' );

END;
```

Load data using DBMS_CLOUD

```
SELECT table_name, owner_name, type, status, start_time, update_time,
logfile_table, badfile_table FROM user_load_operations WHERE
type = 'COPY';

TABLE_NAME OWNER_NAME TYPE STATUS START_TIME UPDATE_TIME LOGFILE_TABLE BADFILE_TABLE
------------------------------------------------------------------------------
FRUIT ADMIN COPY COMPLETED 2020-04-23 22:27:37 2020-04-23 22:27:38 "" ""
FRUIT ADMIN COPY FAILED 2020-04-23 22:28:36 2020-04-23 22:28:37 COPY$2_LOG COPY$2_BAD
```

```
SELECT credential_name, username, comments FROM all_credentials;

CREDENTIAL_NAME USERNAME COMMENTS
------------------------------------------------- -------------------
ADB_TOKEN user_name@example.com {"comments":"Created via
DBMS_CLOUD.create_credential"}

DEF_CRED_NAME user_name@example.com {"comments":"Created via
DBMS_CLOUD.create_credential"}
```

ADB now supports SQL access to tenancy details



How to get the tenancy details for your instance

When you file a service request for Autonomous Database, you need to provide the tenancy details for your instance. Tenancy details for the instance are available on the Oracle Cloud Infrastructure console.

However, if you are connected to the database, you can now obtain these details by querying the **CLOUD_IDENTITY** column of the **V$PDBS view**. For example:

```
SELECT cloud_identity FROM v$pdbs;
```

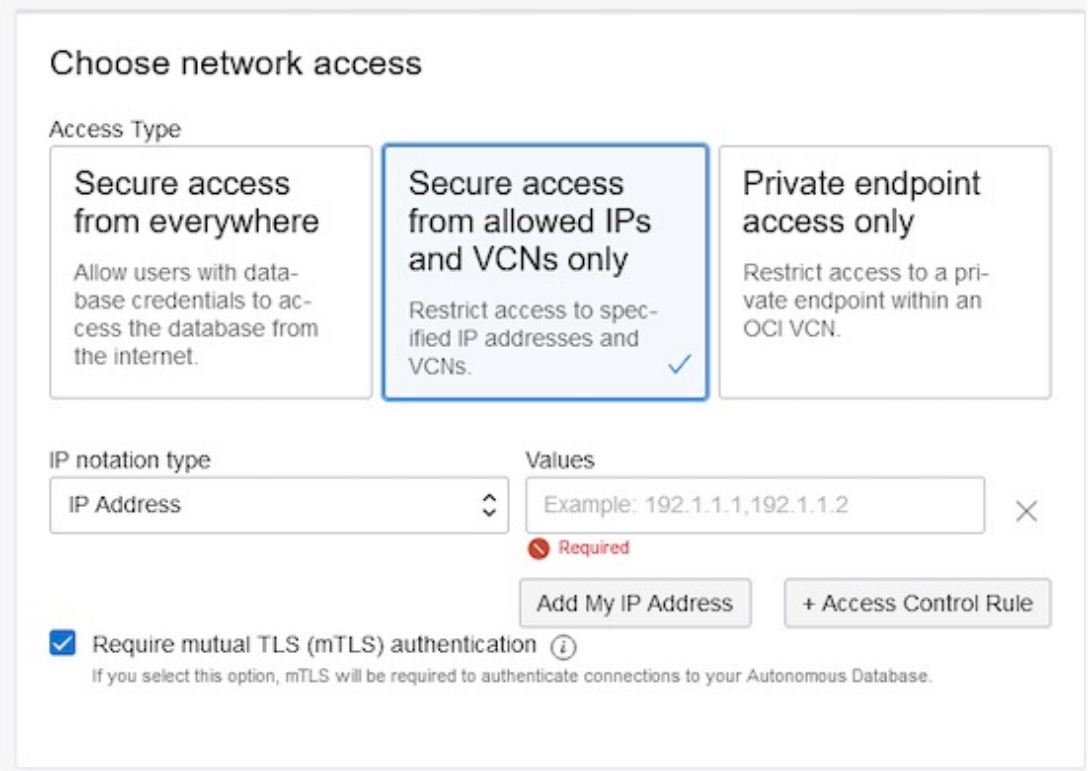
...will generate something similar to the following:

```
{"DATABASE_NAME" : "DBxxxxxxxxxxxx",
"REGION" : "us-phoenix-1",
"TENANT_OCID" :
"OCID1.TENANCY.REGION1..ID1",
"DATABASE_OCID" :
"OCID1.AUTONOMOUSDATABASE.OC1.SEA.ID2
",
"COMPARTMENT_OCID" :
"ocid1.tenancy.region1..ID3"}
```

IP Address ACL's



UI Button To Quickly Add Your IP Address To ACLs



Makes it easier to setup ACLs where you need to add the IP address of the client to your Network Access Control List (ACL)

The button "Add My IP Address" adds your current IP address to the ACL entry

Removes the need to manually look for your client IP via a 3rd party website or app anymore (e.g. Google, whatsmyip.com etc)

Available in both the Create Autonomous Database flow for new ADB-S instances to be provisioned and the Update Network Access flow for existing ADB-S instances

Multicloud with OCI and Azure



Multicloud with OCI and Azure

Low-latency connectivity between Microsoft Azure and OCI

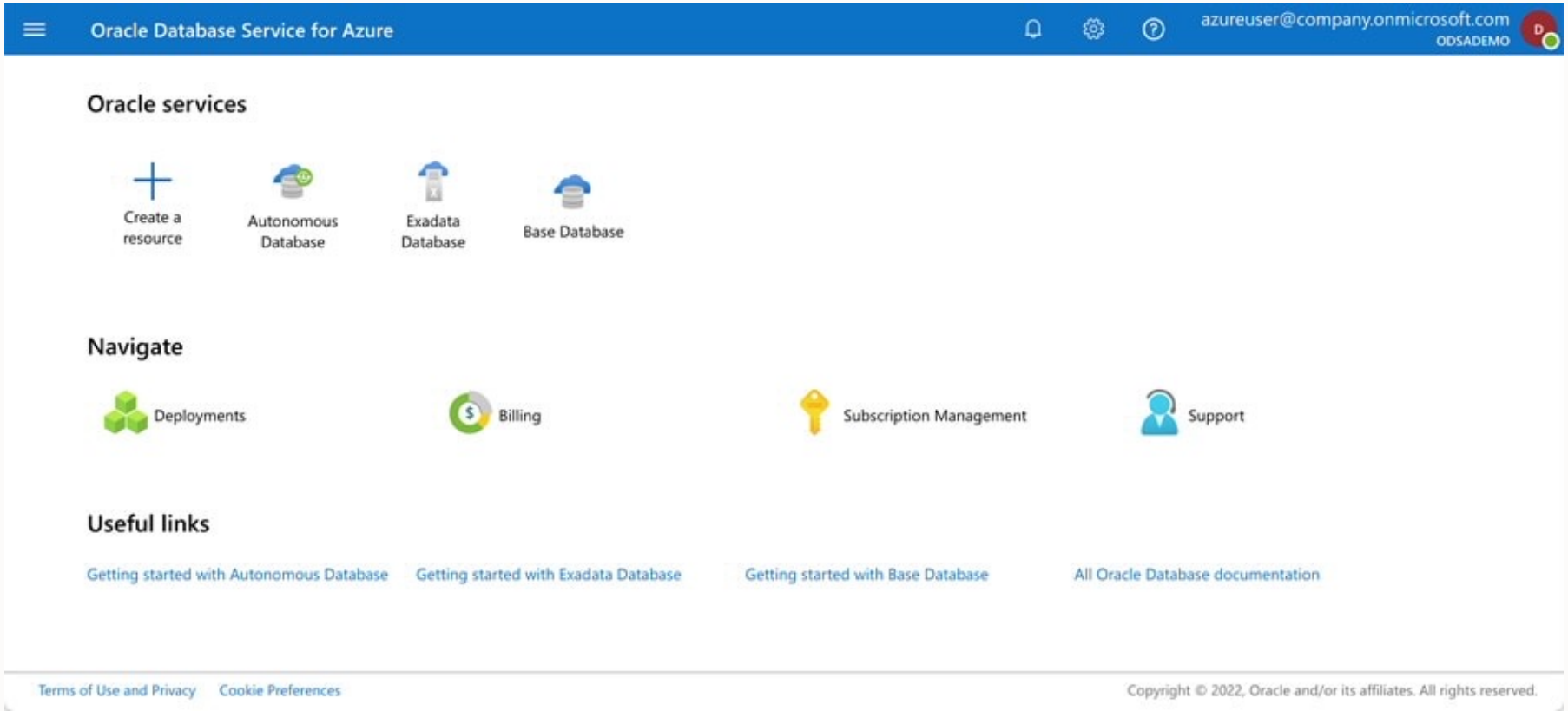
Deploys Oracle Database on OCI, and provides metrics on Azure

Combine the full Azure catalog of AI and application services with OCI's most powerful database services

No charges for Oracle Interconnect for Microsoft Azure ports or data ingress/egress over the Interconnect

Billing normally for consumption of Oracle Database services, such as Autonomous Database

Oracle Database Service for Microsoft Azure (ODSA)



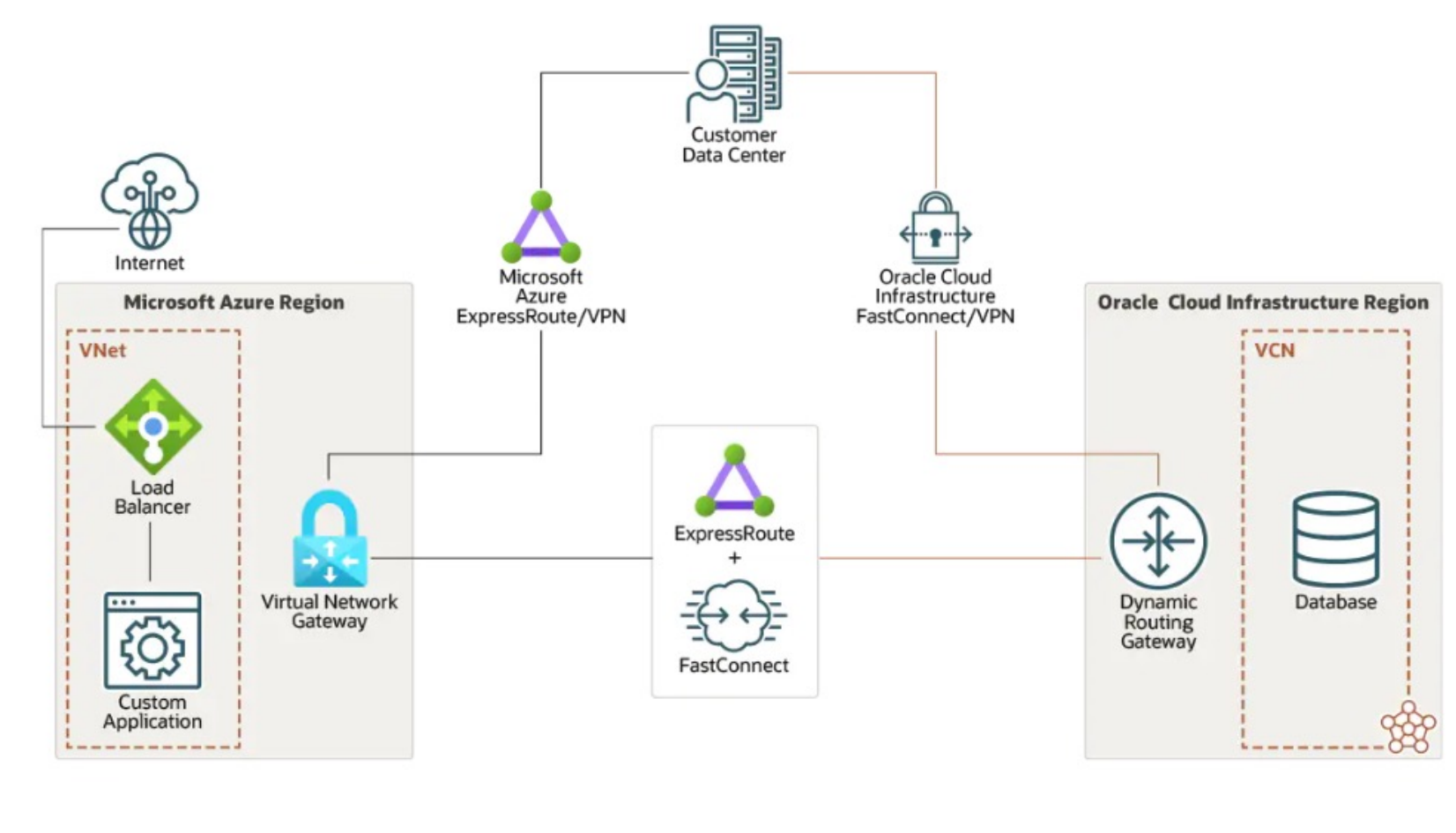
Automatically configures everything required to link the two cloud environments

Federates Azure active directory identities

Azure like UI & API experience for provisioning and managing Oracle database services on OCI

Sends metrics, logs, and events for the OCI databases to Azure tooling for unified telemetry and monitoring

Collaborative support model



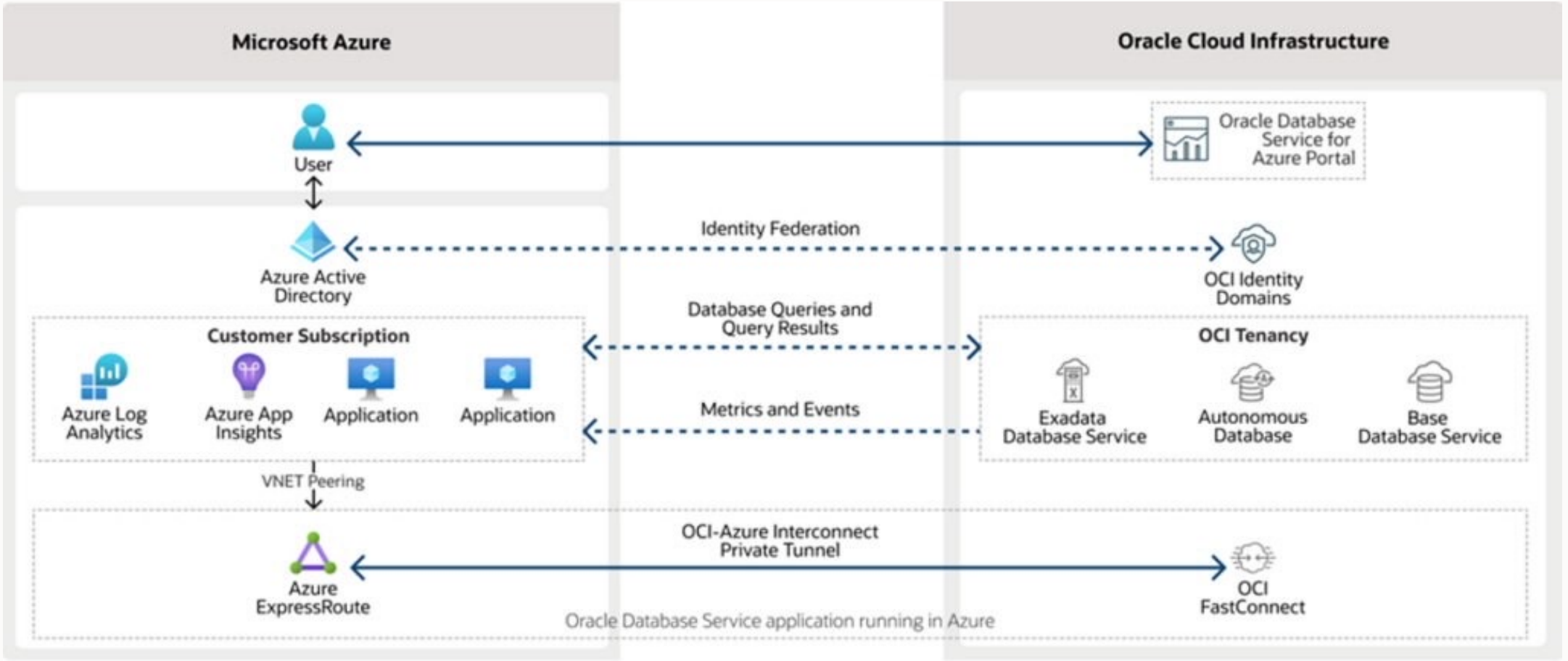
Direct connection between cloud vendors

<2ms latency for traffic between OCI and Microsoft Azure

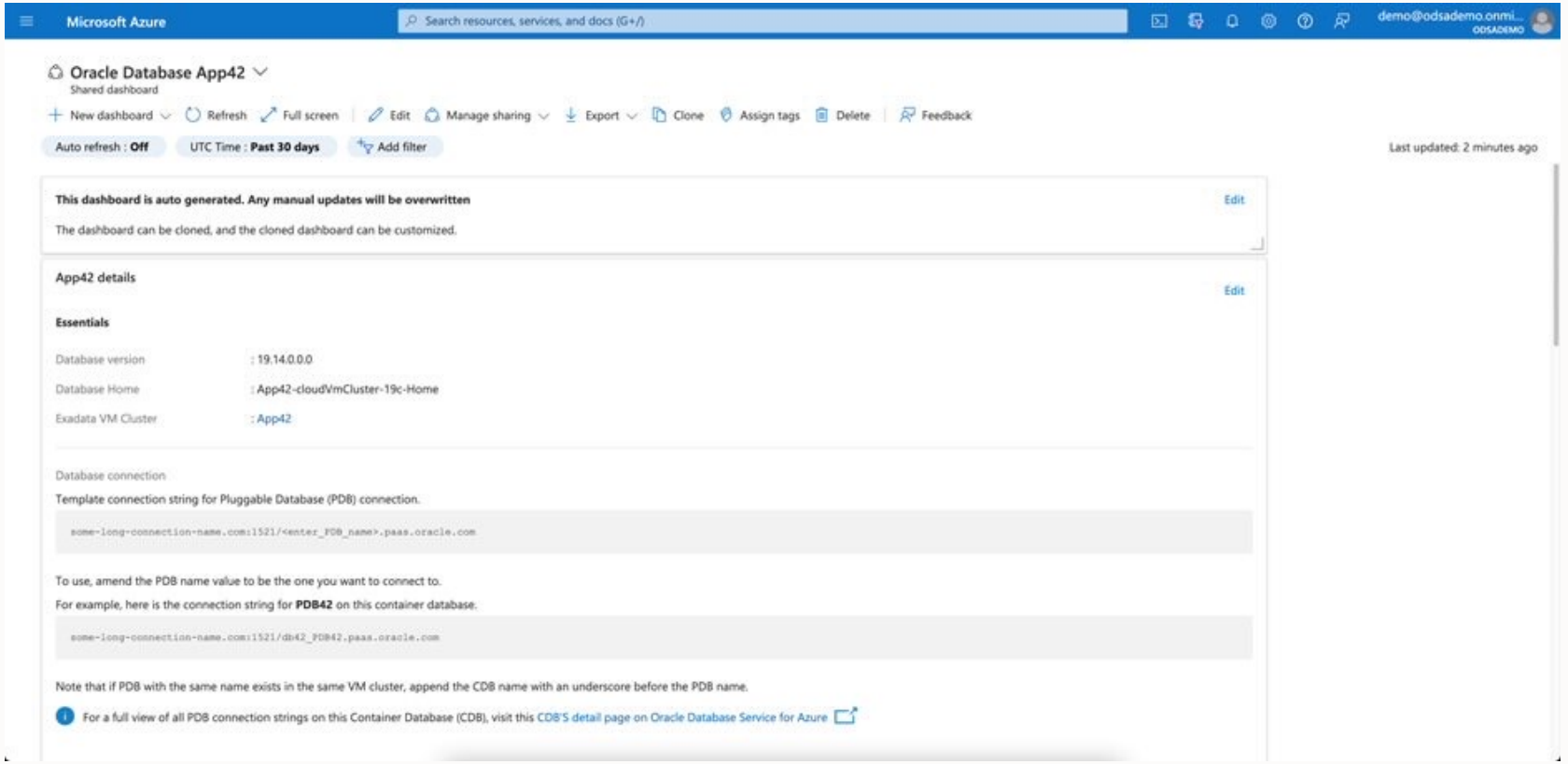
Pricing is based solely on port capacities for OCI FastConnect and Azure ExpressRoute Local Circuit

No charges for inbound or outbound bandwidth consumed

Architecture



Azure tools integration

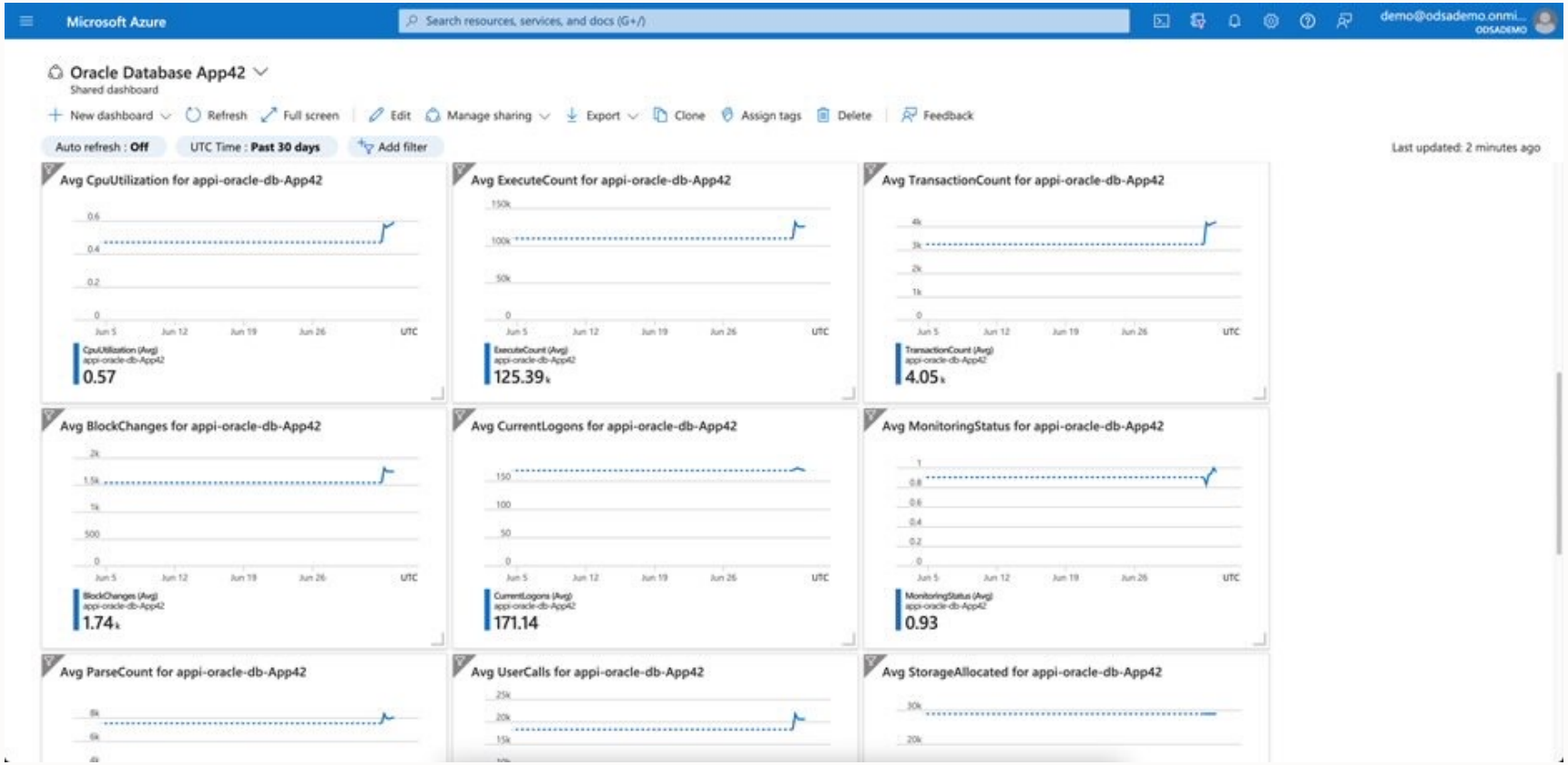


Delivers OCI database metrics, events, and logs to tools such as Azure Application Insights, Azure Event Grid, and Azure Log Analytics

Enables Azure users to view OCI databases alongside the rest of your Azure data, for unified telemetry and monitoring

Also creates a custom dashboard that provides Azure developers with Oracle database resource details, and connection strings for their applications

Custom dashboard



Displays graphs for each of the standard Oracle database metrics for the resource

Give developers and administrators a quick view of all metrics in one place

Support for wider range of database character sets

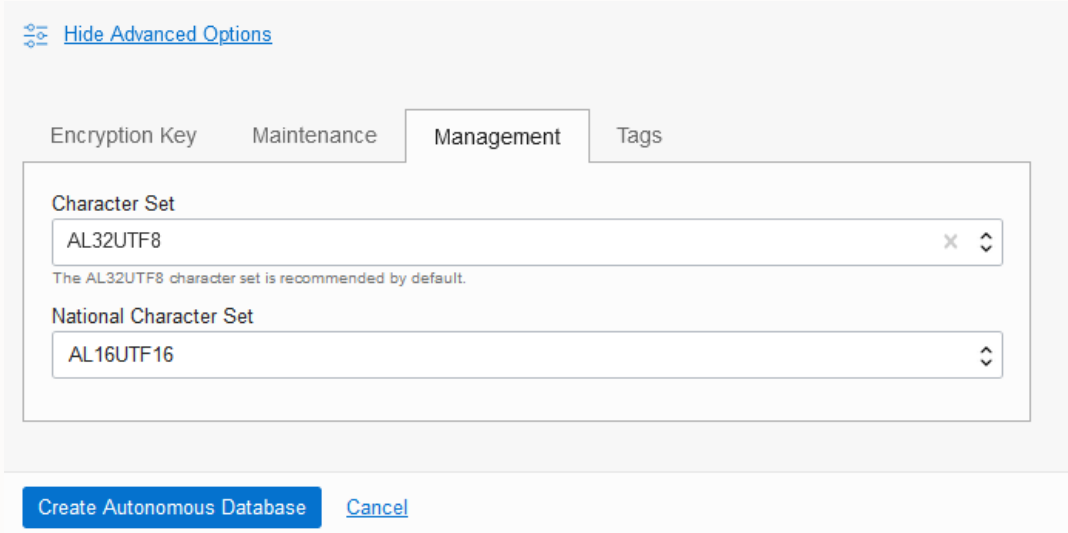


Support for wider range of database character sets

Migrating on-premises database with a non-Unicode character set can be complex

Autonomous Database **lets you choose a character set when you provision an instance**

1. Create an Autonomous Database following the provisioning steps.
2. On the **Create Autonomous Database** page expand **Show Advanced Options**.
3. In the advanced options area, select the **Management** tab
   - In the **Character Set** field, use the selector to choose a character set.
   - In the **National Character Set** field, use the selector to choose a national character set.



To make selecting a value easier, when you type in the text area this filters the list. For example, if you type JA you see only the options containing JA, including: JA16EUC, JA16EUCTILDE, JA16SJIS, JA16JA16SJISTILDE, and JA16VMS.

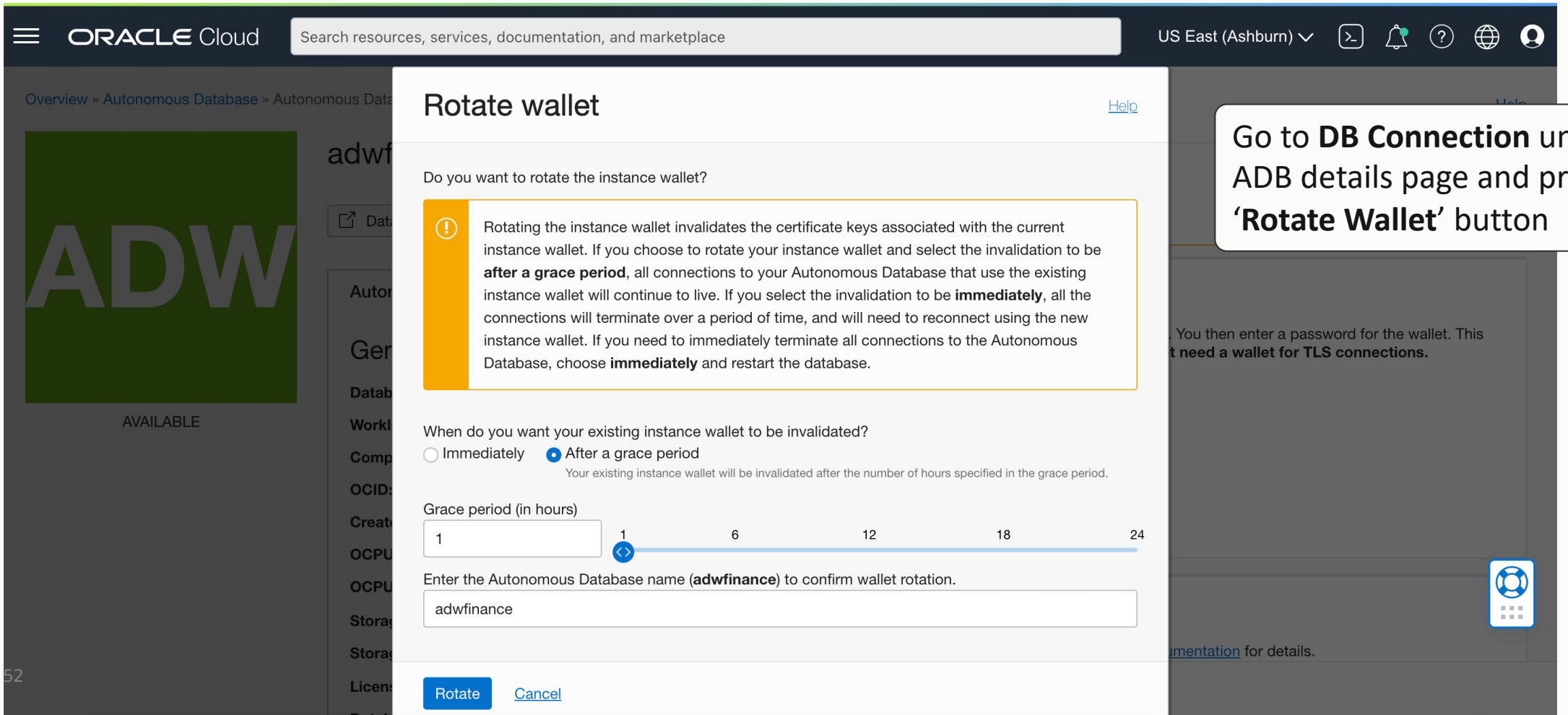
4. Click **Create Autonomous Database** to provision the Autonomous Database.

Gradual Wallet Rotation



Gradual Wallet Rotation

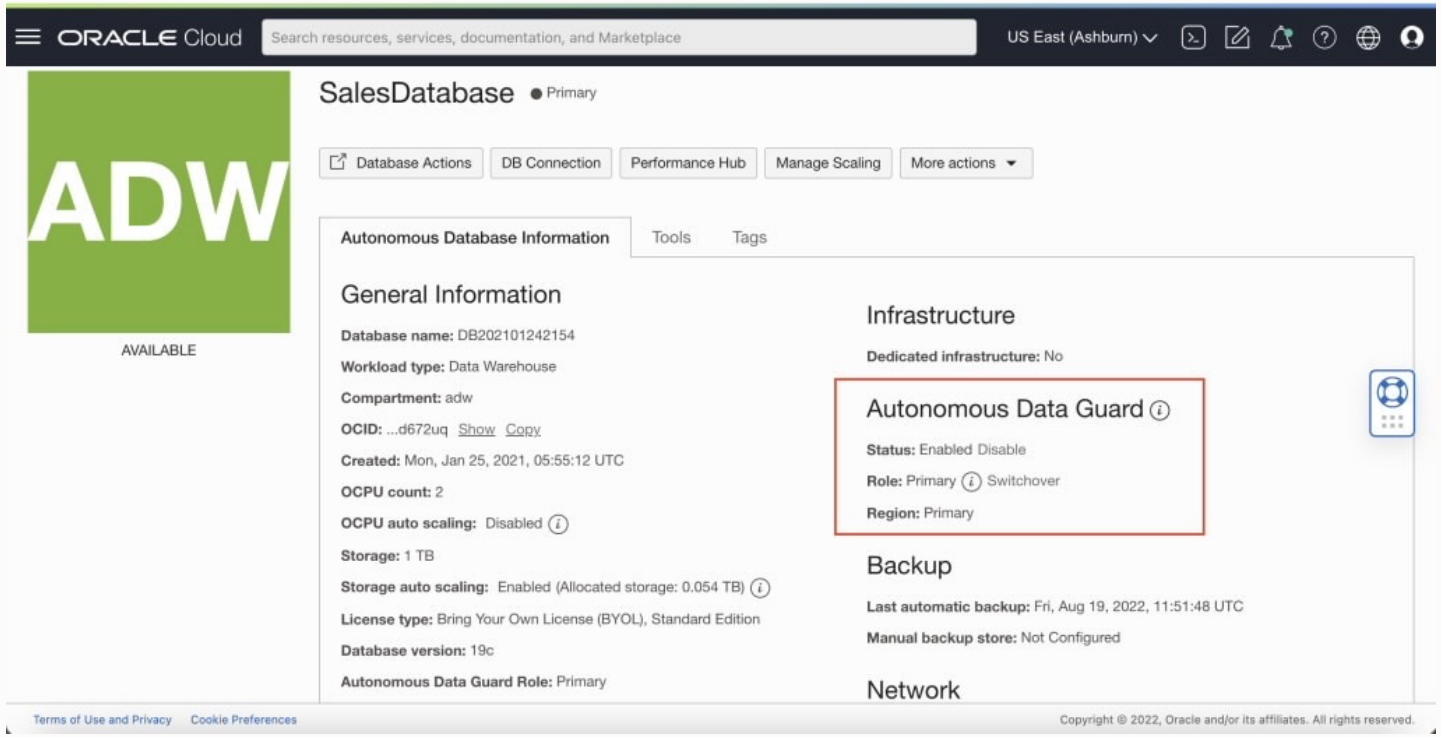
- Immediately invalidate your wallet if you think it's been compromised
- Periodically rotate your wallet every 3 or 6 months for compliance reasons



Go to **DB Connection** under ADB details page and press '**Rotate Wallet**' button

99.995% = Maximum of 2 mins 11 secs of Downtime per Month

No Hidden Exclusion Clauses!



Covers:

- Unplanned downtime: hardware / network outages
- Planned download: database maintenance tasks & patching

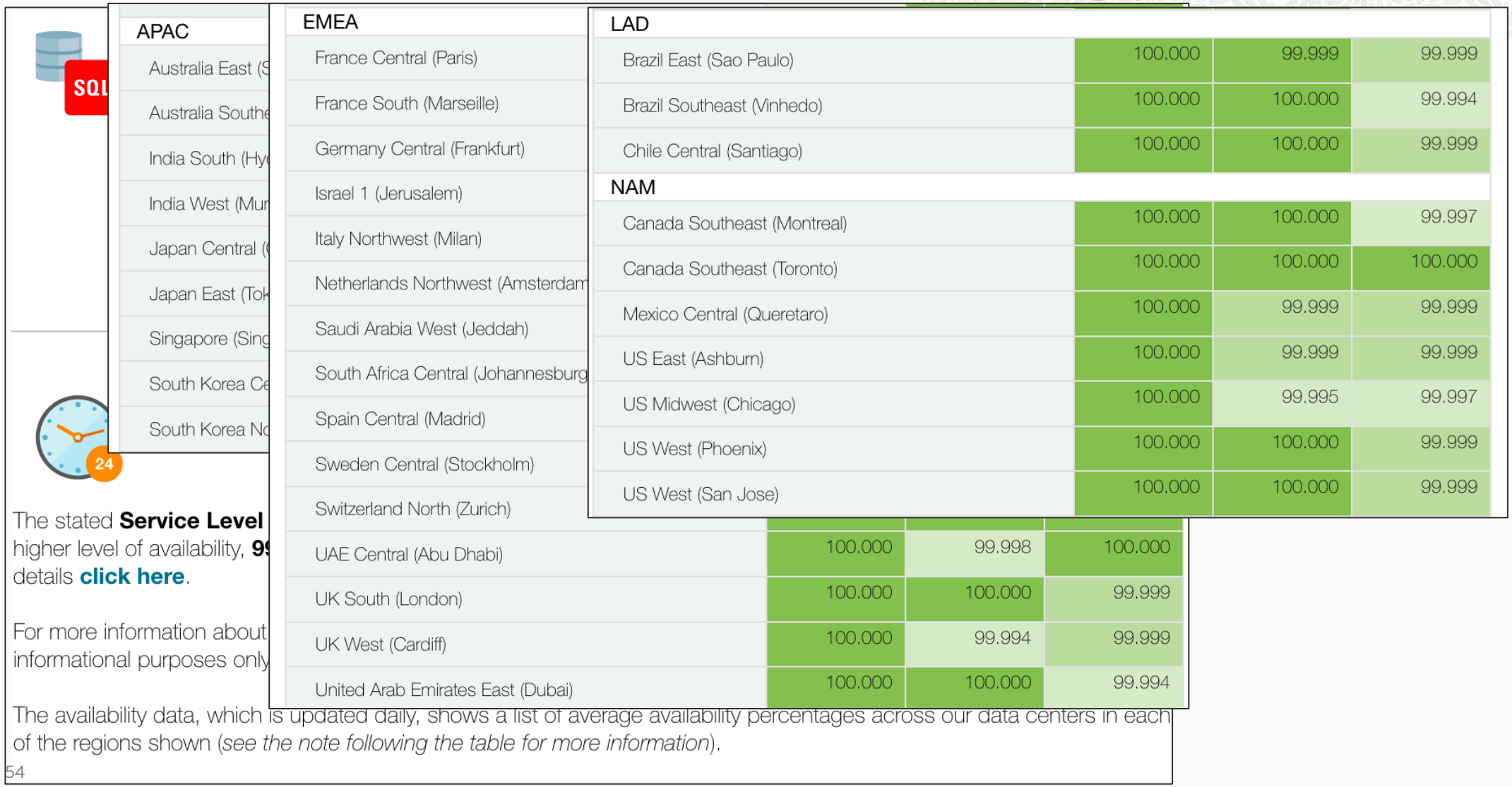
Applies to both Shared & Dedicated infrastructure

Configuration	SLA
With ADG standby enabled	99.995%
Without ADG standby enabled	99.95%

https://blogs.oracle.com/datawarehousing/post/autonomous-database-updated-sla

Global Database Metrics for Oracle Autonomous Database

https://www.oracle.com/database/technologies/datawarehouse-bigdata/adb-availability-metrics.html



APAC			
Australia East (S			
Australia Southe			
India South (Hy			
India West (Mur			
Japan Central (			
Japan East (Tok			
Singapore (Sing			
South Korea Ce			
South Korea No			
EMEA			
France Central (Paris)			
France South (Marseille)			
Germany Central (Frankfurt)			
Israel 1 (Jerusalem)			
Italy Northwest (Milan)			
Netherlands Northwest (Amsterdam			
Saudi Arabia West (Jeddah)			
South Africa Central (Johannesburg			
Spain Central (Madrid)			
Sweden Central (Stockholm)			
Switzerland North (Zurich)			
UAE Central (Abu Dhabi)	100.000	99.998	100.000
UK South (London)	100.000	100.000	99.999
UK West (Cardiff)	100.000	99.994	99.999
United Arab Emirates East (Dubai)	100.000	100.000	99.994
LAD			
Brazil East (Sao Paulo)	100.000	99.999	99.999
Brazil Southeast (Vinhedo)	100.000	100.000	99.994
Chile Central (Santiago)	100.000	100.000	99.999
NAM			
Canada Southeast (Montreal)	100.000	100.000	99.997
Canada Southeast (Toronto)	100.000	100.000	100.000
Mexico Central (Queretaro)	100.000	99.999	99.999
US East (Ashburn)	100.000	99.999	99.999
US Midwest (Chicago)	100.000	99.995	99.997
US West (Phoenix)	100.000	100.000	99.999
US West (San Jose)	100.000	100.000	99.999

The stated **Service Level**
higher level of availability, **9**
details **click here**.

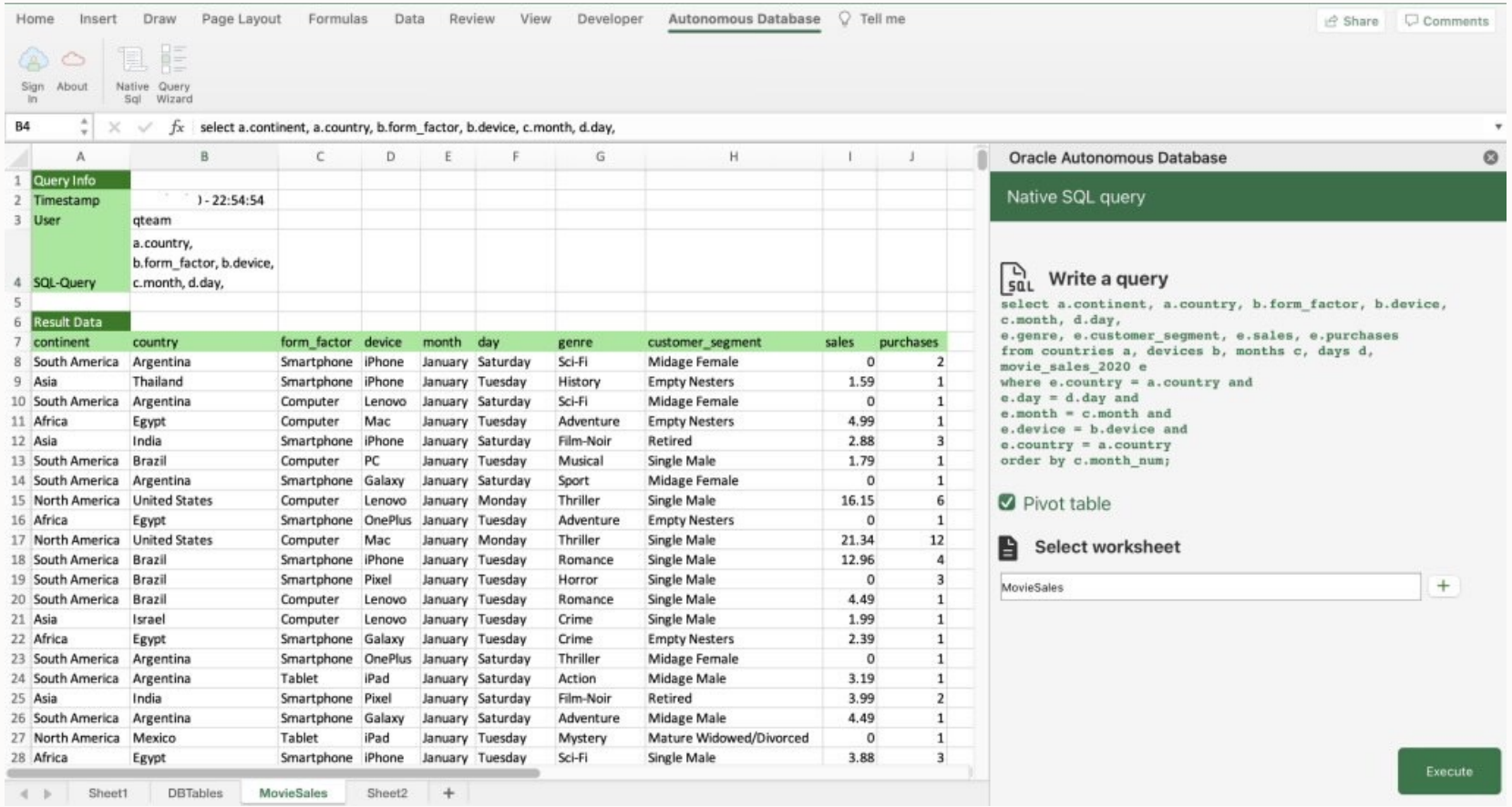
For more information about
informational purposes only

The availability data, which is updated daily, shows a list of average availability percentages across our data centers in each of the regions shown (*see the note following the table for more information*).

Autonomous Database Plugin for **Microsoft Excel**



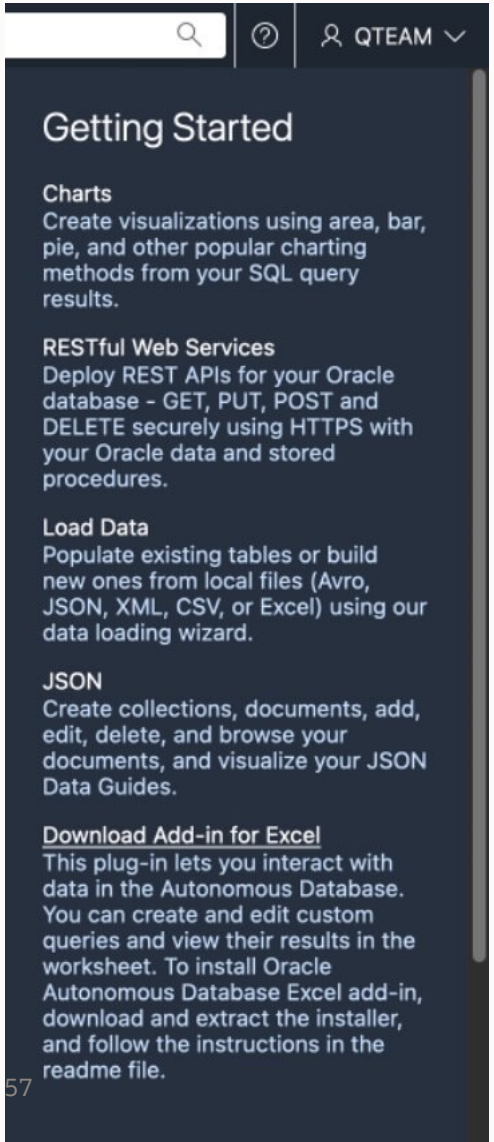
Using the Excel Add-in to Query Autonomous Database



Excel Add-in allows you query data in the Autonomous Database directly from Excel

Run native SQL queries and use a wizard to query Analytic Views created by the Data Analysis tool

Download the Add-in

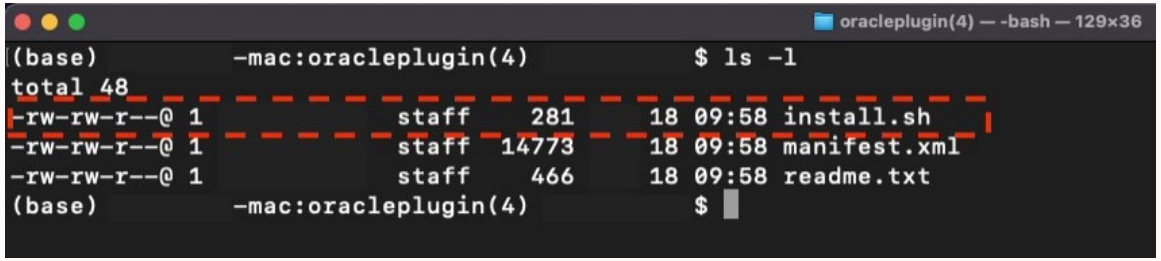


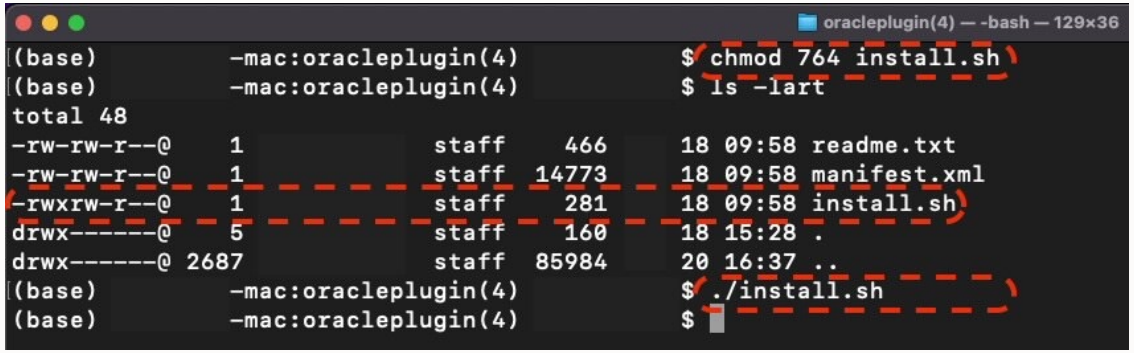
1. Login to the web UI for the DB Actions page.
2. On the right-side navigation menu, click on the link to “Download Add-in for Excel.”

Installation on macOS

3. Unzip the downloaded zip file
4. Open a terminal window and navigate to the unzipped folder
5. Ensure that Excel is not running.
6. The install.sh file does not have to execute permissions

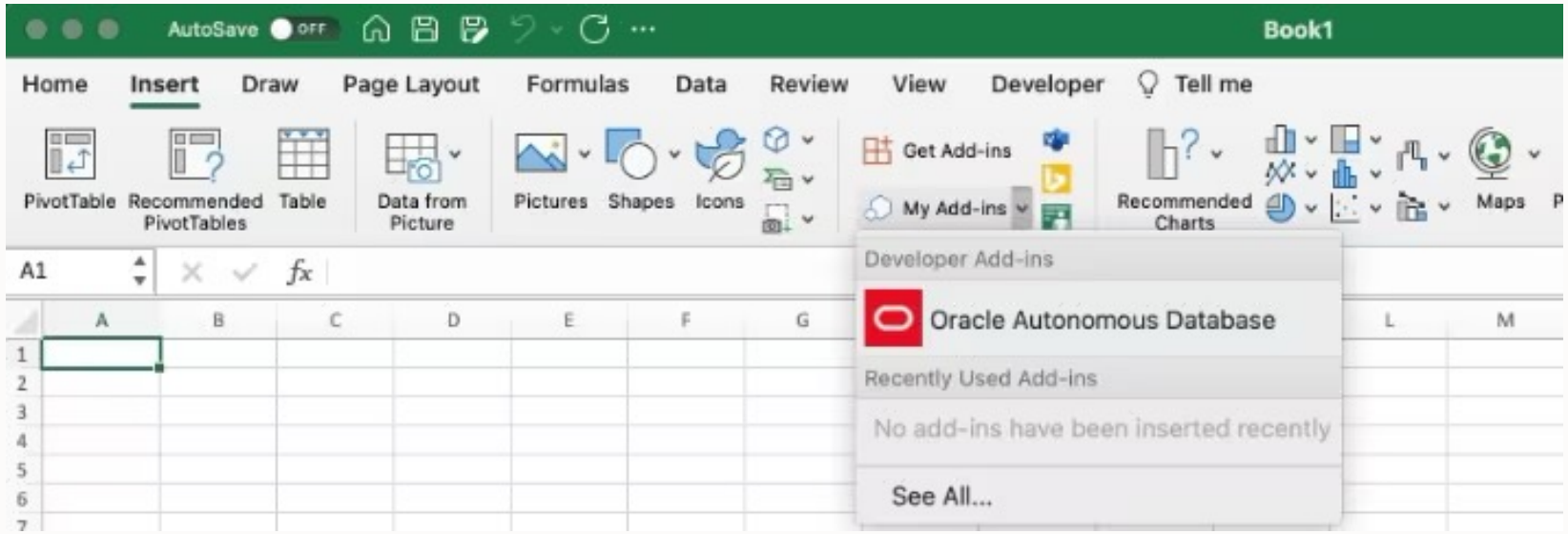
```
chmod 764 install.sh
./install.sh
```





Installation on macOS

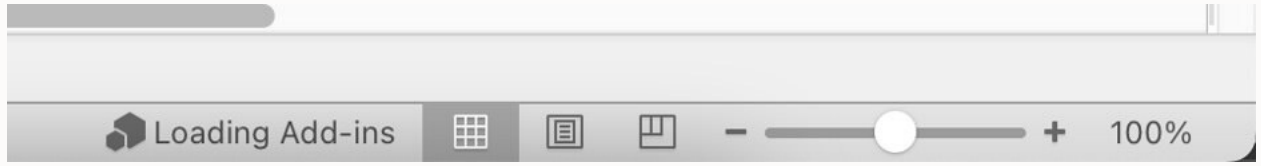
7. Launch Excel
8. On the Insert tab on the ribbon, click the down arrow on the Add-ins / My Add-ins option:



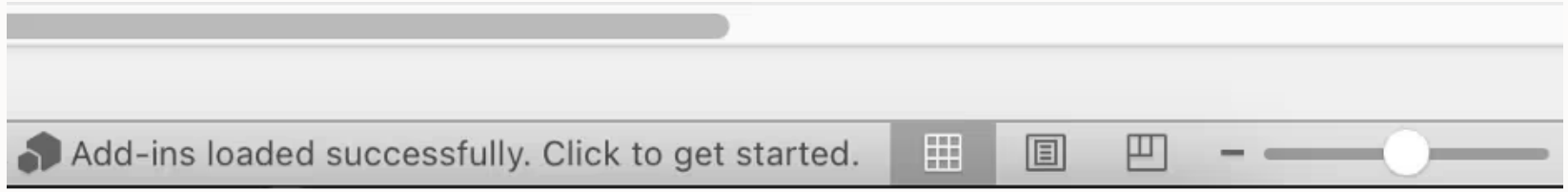
9. Under Developer Add-ins, you will see the Oracle Autonomous Database Add-in.
10. Click to select this Add-in

Installation on macOS

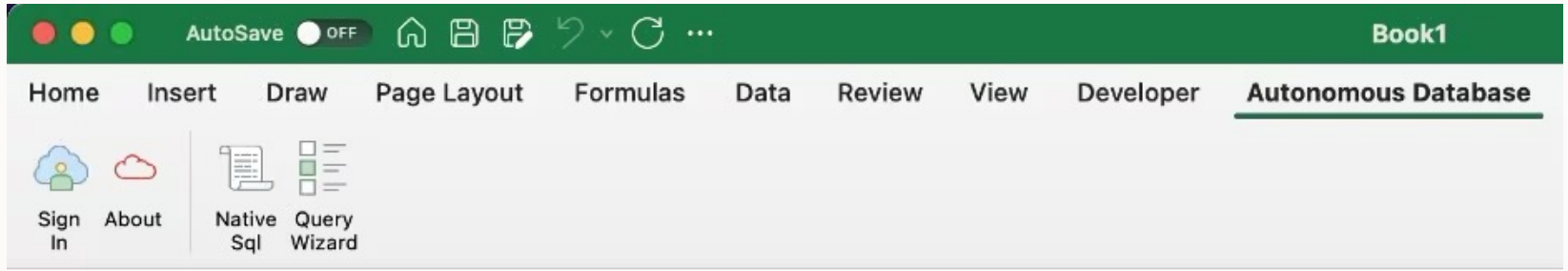
At the bottom, you see a notification about the Add-in being loaded:



After the Add-in is loaded successfully, you will see the following message:

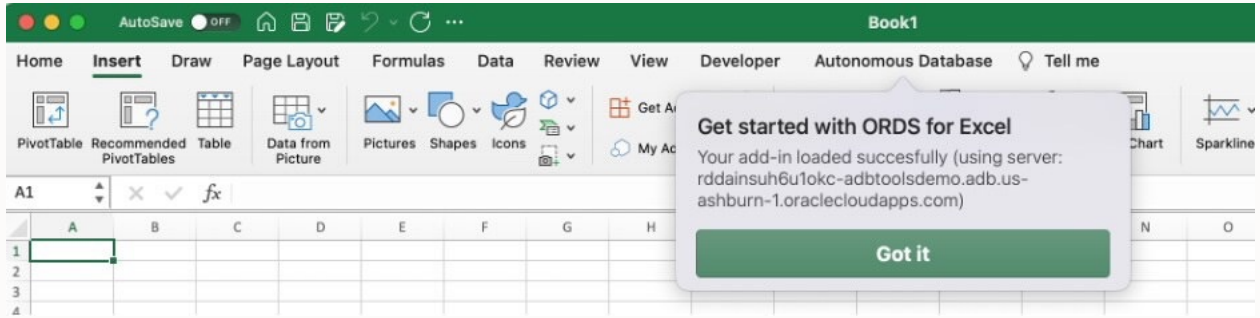


Also, a new ribbon item, “Autonomous Database.”



Installation on macOS

11. Close and Quit Excel
12. Launch Excel and insert the Add-in again. (You must perform these steps 6 thru 9 mentioned above every time you launch Excel.)



You are now ready to connect to the Autonomous Database, run native SQL, and use the Analytic View Query wizard.

Run Native SQL for analysis using Excel Pivot tables

Launch the Native Sql panel by clicking the button in the Autonomous Database ribbon.

Example SQL query

```
select a.continent, a.country, b.form_factor, b.device, c.month, d.day,
e.genre, e.customer_segment, e.sales, e.purchases
from countries a, devices b, months c, days d, movie_sales_2020 e
where e.country = a.country and
e.day = d.day and
e.month = c.month and
e.device = b.device and
e.country = a.country
order by c.month_num;
```

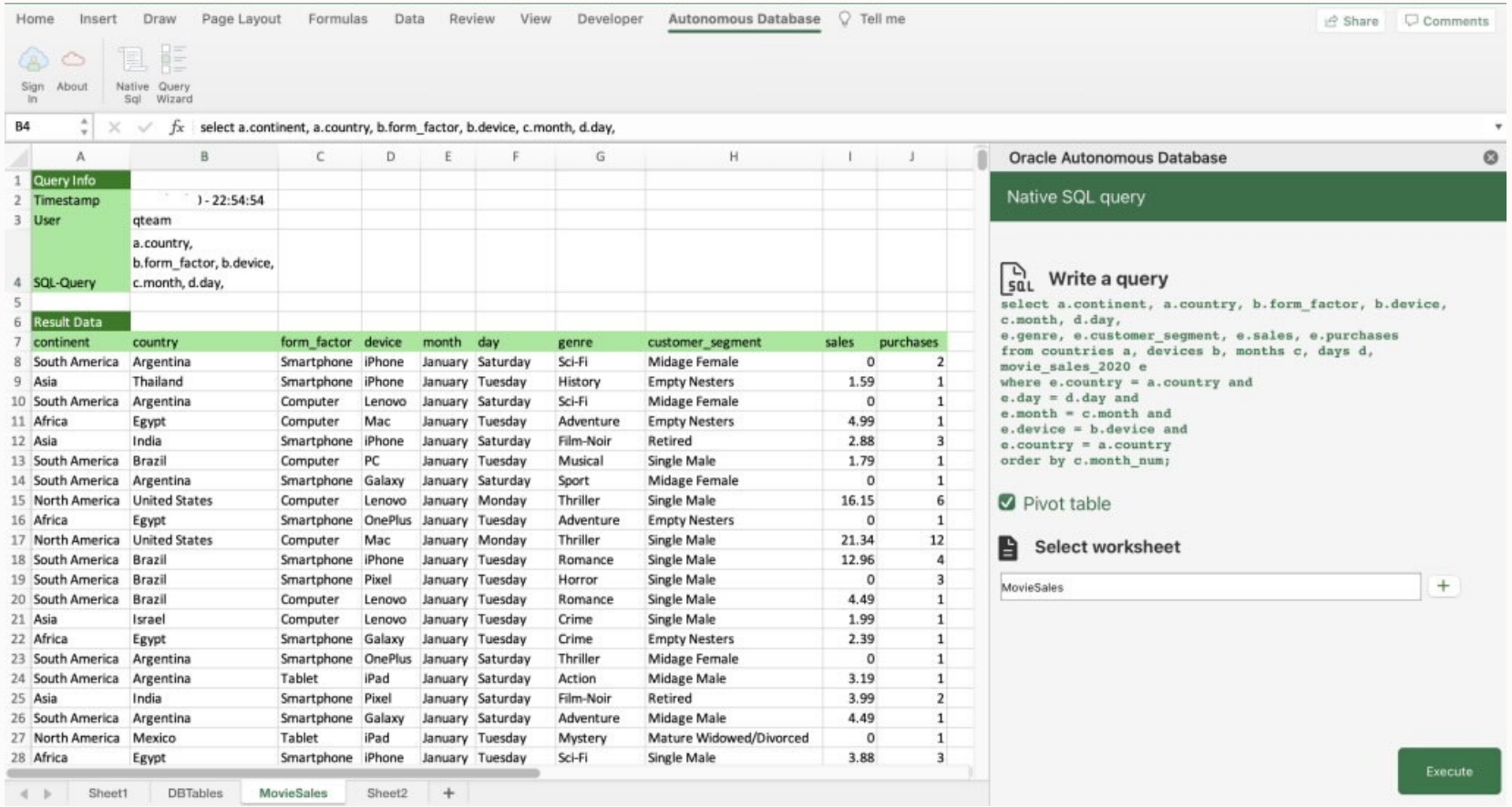
Add the above query in the text box under the Write a query label on the right-side panel.

Check the Pivot table checkbox.

Under Select worksheet, click the “+” icon and provide MovieSales as the name.

Click the check button then Execute

Run Native SQL for analysis using Excel Pivot tables

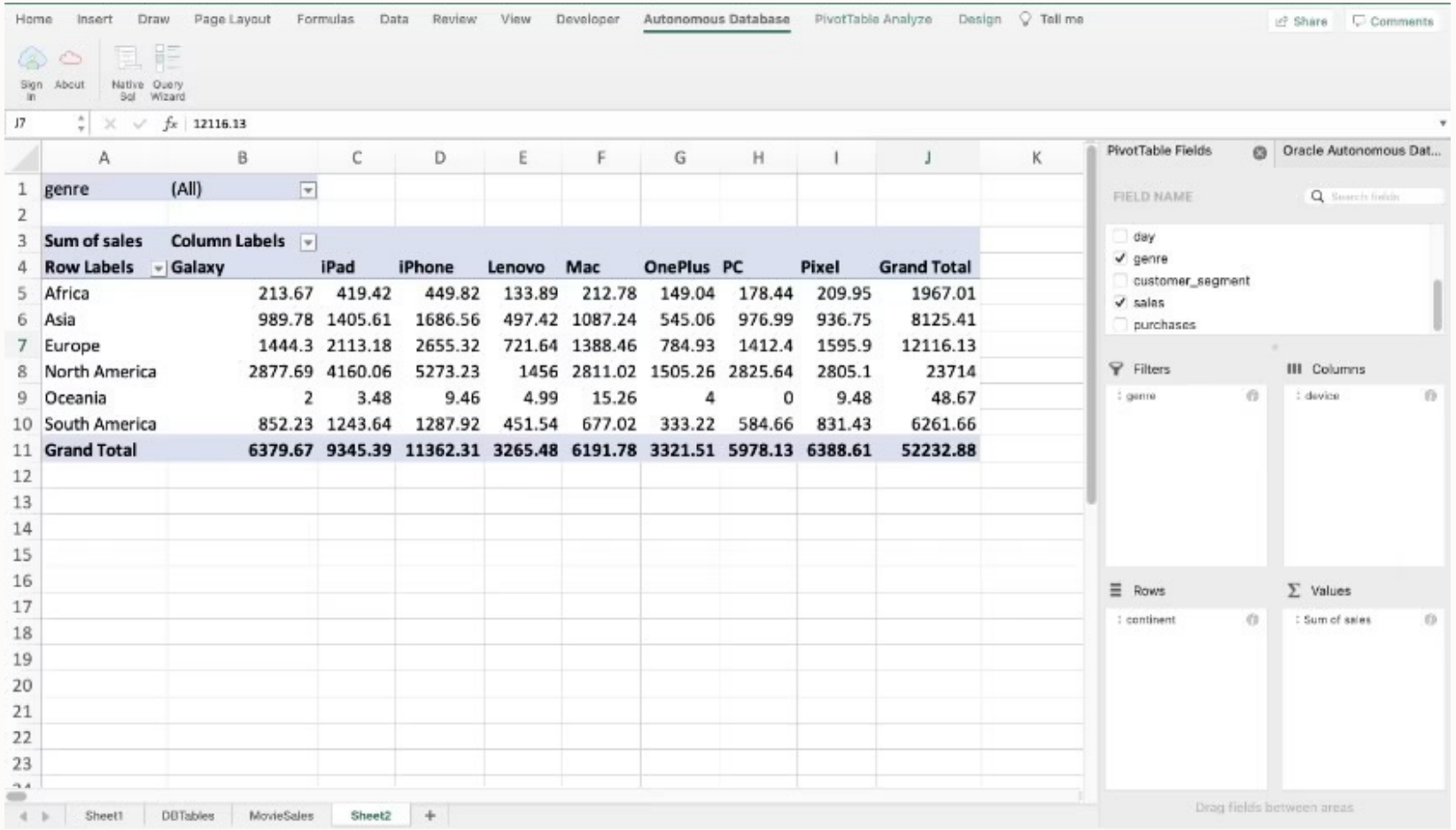


Two new tabs are created, viz. MovieSales and Sheet2 (Sheet number might vary in your case)

On Sheet2, an Excel pivot table is created with the data fetched from the Autonomous Database.

Run Native SQL for analysis using Excel Pivot tables

Setup the Pivot table options as shown on the screen below:



The data for this pivot table is fetched from the MovieSales worksheet.

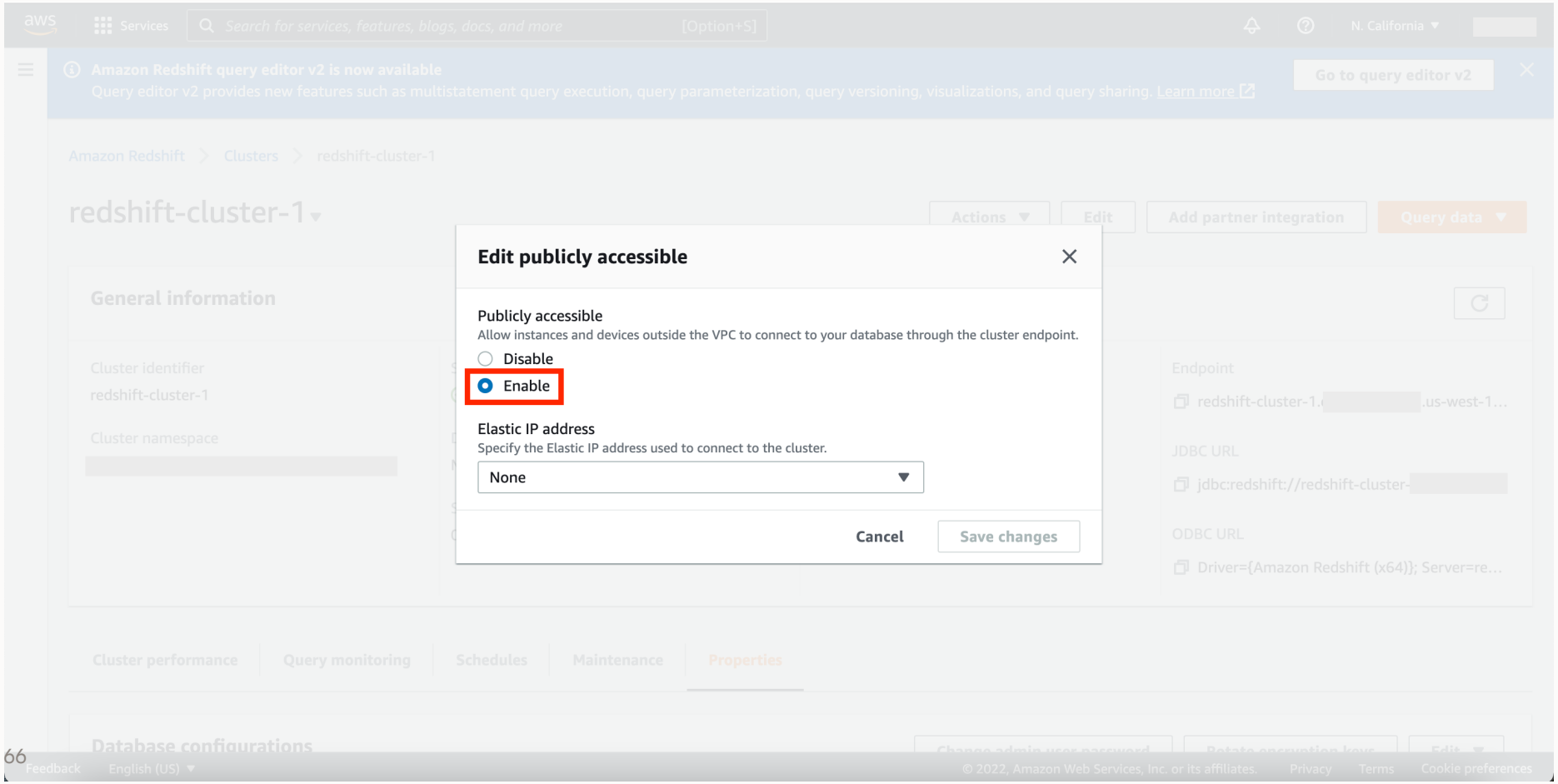
Now you can use the native Excel capabilities to analyze data.

Access Amazon Redshift, Snowflake and Other Non-Oracle Databases from Your Autonomous Database



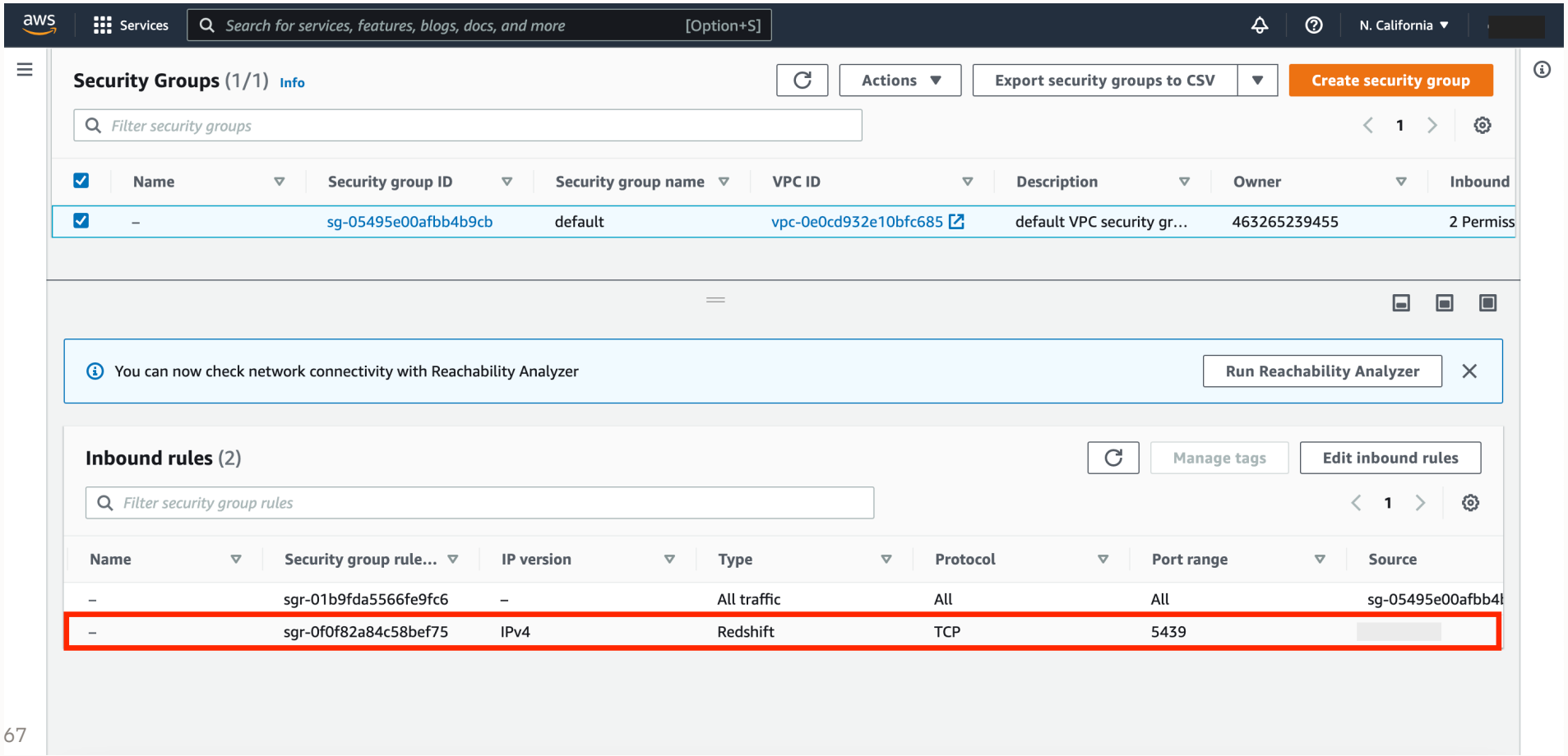
Step-1: Make Sure Redshift is Configured to Allow Public Access

Under “Actions” menu of your Redshift cluster on AWS console, we need to select “Modify publicly accessible setting” to make sure our Redshift cluster is publicly accessible:



Step-1: Make Sure Redshift is Configured to Allow Public Access

Navigate to the VPC security group that is assigned to our Redshift cluster and create an inbound rule to port 5439 from the source IP or CIDR range of our choice:



Step-2: Create a Database Link to your Redshift Instance

Create a credential object with the credentials (username and password) of the target database:

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL(
    credential_name => 'REDSHIFT_CRED',
    username => 'awsadmin',
    password => '************');
END;
/

PL/SQL procedure successfully completed.
```

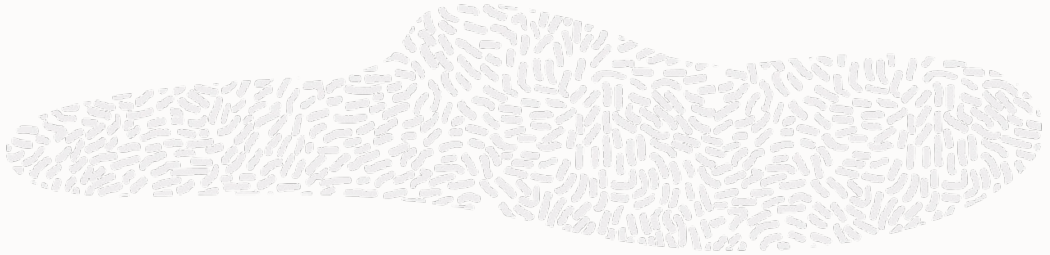
Step-2: Create a Database Link to your Redshift Instance

Create a database link to our Redshift instance.

Nearly identical to any other database link creation except for the gateway_params parameter for which we are passing 'awsredshift' as our database type:

```
BEGIN
     DBMS_CLOUD_ADMIN.CREATE_DATABASE_LINK(
          db_link_name => 'REDSHIFT_LINK',
          hostname => 'redshift-cluster-1.******.us-west-
1.redshift.amazonaws.com',
          port => '5439',
          service_name => 'dev',
          credential_name => 'REDSHIFT_CRED',
          gateway_params => JSON_OBJECT('db_type' value 'awsredshift'),
          ssl_server_cert_dn => NULL);
END;
/

PL/SQL procedure successfully completed.
```

Step-3: Run a Query Over the Database Link



```
SELECT COUNT(*) FROM SALES@REDSHIFT_LINK;

COUNT(*)
-----------
172456
```

The evolution of data management and analytics



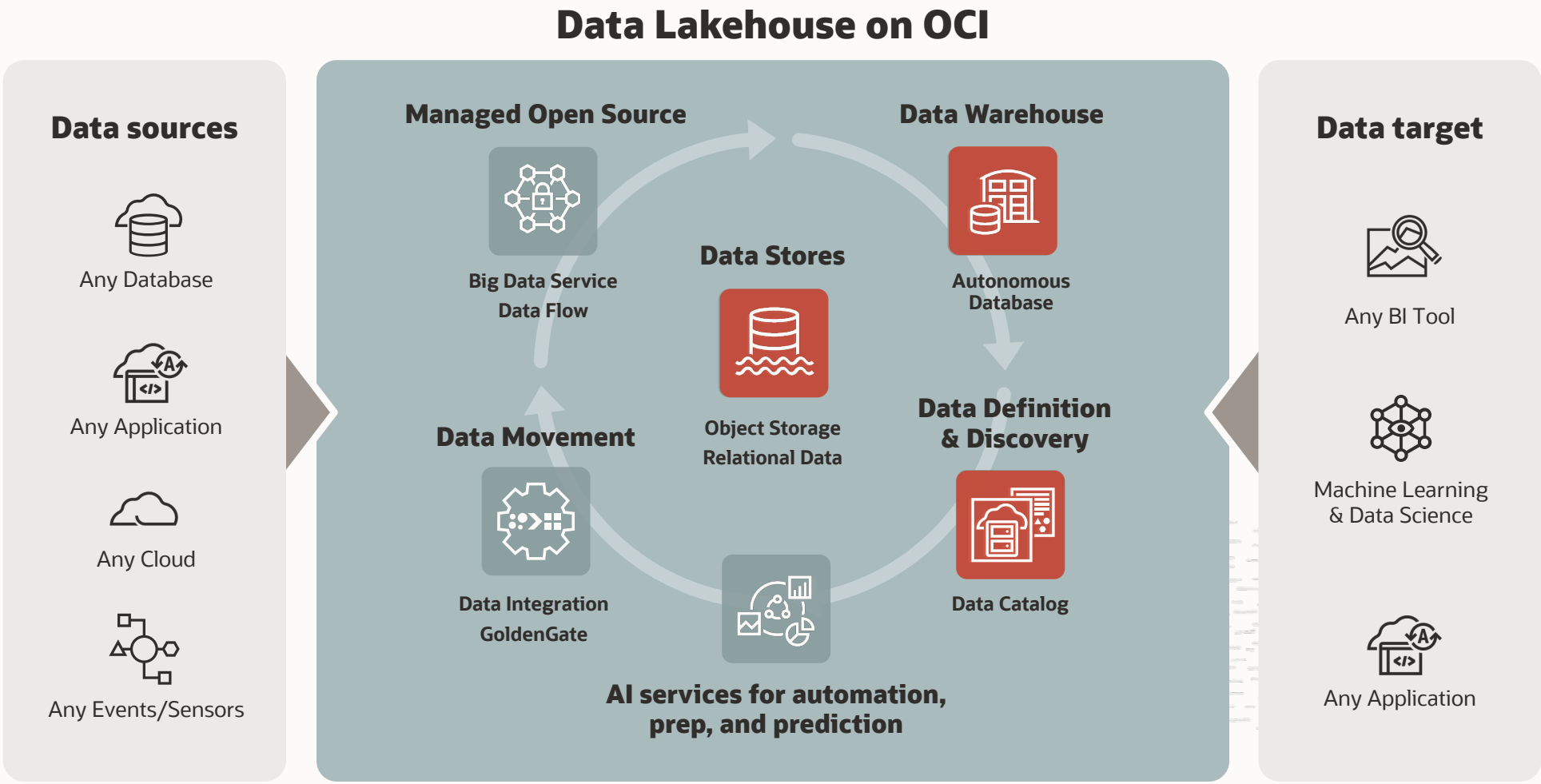






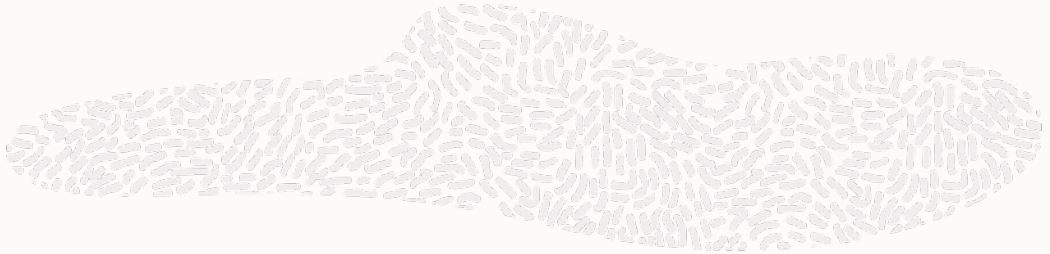
Data Lakehouses on OCI

Open & flexible: analyze any database, any application, from anywhere



Understand data in your Lake

Create External table for external data

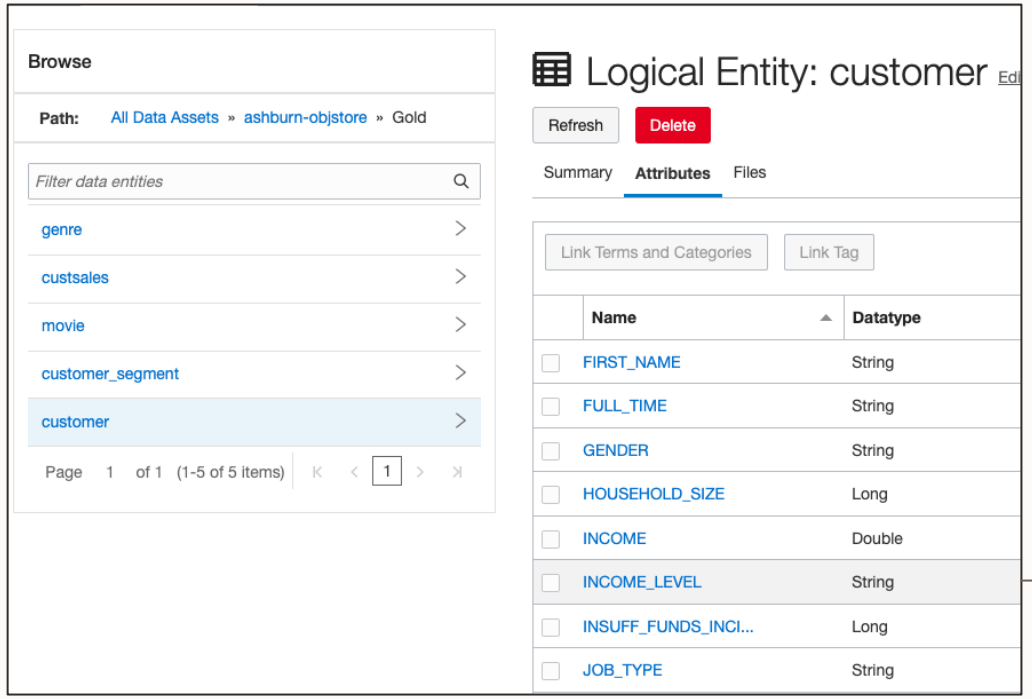


```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL(
    credential_name => 'DEF_CRED_NAME',
    username => 'adb_user@example.com',
    password => 'password' );
END;
/

BEGIN
   DBMS_CLOUD.CREATE_EXTERNAL_TABLE(
    table_name =>'sales_extended_ext',
    credential_name =>'DEF_CRED_NAME',
    file_uri_list =>'https://objectstorage.us-phoenix-
1.oraclecloud.com/n/namespace-string/b/bucketname/o/sales_extended.parquet',
    format =>  '{"type":"parquet", "schema": "first"}');
END;
/
```

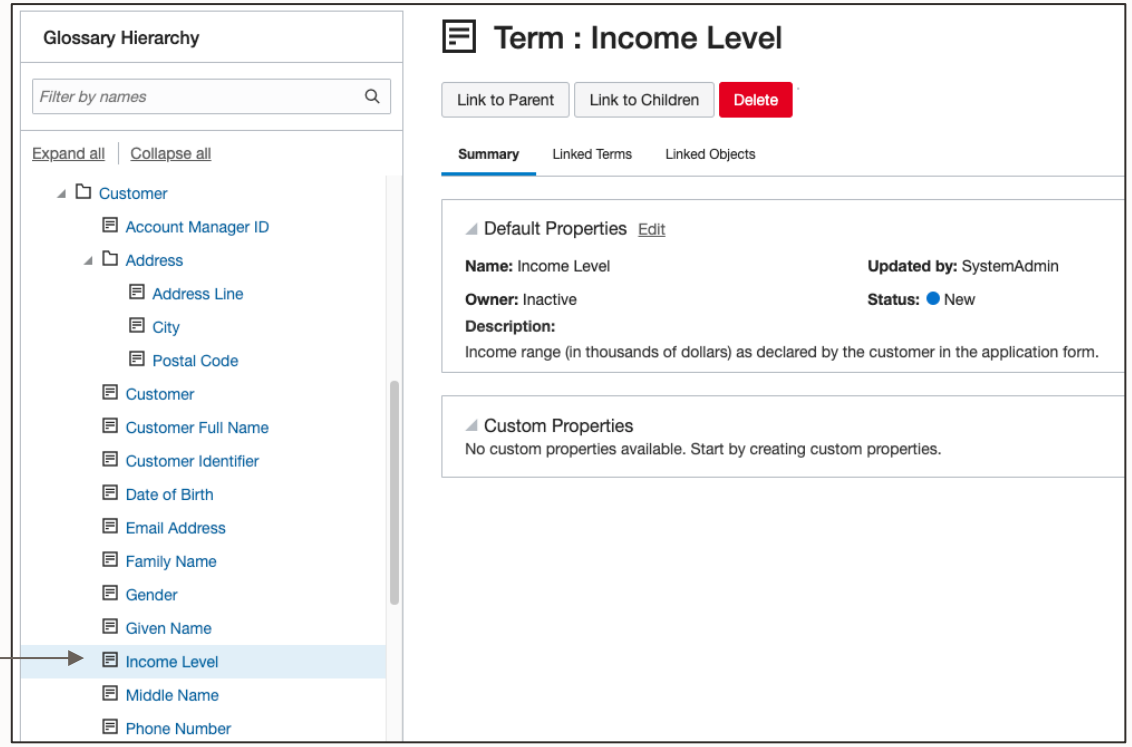
OCI Data Catalog

Evolve object storage files into meaningful business entities

Derive technical metadata automatically



Easily apply business context



Data Catalog integrates with Autonomous Database

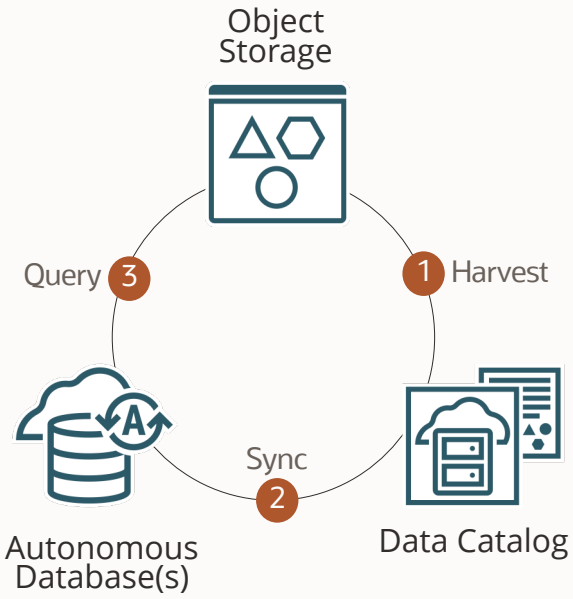
Autonomous Database both a Data Catalog source and consumer

Data Catalog is the source of truth for Object Store metadata

- Harvest object storage to derive schemas
- Manage business glossary, terms and tags
- Discover data using powerful search

Use Autonomous Database to discover and analyze data sets

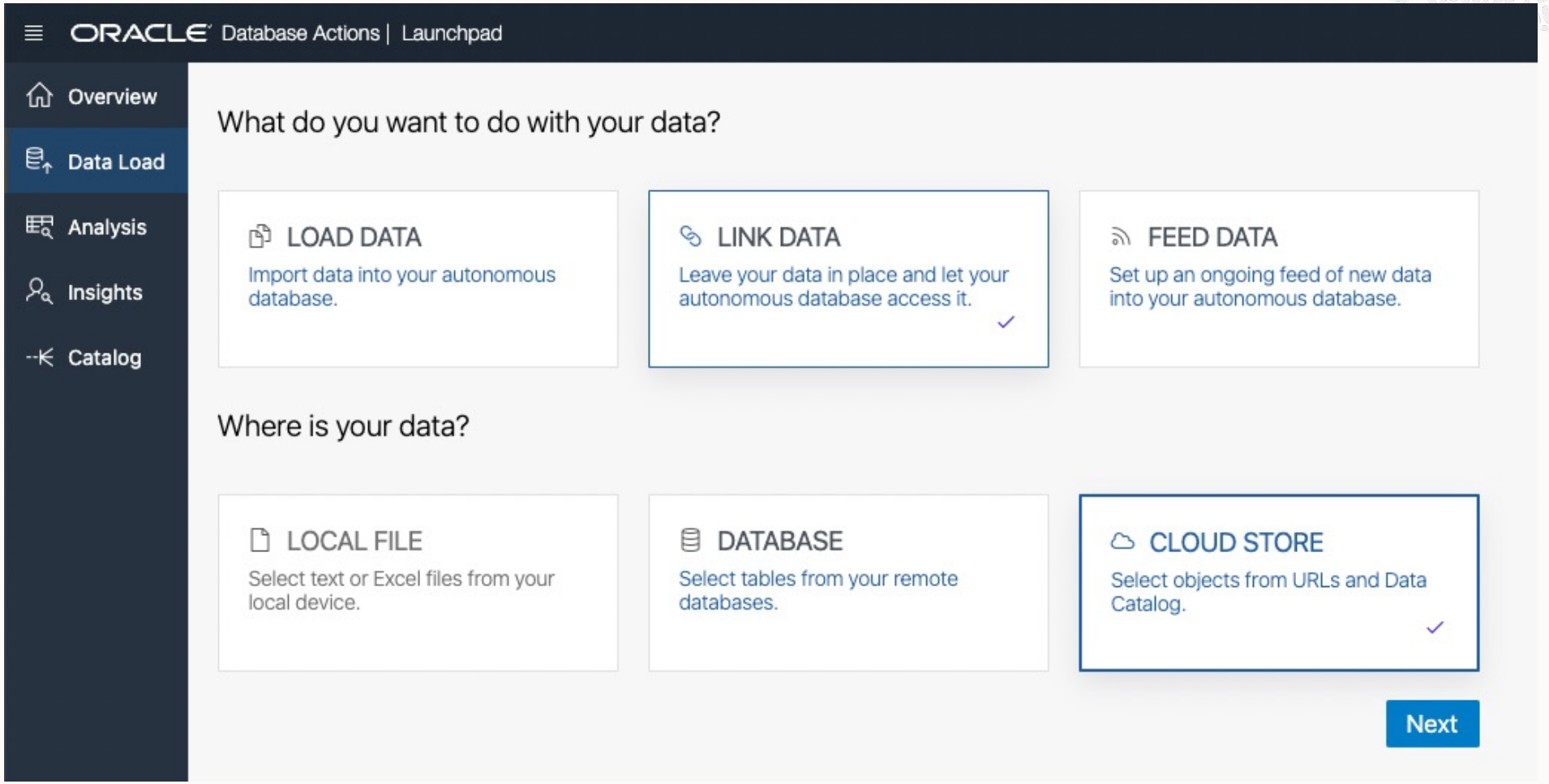
- Managed schemas and tables defined automatically
- No management required
- Use Oracle SQL to query both metadata and object storage data



ADB DCAT integration Live Lab

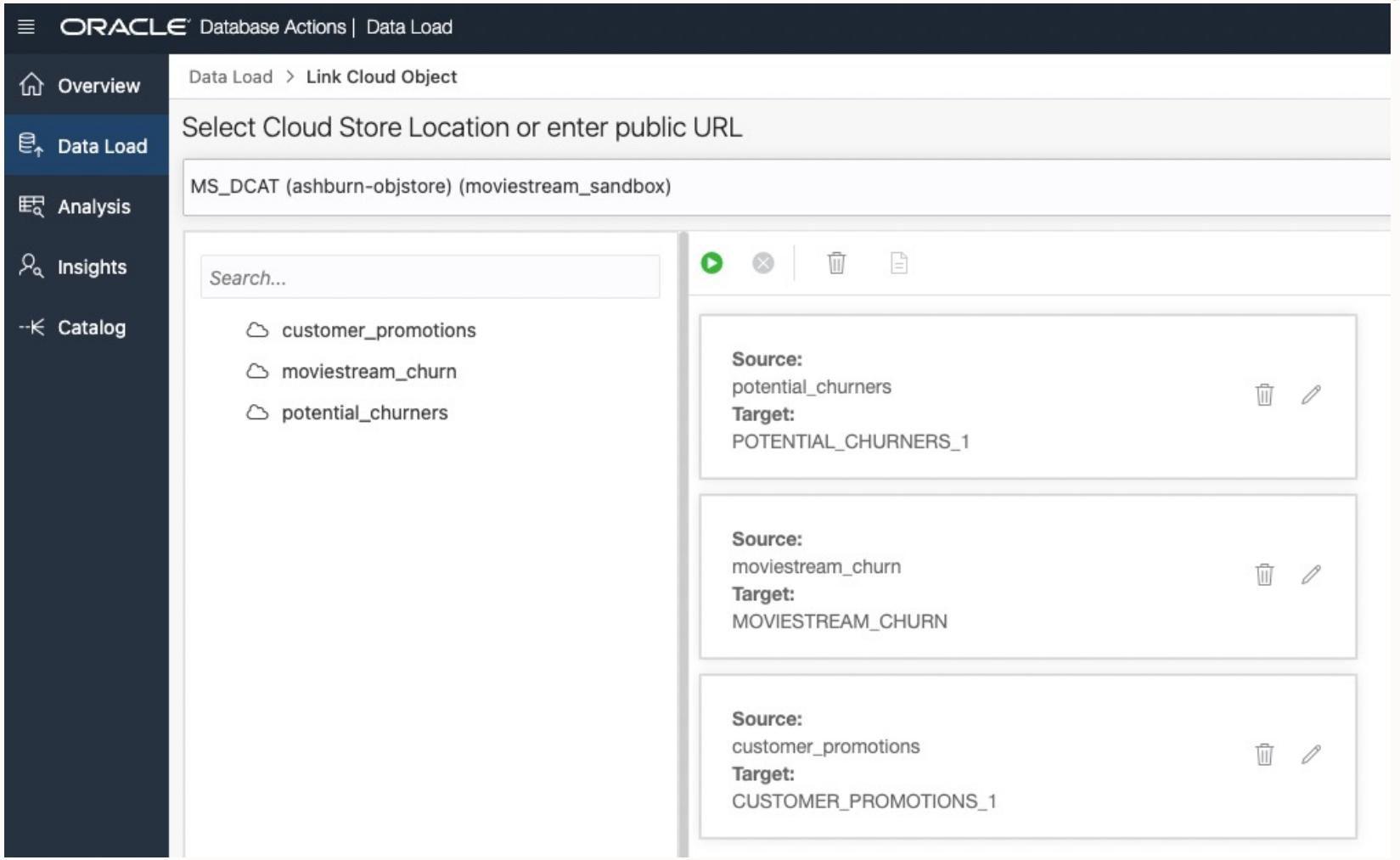
Want to load/link data with UI?

Database actions



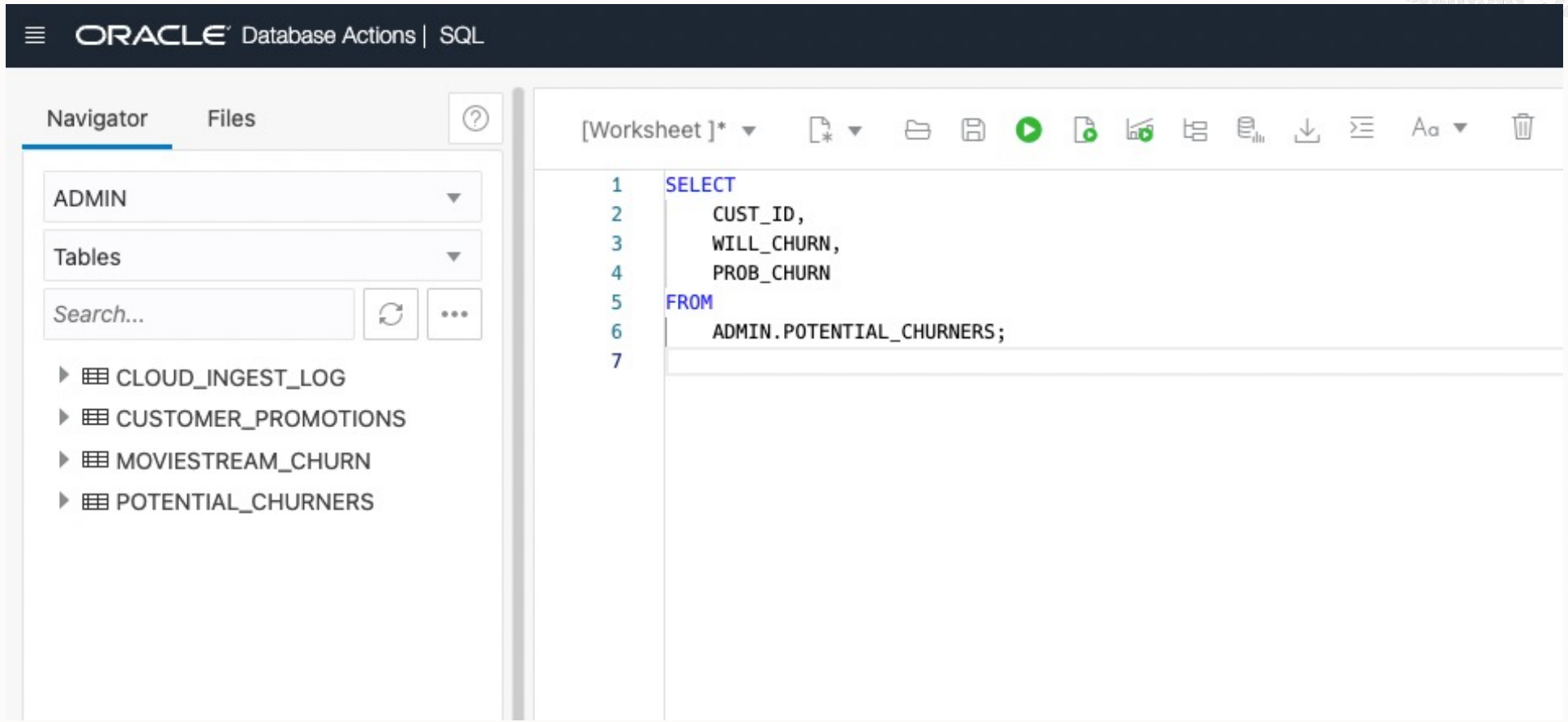
Want to load/link data with UI?

Database actions



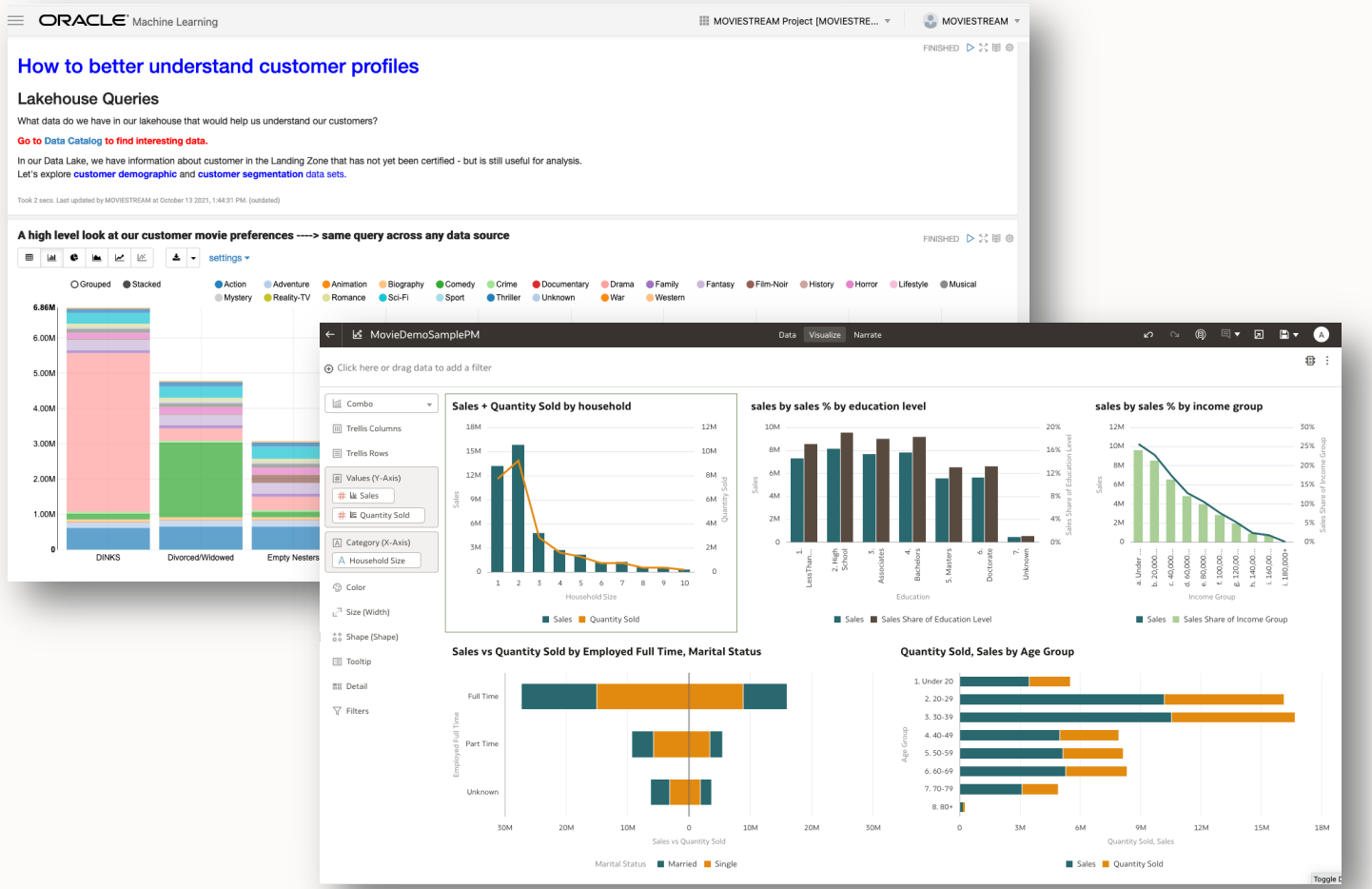
Want to load/link data with UI?

Database actions



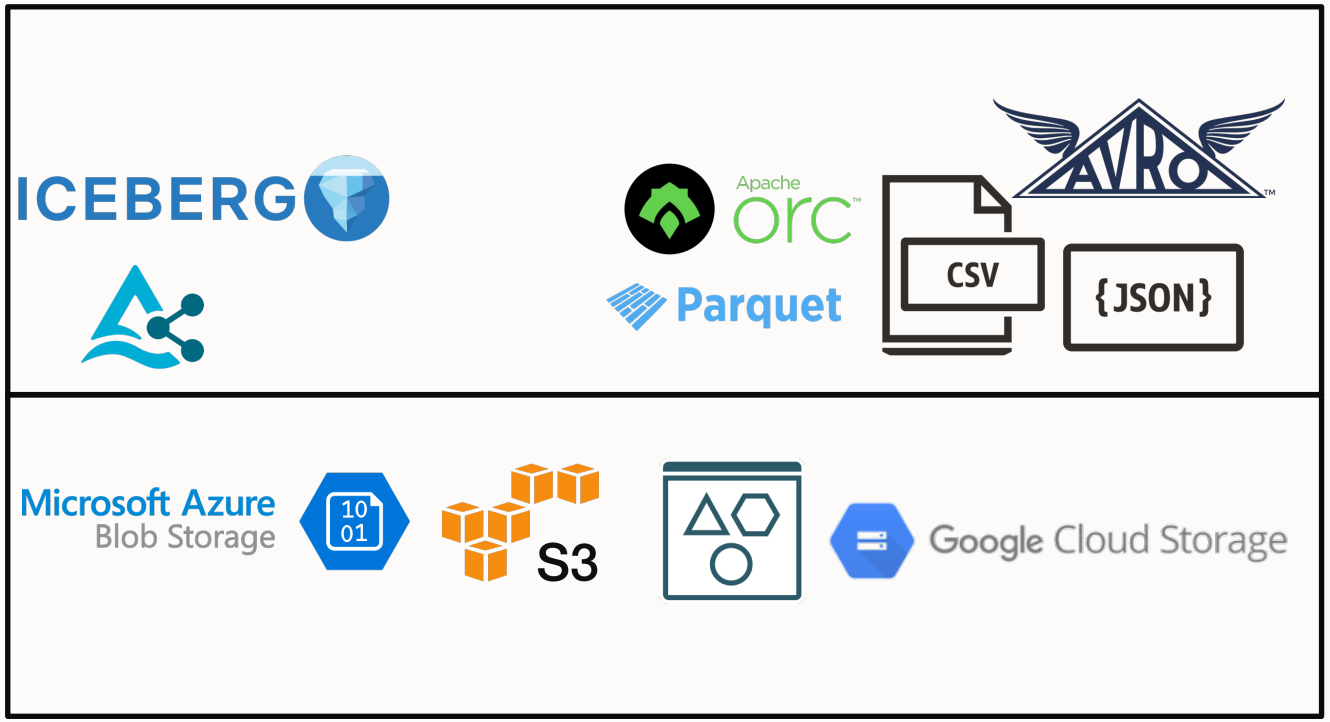
Gain insights from the lakehouse

One query spans storage types



- Correlate information from data lake and data warehouse
- Access from any SQL tool or application
- Preserve your investment in tools and skill sets
- Safeguard sensitive data using Oracle Database advanced security policies

ADB Lakehouse Sources
Summary



ADW processes data in multiple Clouds

- Oracle Cloud Infrastructure
- AWS
- Azure
- S3 compliant object stores, e.g. Google Cloud Platform

Multiple formats can be processed:

- Flat Files
  - CSV, Parquet, JSON, Avro, ORC
- Table Formats:
  - Delta Share
  - Apache Iceberg

Integration with Slack



Send notifications or query results to a Slack channel from ADB

Send messages to a Slack channel:

```
BEGIN
   DBMS_CLOUD_NOTIFICATION.SEND_MESSAGE(
      provider          => 'slack',
      credential_name   => 'SLACK_CRED',
      message           => 'Alert from Autonomous Database...',
      params            => json_object('channel' value 'C0....08'));
END;
/
```

Send notifications or query results to a Slack channel from ADB

Send output from a query to a Slack channel:

```
BEGIN
  DBMS_CLOUD_NOTIFICATION.SEND_DATA(
      provider => 'slack',
      credential_name => 'SLACK_CRED',
      query => 'SELECT username,account_status, expiry_date
              FROM account_users
              WHERE rownum < 5', params =>
                        json_object('channel' value 'C0....08',
                                    'type' value 'csv'));
END;
/
```

Oracle Data Safe



Security Assessment

User Assessment

Activity Auditing

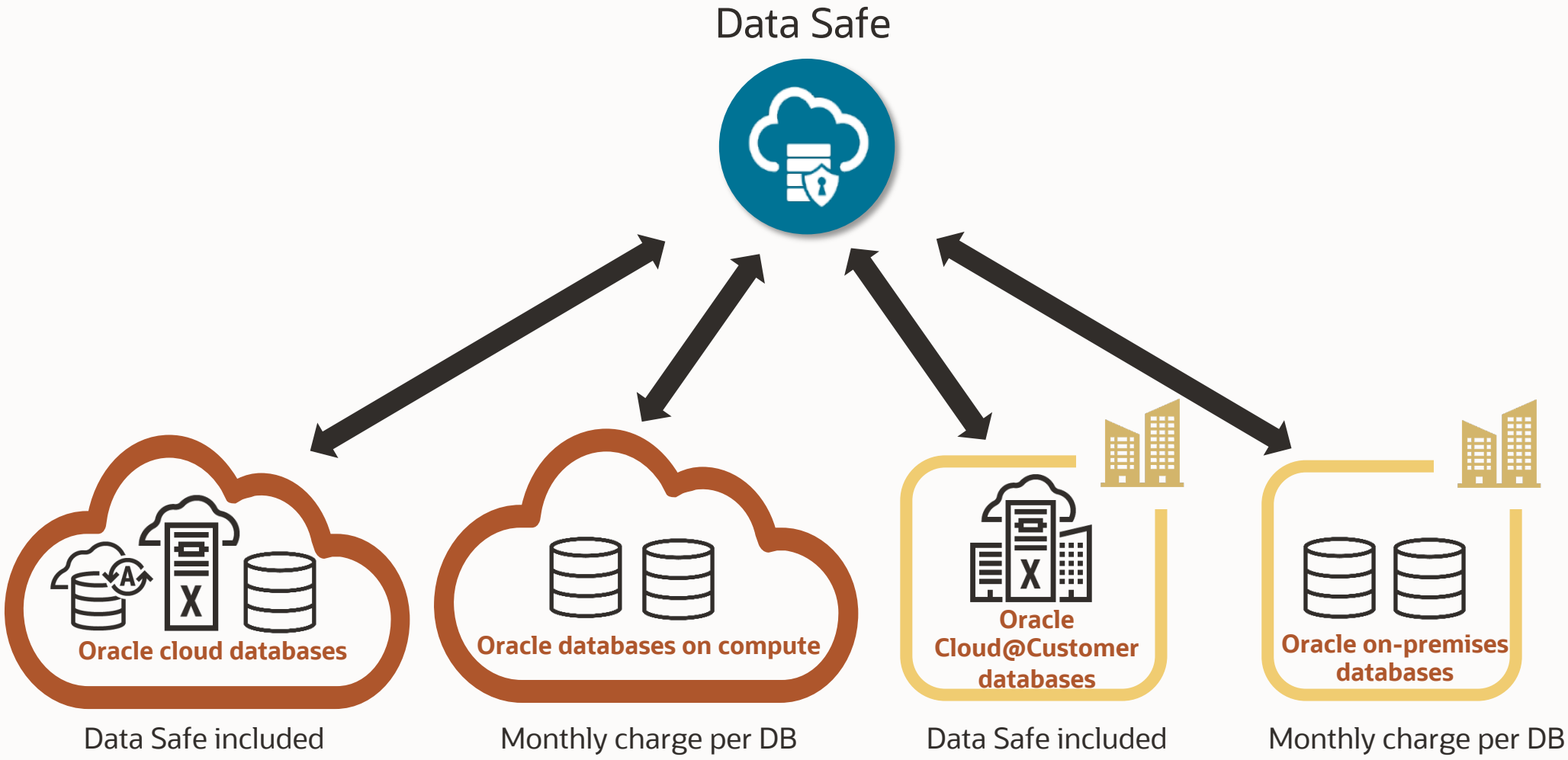
SQL Firewall*

Sensitive Data Discovery

Data Masking

*SQL Firewall is currently in limited availability in Data Safe (23c target databases only)

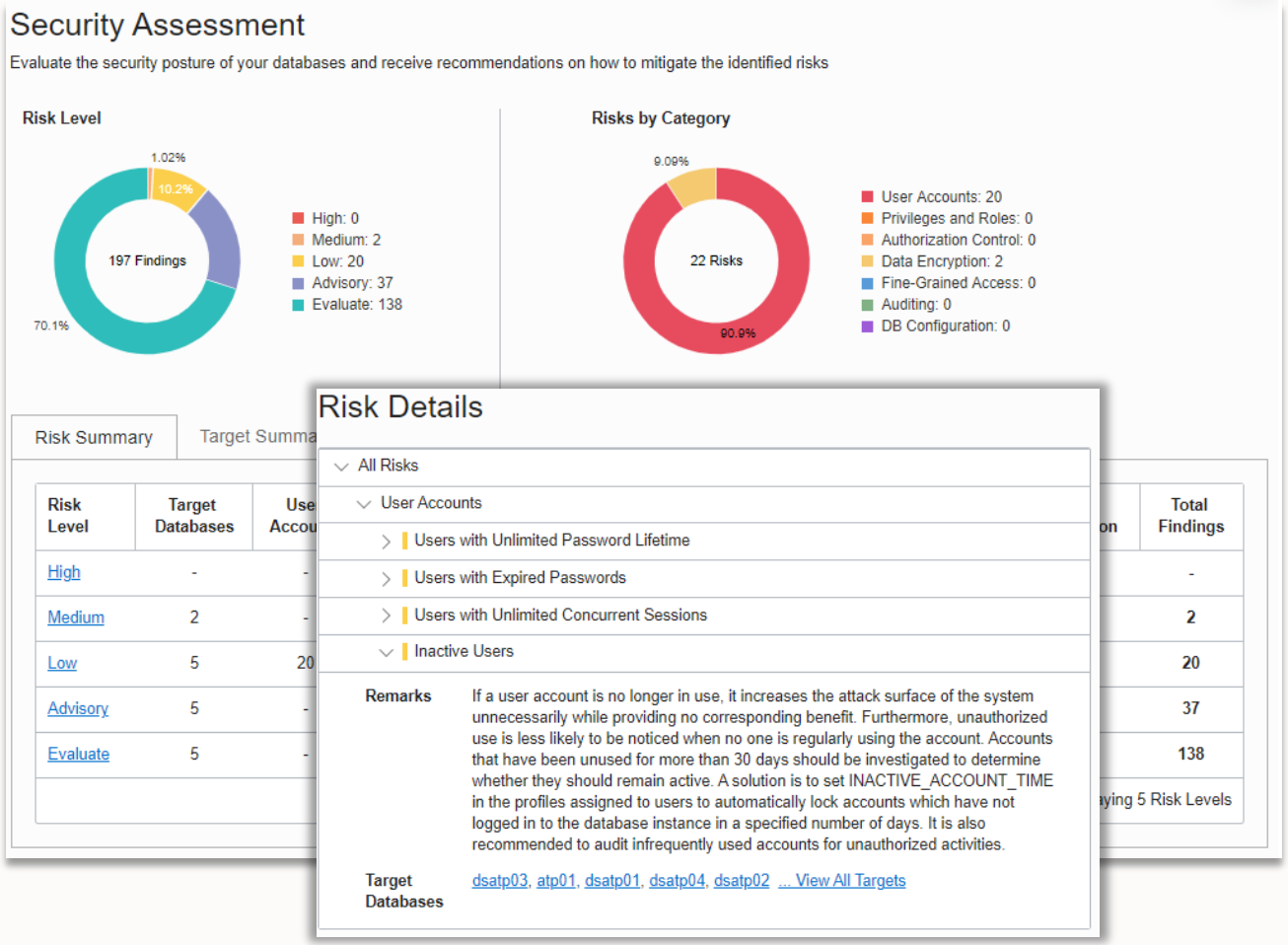
Data Safe is available for all your Oracle Databases



Security Assessment

Instant insights into configurations that may introduce unnecessary risk

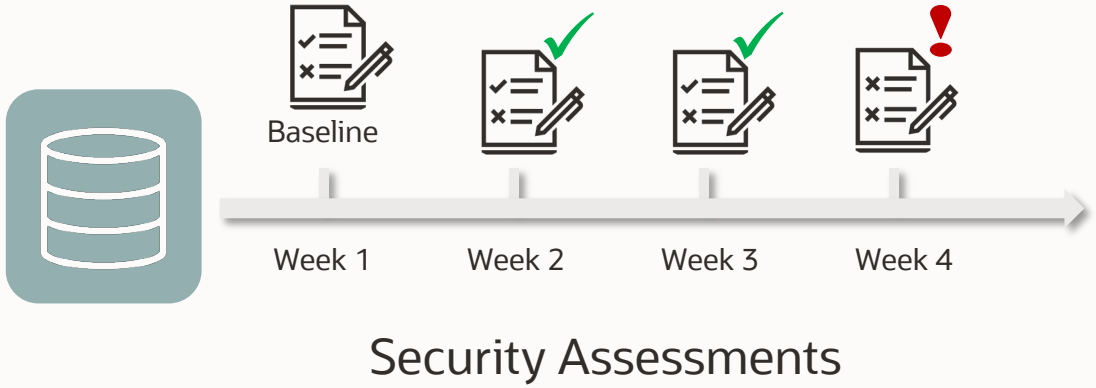
- Comprehensive assessment
  - Database configuration
  - Security controls in use
  - User roles and privileges
- Landscape-wide view on identified risks and recommendations
- Compliance mappings (GDPR, STIG, CIS)



Security Assessment

Detecting Configuration Drifts

- Establish a security baseline
- New assessments are automatically compared against the baseline
- Get notified and review any drift from your security baseline

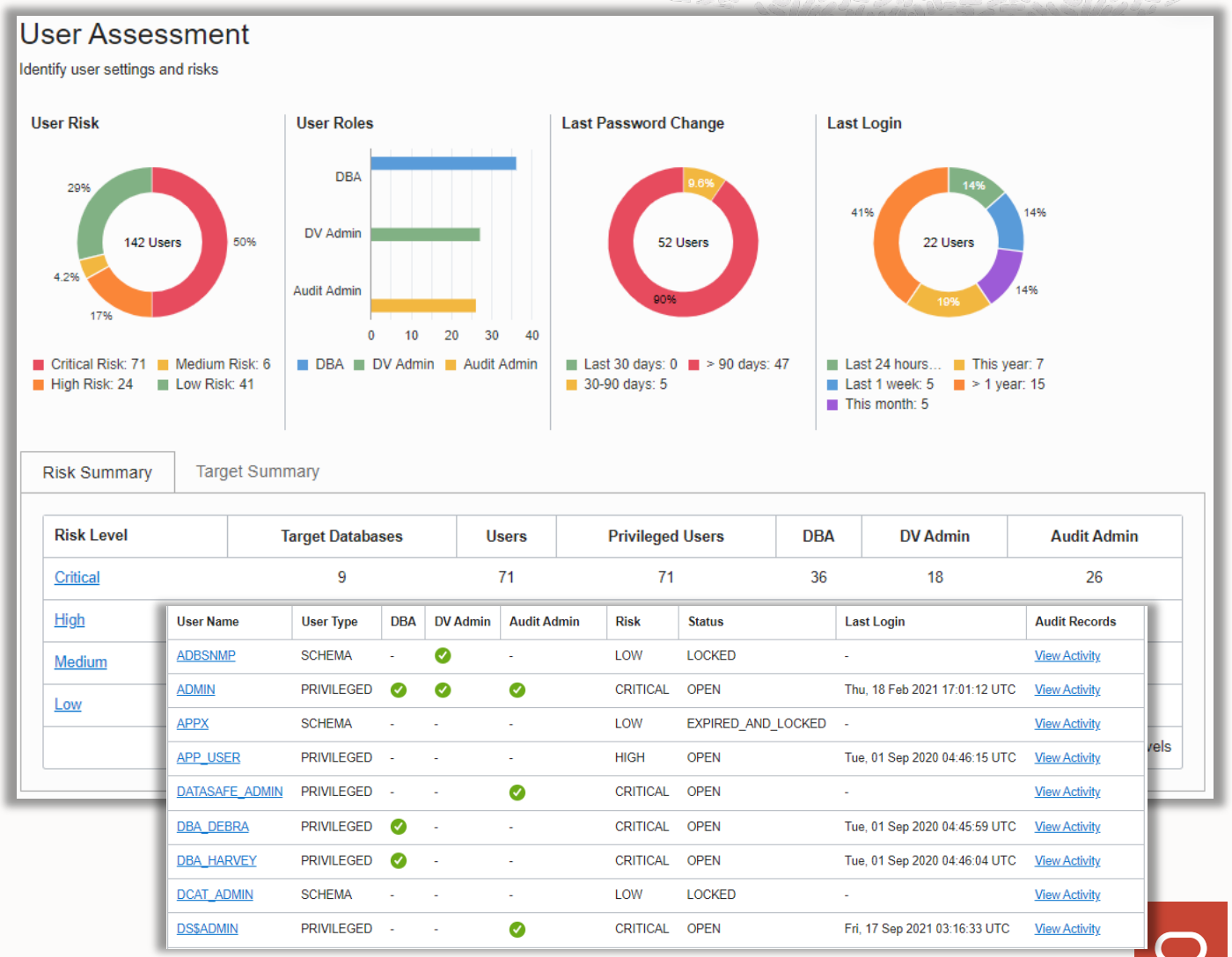


Security Assessments

User Assessment

Reduce risk from users by managing roles/privileges and policies

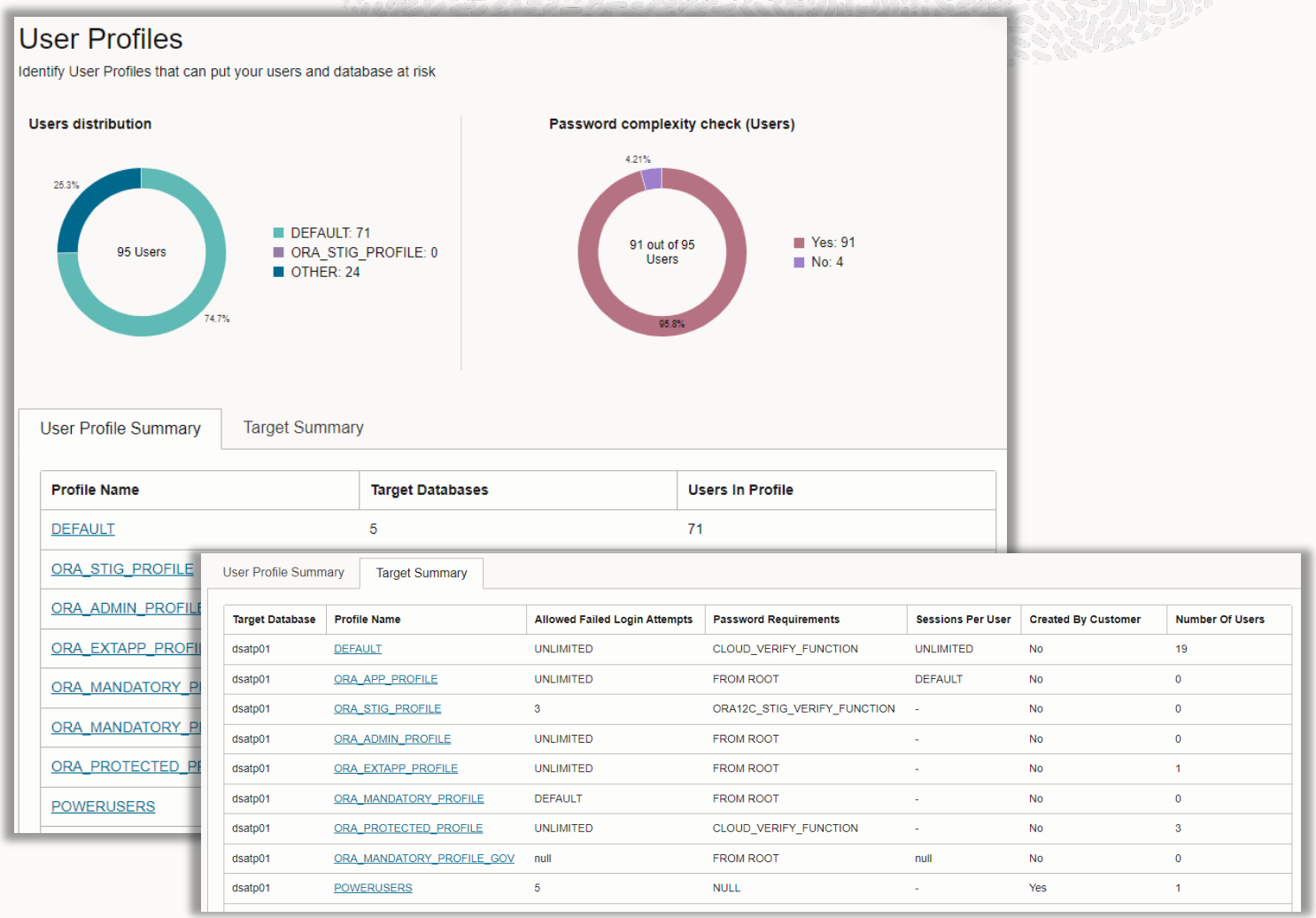
- Identify highly privileged users
- Review their roles and system privileges
- Evaluate user details like last login, password change, database activity



User Profile Insight

Evaluate password-related configurations

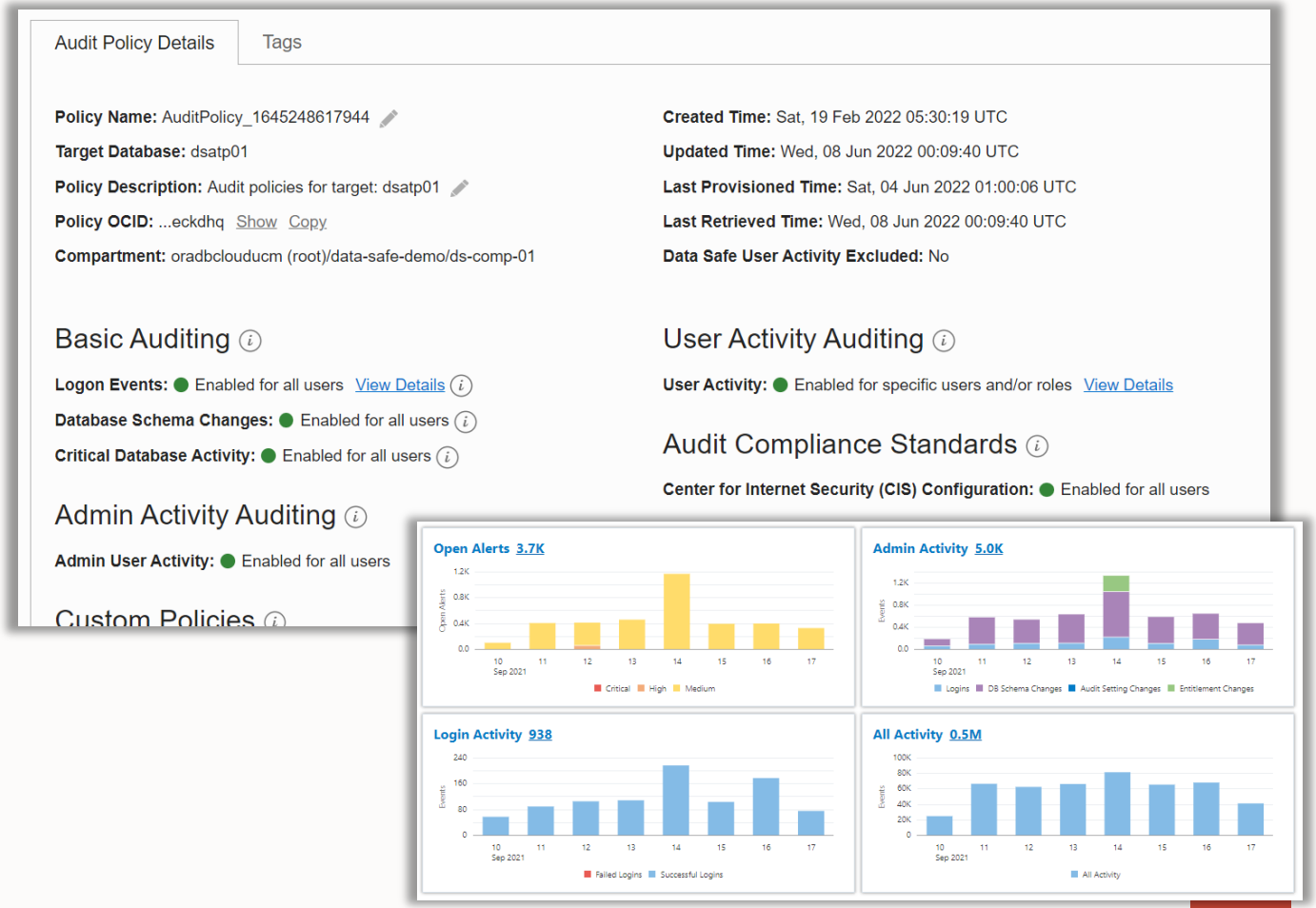
- Review existing user profiles and their parameters
- Identify which profiles are assigned to which users
- Easily identify users and profiles without a password complexity function



Activity Auditing

Track user actions and streamline auditing with robust reporting

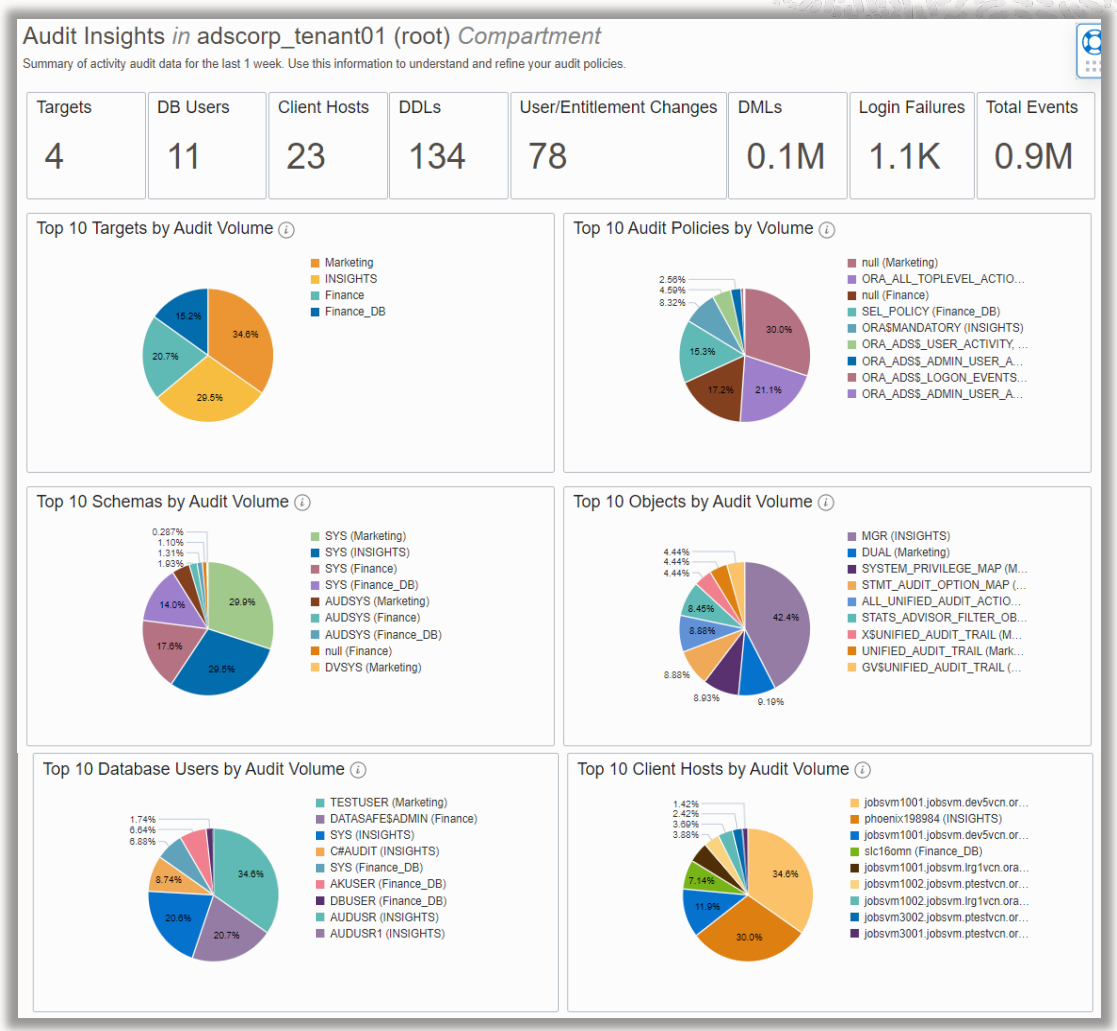
- Provision audit, compliance, and alert policies
- Collect audit data from databases, and track sensitive operations
- Audit Reports
  - Interactive reports for forensics
  - Summary and detailed reports
  - PDF reports for compliance



Auditing Insights

Fine-tune your audit policies

- Get more insight into your audit data
  - Which audit policy is generating the most records?
  - Which target is generating the highest volume?
  - Which objects or schemas are accessed the most?
  - etc.

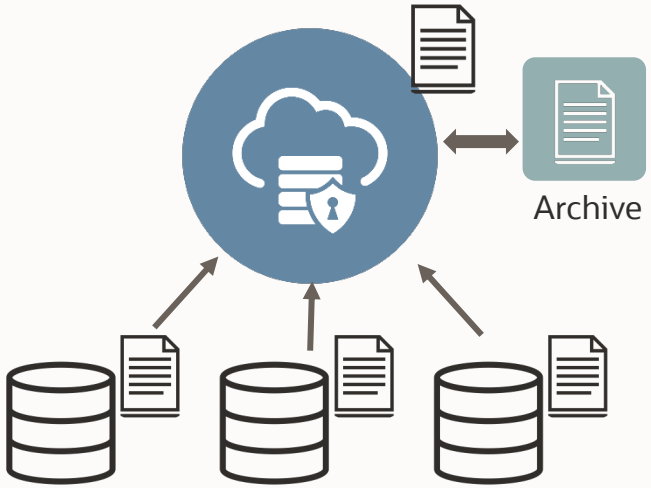


Auditing data retention management

Comply with regulations and corporate policies

Keep your Audit Data for up to 7 years

- Online retention period up to 1 year
- Archive retention period up to 6 additional years
- Configure retention periods globally and/or per target
- Easily retrieve audit records from the archive
- Fully managed by Data Safe



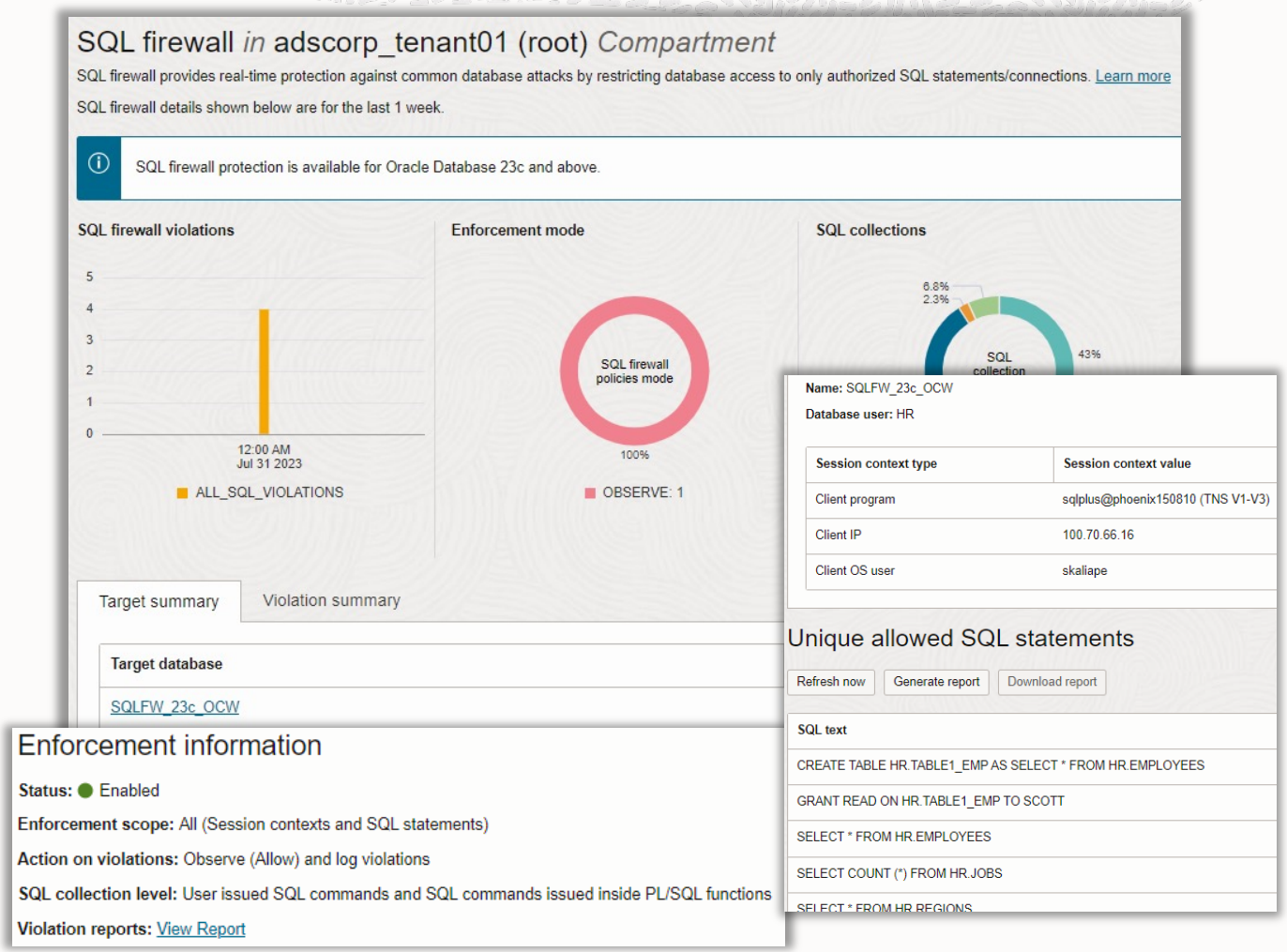
There is no additional fee for archiving your audit data in Data Safe!

SQL Firewall*

Prevent SQL injection and access from unauthorized access points

Limited Availability

- Provides real-time protection against common database attacks by restricting database access to
  - authorized connections
  - authorized SQL statements
- Block or monitor any violations
- Mitigates risks from SQL injection attacks, anomalous access, and credential theft/abuse



SQL Firewall is currently in limited availability in Data Safe (23c target databases only)

SQL Firewall*

Easy configuration, management, and monitoring in Data Safe

Limited Availability



1

Collect

Turn on the SQL statement and user connection collection

2

Review & Modify

Review the SQL collection

Review and modify the allowed user connections (as required)

3

Enforce

Block or monitor any unauthorized SQL and/or user connections

4

Monitor

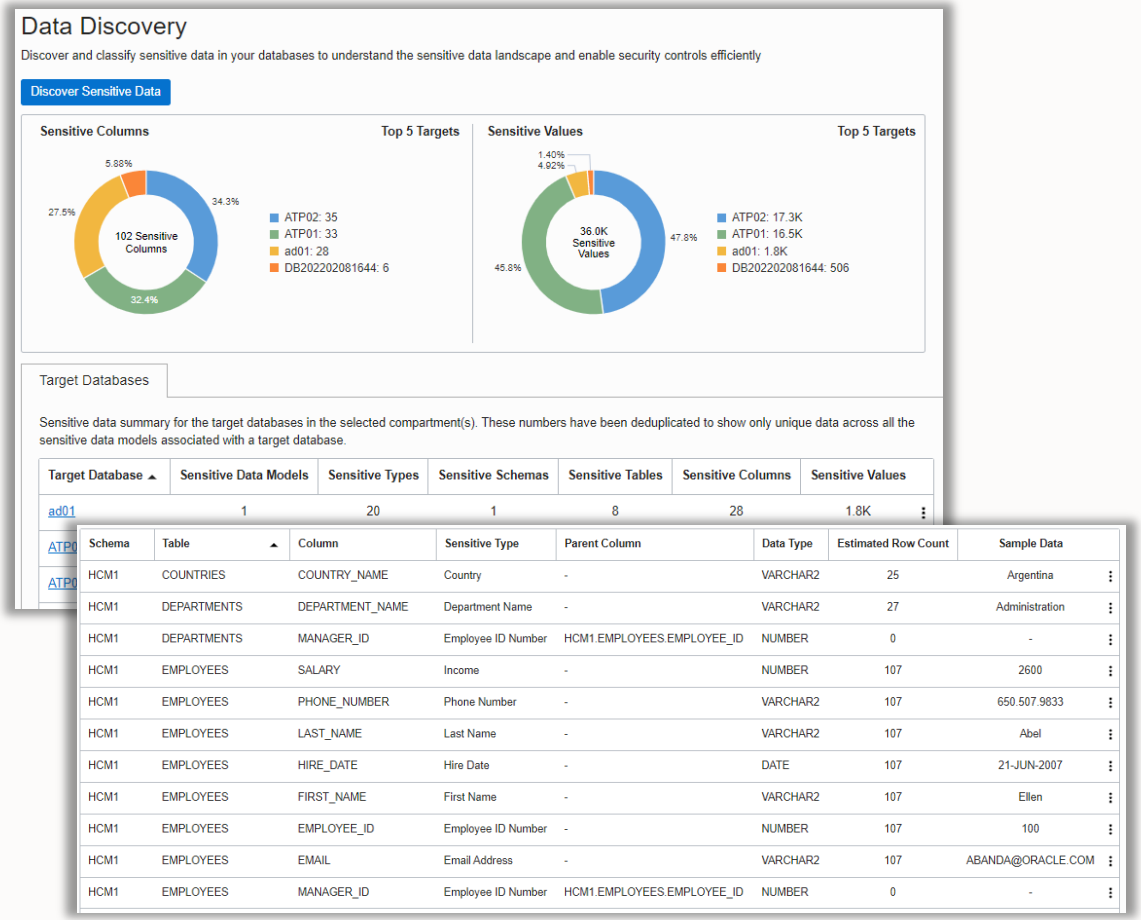
Monitor any violations

**SQL Firewall is currently in limited availability in Data Safe (23c target databases only)*

Sensitive Data Discovery

Prioritize security efforts by identifying the location, type and amount of sensitive data

- Discovers/classifies 150+ sensitive types
- User-defined sensitive types
- Incremental discovery
- Reports amount / type of sensitive data



Sensitive Data Discovery

150+ pre-defined sensitive data types



Identification

SSN
Name
Email
Phone
Passport
DL
Tax ID
…



Biographic

Age
Gender
Race
Citizenship
Address
Family Data
Date of Birth
Place of Birth
…



IT

IP Address
User ID
Password
Hostname
GPS location
…



Financial

Credit Card
CC Security
PIN
Bank Name
Bank Account
IBAN
Swift Code
…



Healthcare

Provider
Insurance
Height
Blood Type
Disability
Pregnancy
Test Results
ICD Code
…



Employment

Employee ID
Job Title
Department
Hire Date
Salary
Stock
…



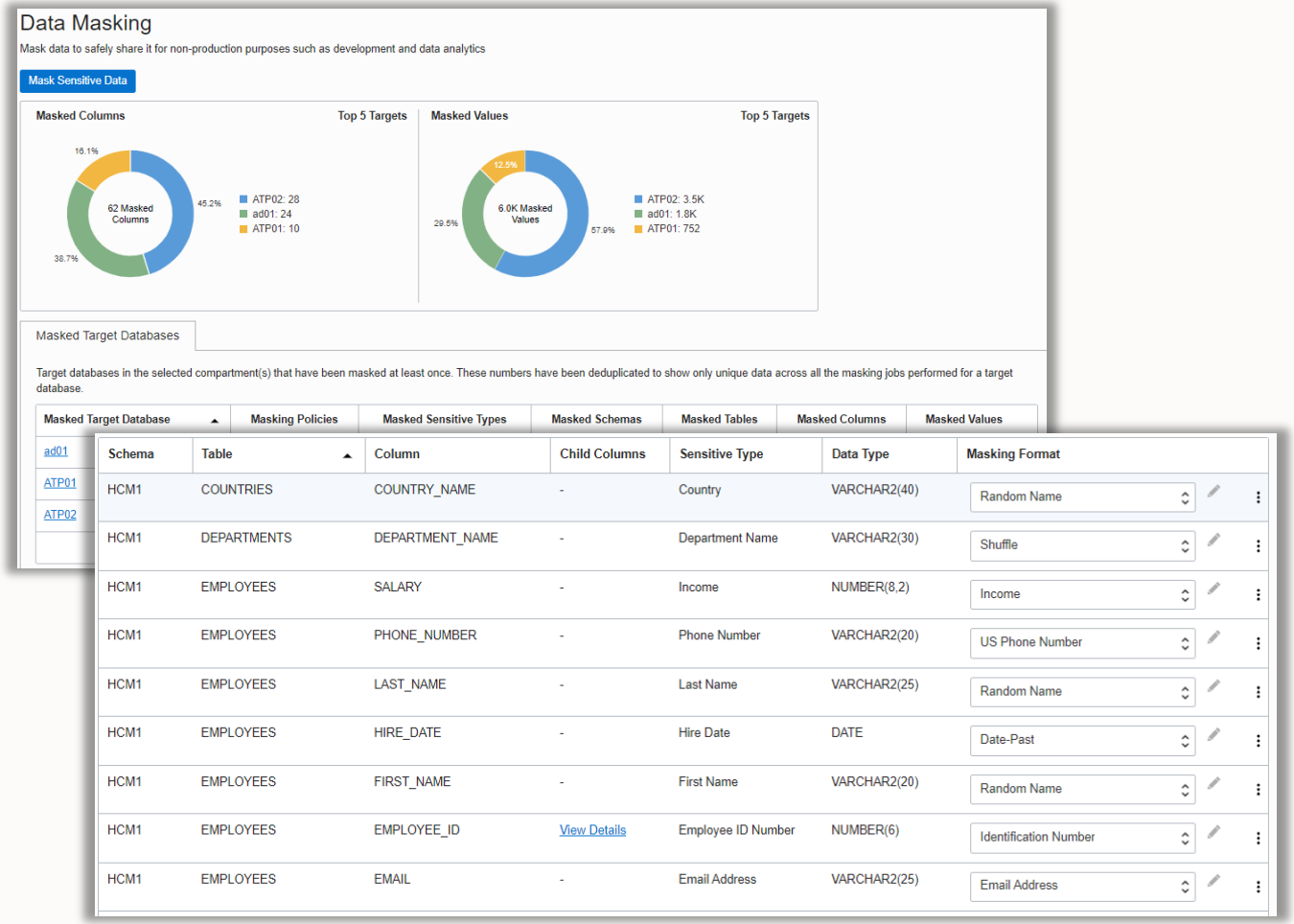
Academic

College Name
Grade
Student ID
Financial Aid
Admission Date
Graduation Date
Attendance
…

Sensitive Data Masking

Prioritize security efforts by identifying the location, type and amount of sensitive data

- Mask data identified as sensitive
  - 50+ predefined masking formats
  - Automated format selection based upon sensitive type
  - Optional user-defined masking formats
- Rich masking transformations for complex cases
- Extensible & customizable

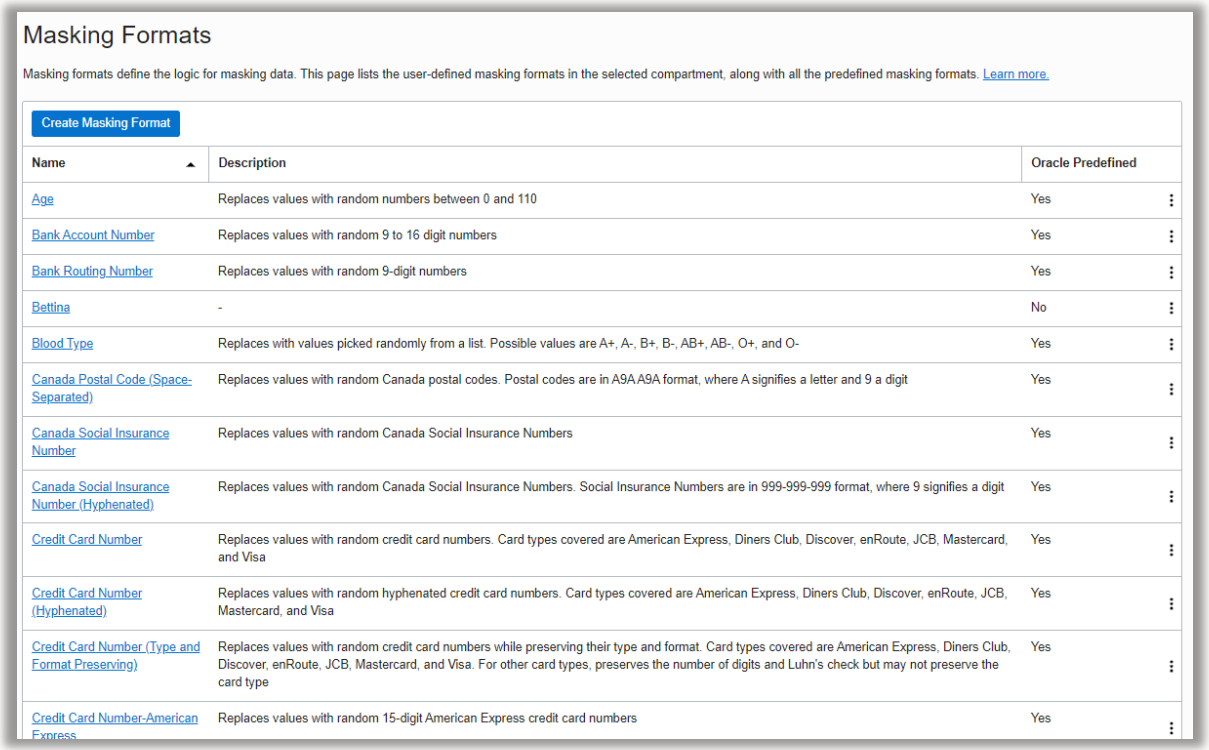


Sensitive Data Masking

Pre-defined masking formats

Data Safe provides 50+ pre-defined masking formats, including

- Specific masking formats like
  - social security number
  - credit card number
  - email address
  - etc.
- As well as generic masking formats
  - random date, number, name, …
  - fixed number, fixed string
  - format preserving randomization
  - regular expression
  - truncate data
  - group masking
  - etc.



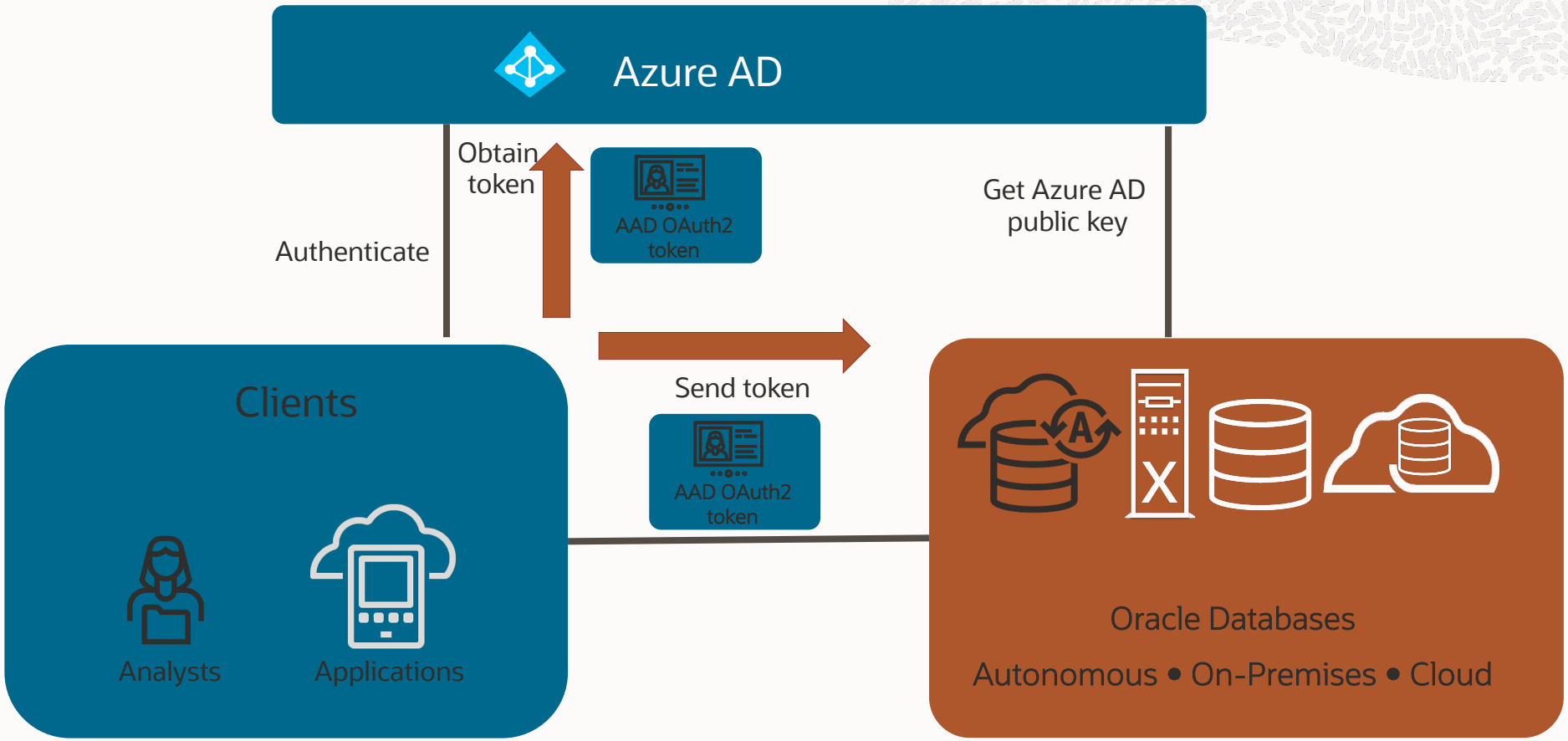
Masking Formats

Masking formats define the logic for masking data. This page lists the user-defined masking formats in the selected compartment, along with all the predefined masking formats. Learn more.

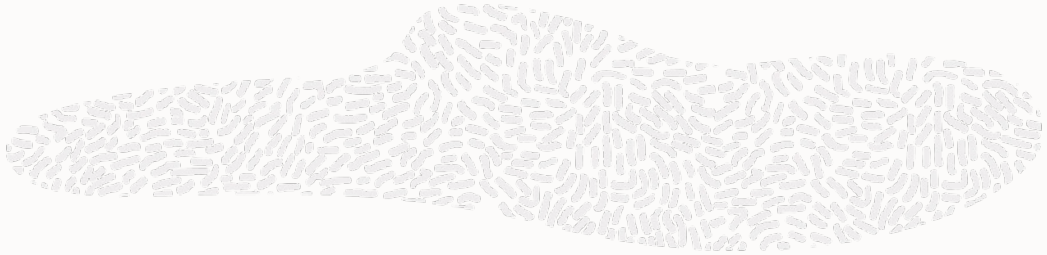
Create Masking Format

Name	Description	Oracle Predefined
Age	Replaces values with random numbers between 0 and 110	Yes
Bank Account Number	Replaces values with random 9 to 16 digit numbers	Yes
Bank Routing Number	Replaces values with random 9-digit numbers	Yes
Bettina	-	No
Blood Type	Replaces with values picked randomly from a list. Possible values are A+, A-, B+, B-, AB+, AB-, O+, and O-	Yes
Canada Postal Code (Space-Separated)	Replaces values with random Canada postal codes. Postal codes are in A9A A9A format, where A signifies a letter and 9 a digit	Yes
Canada Social Insurance Number	Replaces values with random Canada Social Insurance Numbers	Yes
Canada Social Insurance Number (Hyphenated)	Replaces values with random Canada Social Insurance Numbers. Social Insurance Numbers are in 999-999-999 format, where 9 signifies a digit	Yes
Credit Card Number	Replaces values with random credit card numbers. Card types covered are American Express, Diners Club, Discover, enRoute, JCB, Mastercard, and Visa	Yes
Credit Card Number (Hyphenated)	Replaces values with random hyphenated credit card numbers. Card types covered are American Express, Diners Club, Discover, enRoute, JCB, Mastercard, and Visa	Yes
Credit Card Number (Type and Format Preserving)	Replaces values with random credit card numbers while preserving their type and format. Card types covered are American Express, Diners Club, Discover, enRoute, JCB, Mastercard, and Visa. For other card types, preserves the number of digits and Luhn's check but may not preserve the card type	Yes
Credit Card Number-American Express	Replaces values with random 15-digit American Express credit card numbers	Yes

Using Azure AD OAuth2 access tokens with Oracle Database



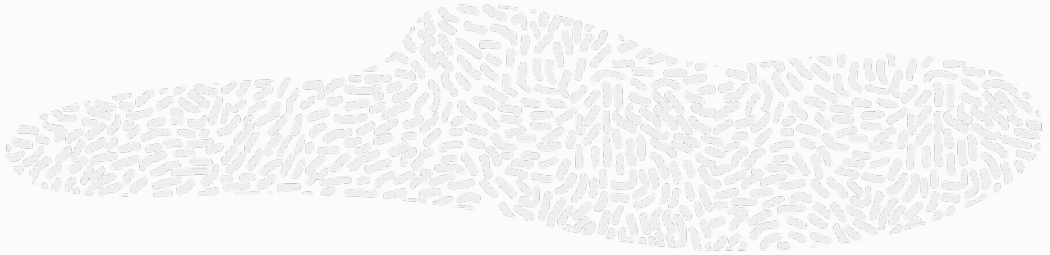
What's next?



Oracle Database 23c

- Starting with the following drivers: JDBC-thin, ODP.NET core, ODP.NET managed
  - Native cloud authentication
  - Azure AD device code flow
  - Azure managed identity flow

What's next?



Azure AD v2 tokens

- Available with ADB-Serverless 19c today

Microsoft Power BI integration

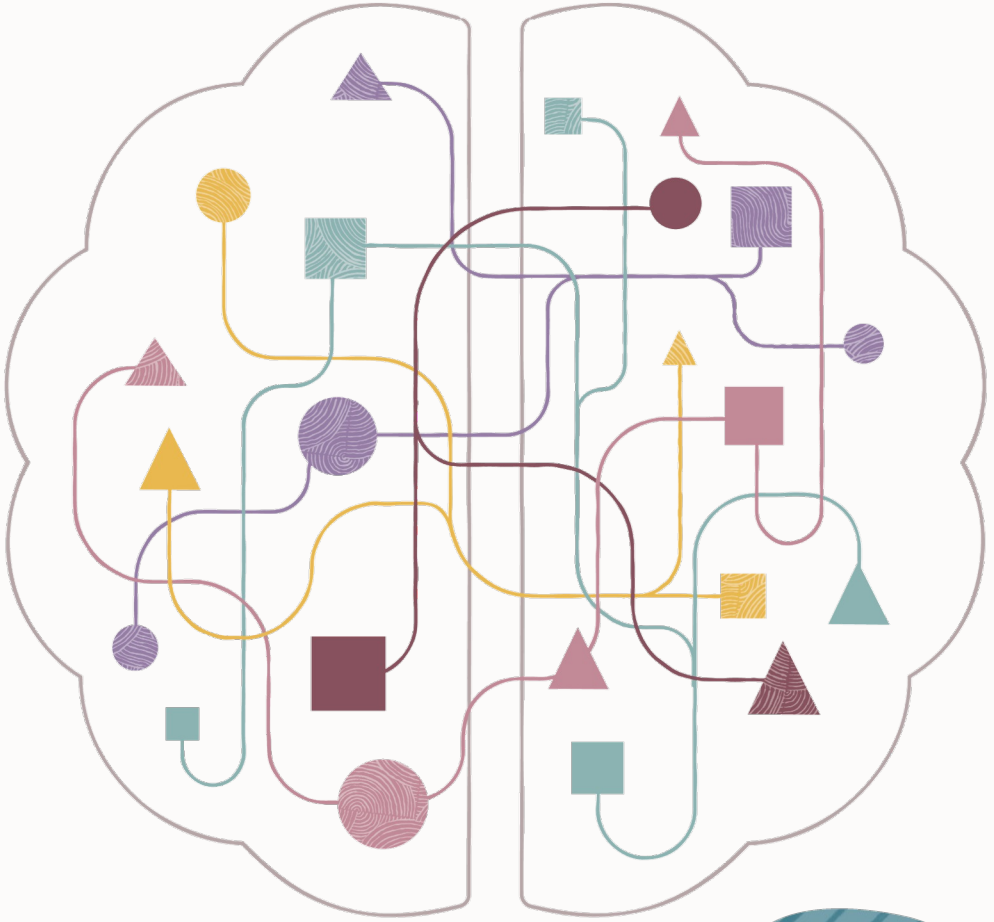
- Power BI desktop includes this feature today
- Microsoft adding SSO experience for Power BI users through the Oracle Database – coming soon!

Autonomous Database Select AI

Simplest way to get answers about your business

Use **your language** to query data

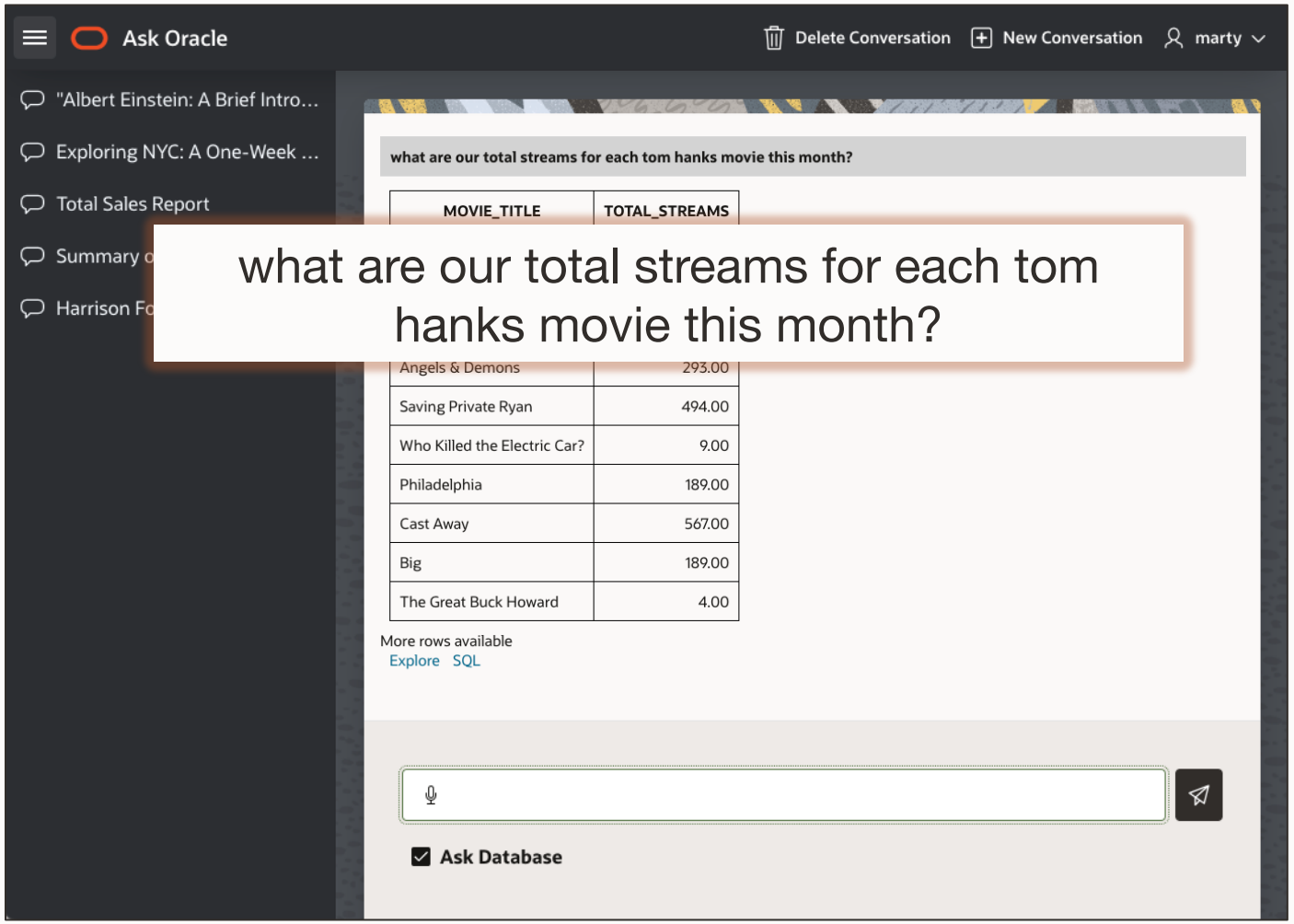
No need to understand where and how your data is stored to gain insights



Select AI - Simplest way to get answers about your business

Just ask a question

Autonomous Database manages the entire query process to produce your answer



Historically, answering these types of questions has not been easy

We infer a lot from human language

what are our total streams for each tom hanks movie this month?

- total streams → total number of movie views
- for each → breakout views by movie
- tom hanks → tom hanks is an actor
- this month → understanding of time



LLMs are remarkable at inferring intent (and getting better)
They are not perfect! It is **very important** to verify results

Select AI translates your language into Oracle SQL language

Processes question using an AI large language model (LLM)

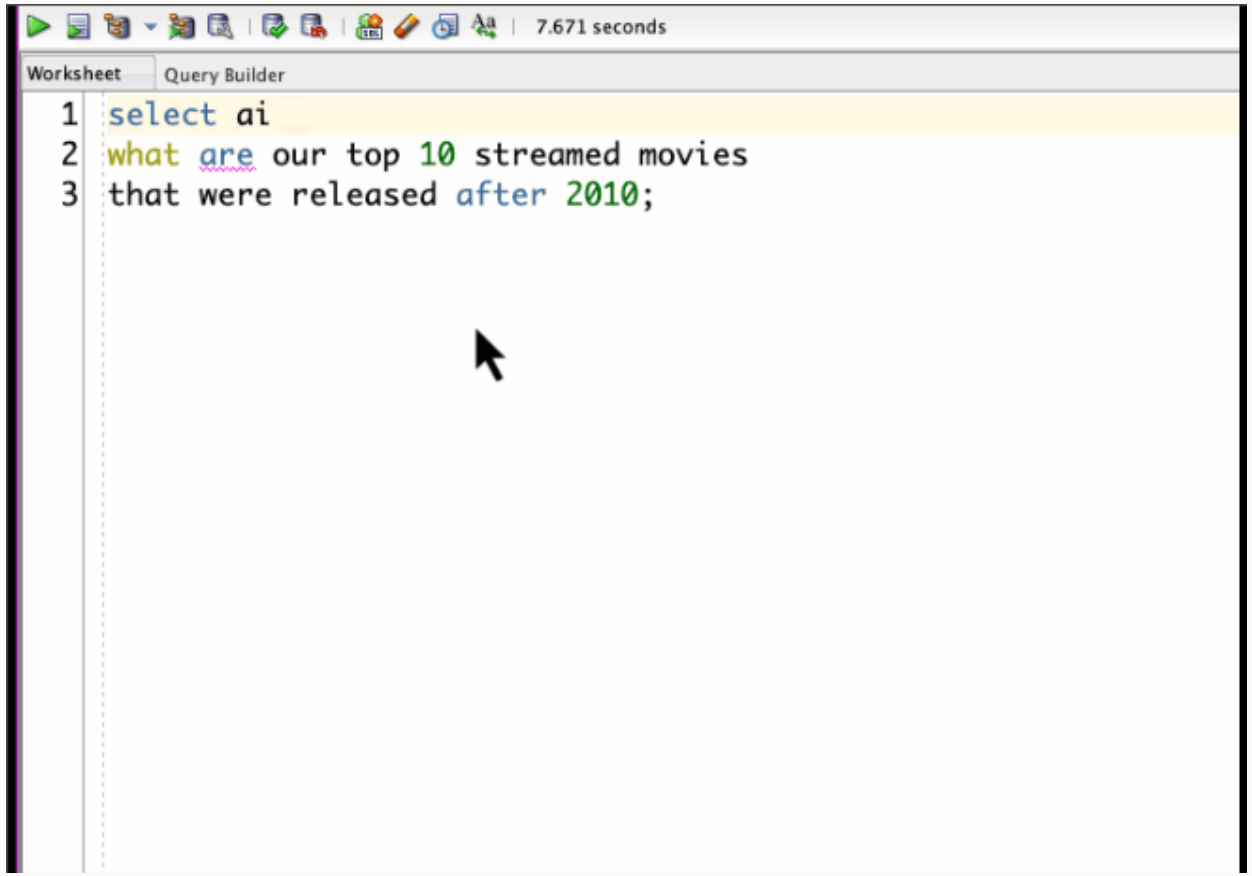
Question	Inference
total streams →	total number of movie views →
for each →	breakout views by movie →
tom hanks →	tom hanks is an actor →
this month →	Understanding of time →

```
SELECT
  m.title AS movie_title,
  COUNT(s.views) AS total_streams
FROM movie m
  JOIN sales_sample s ON m.movie_id = s.movie_id
  JOIN actors a ON m.movie_id = a.movie_id
WHERE a.actor = 'Tom Hanks'
  AND EXTRACT(MONTH FROM s.day_id) =
      EXTRACT(MONTH FROM SYSDATE)
GROUP BY m.title
```

Easy to extend and build new natural language apps

Use a standard SELECT statement followed by AI and your question

Process the result as you would any other SQL result set



Integrate natural language queries with your application

SELECT AI what are our top selling movies?

Result:

Movie Title	Total Sales
Aladdin	82426.71
Aquaman	78091.19
Captain Marvel	77146.21
The Lion King	66606.75
Spider-Man: Far from Home	66516.26

Optional ACTION keywords:

```
runsql    return sql result set
narrate   return a conversational result
showsql   return the generated query
chat      general AI-chat conversation
```

Easy to configure your data for natural language queries

Use one or more AI profiles that is best for your business

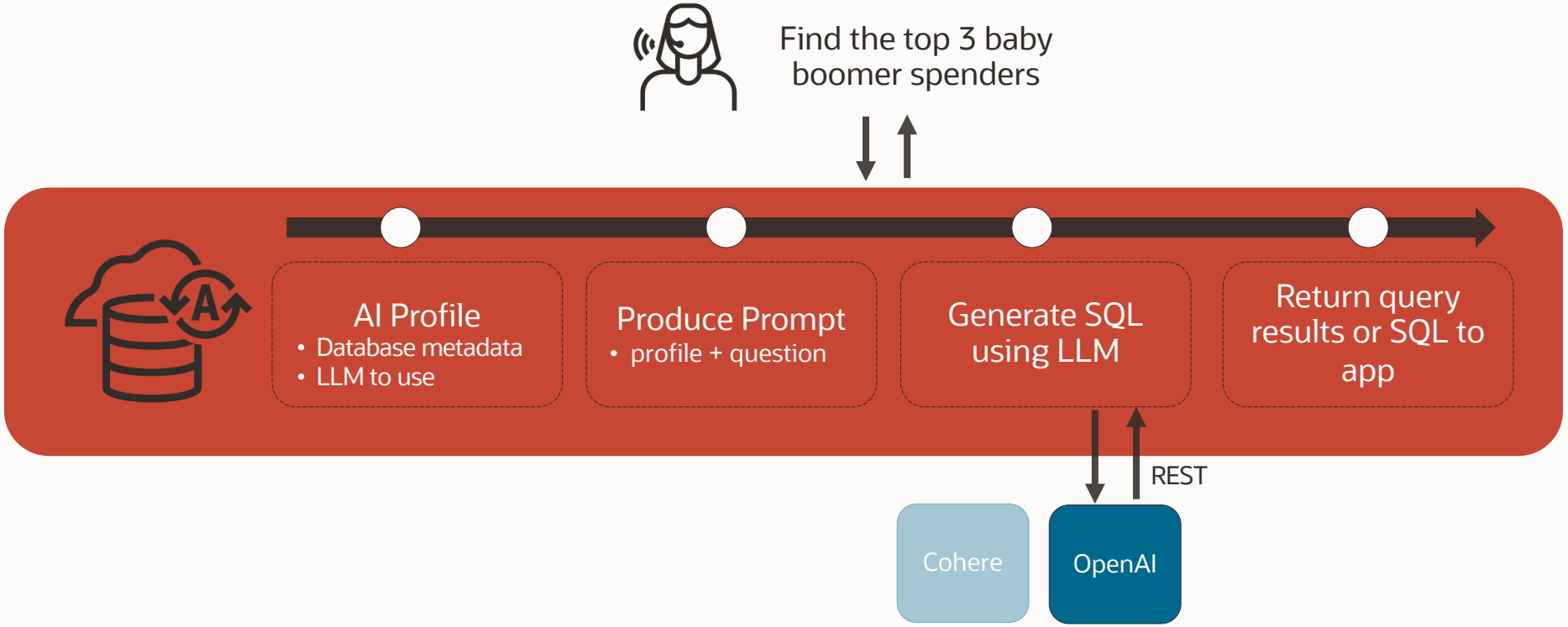
1. Choose an LLM to generate a database query from natural language

2. Specify schemas, tables and/or views to participate in processing

Simple PLSQL API creates AI profile:

```
dbms_cloud_ai.create_profile(
  profile_name => 'movie_nl_processing',
  attributes =>
  '{"provider": "openai",
    "credential_name": "OPENAI_CRED",
    "object_list": [{"owner": "myschema",
                     "name":  "movie"},
                    {"owner": "myschema",
                     "name":  "sales_sample"},
                    {"owner": "myschema",
                     "name":  "customer"} ]
   }'
);
```

SQL query generation process flow



Oracle Database Announcements at CloudWorld 2023

- Semantic search capabilities in Oracle Database 23c, **AI Vector Search**, enables the database to store semantic data as vectors and use them to run fast similarity queries

- Autonomous Database – extend applications with natural language capabilities by posing **natural language questions** directly to Autonomous Database. Developers can select the best **LLM** from Oracle or 3rd parties to generate SQL queries to respond to these questions

- APEX – accelerate the process of building secure, scalable, and feature-rich enterprise applications by using AI to **convert natural language** prompts into SQL Queries and then create the desired app components.

Press Release: www.oracle.com/news/announcement/ocw-integrated-vector-database-augments-generative-ai-2023-09-19

Larry's OCW Keynote: Is Generative AI the most important New Computer Technology Ever? Probably!

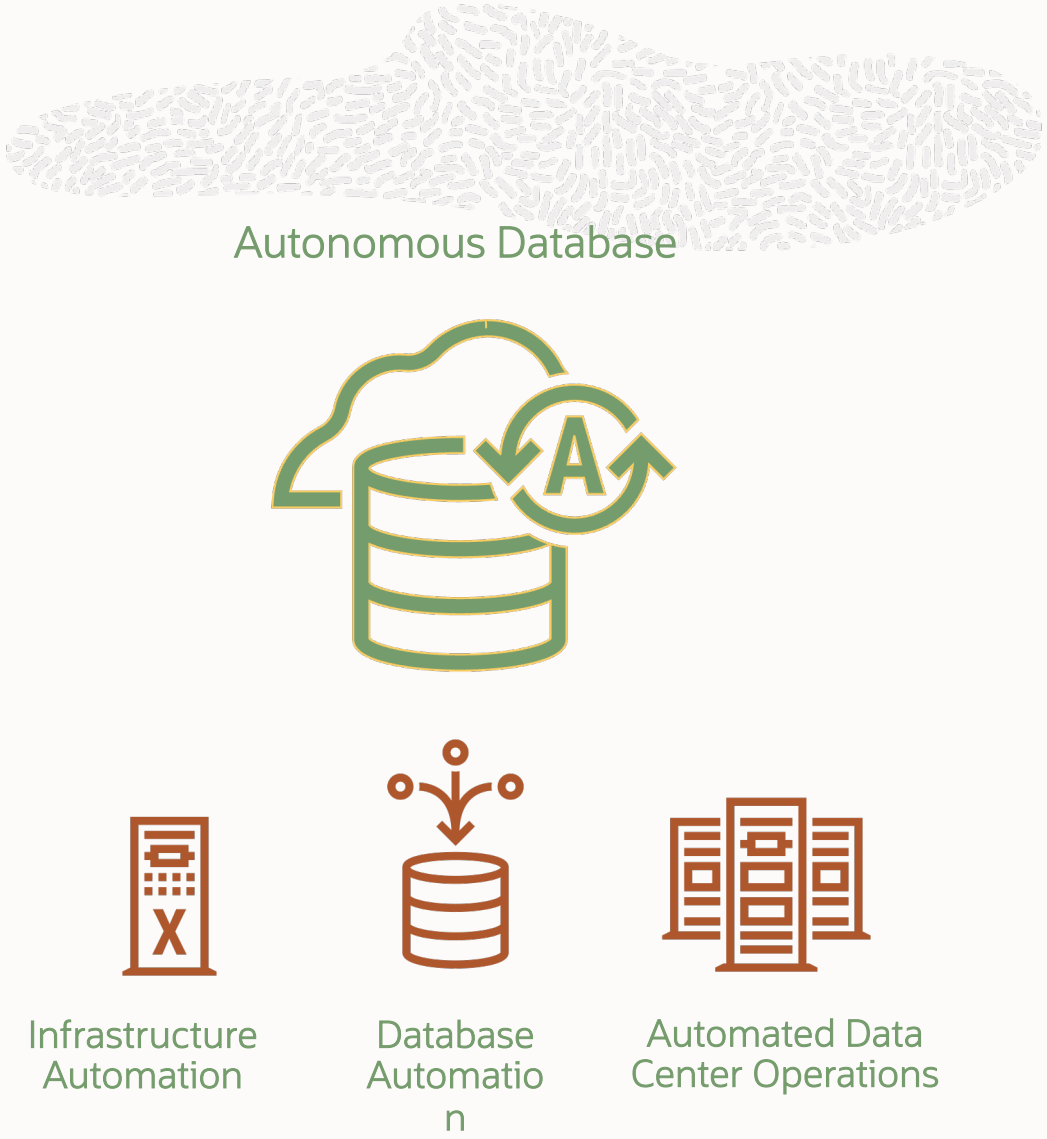
Autonomous Database and AI

Simplest database for developing and running any applications or analytics at any scale or criticality

Uses **AI** to eliminate routine management

- Provisioning, security, patching, scaling, optimization

Uses **AI** to find and fix workload issues



ADB Configurations are customizable

Feature		
Database Version	Latest 19c update	Latest and previous 19c Database RU
Memory Size (SGA/PGA)	Fixed per OCPU	Configurable, important for DB In-Memory
User Tablespace	No	Yes, CREATE/ALTER TABLESPACE
Database Parameters	Service managed	ALTER SYSTEM/SESSION, Apps-specific parameters via tagging
Database Character Set	Unicode AL32UTF8	Configurable - any 19c character sets
SYSDATE/SYSTIMESTAMP Time Zone	UTC	Configurable – any time zones
Minimum Database Size	1 OCPU and 1 TB storage	0.1 OCPU and 32 GB storage
Availability Domain Placement	Service managed	Yes
Backup Retention Period	Fixed 60 days	1 day to 10 years
Access to File System	No	Attach external NFS file system or OCI FSS
Database connection	Requires Wallet	+ regular SQL*Net TCP no wallet, one-way TLS
Listener Port#	Fixed Port#	Configurable
SSL and ORD Certificates	Service managed	+ BYOC
TDE Encryption Keys	Service managed	+ Customer Managed Keys
Patching Window and Frequency	Service managed	User-specified date and time
Access to Standby database	No	Yes, Standby available for read/write

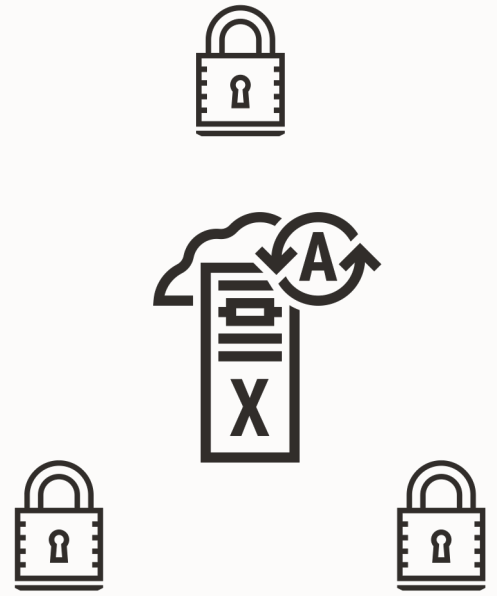
The most secured Oracle Database

Autonomous Database has **all major compliance certifications**

- FedRAMP High, HIPAA, ISO/IEC, PCI DSS, SOC

More secure than on-premises, and other Database Cloud Services

- Always encrypted, always audited, always patched
- Security at all levels: network, OS, DB
- No highly privileged access
  - No database SYS or SYSDBA
  - No login allowed to OS or CDB
  - Prevents installing or modifying any software on system



Copyright © 2023, Oracle and/or its affiliates

Summary

Self-securing



Oracle's most secure Database

- Automate patching, encrypt data-at-rest and in the network, audit user activity

Next Level



Proven protection reduces risk

- Proven security controls for Autonomous Databases

Reduce Risk



Includes security features for increased protection

- Data Safe, Privilege Analysis, Oracle Database Vault, Data Redaction, Oracle Label Security, Virtual Private Database, Real Application Security.

Thank you

Any Questions?

Sandesh Rao
VP AIOps for the Autonomous Database

@sandeshr
https://www.linkedin.com/in/raosandesh/
https://www.slideshare.net/SandeshRao4

