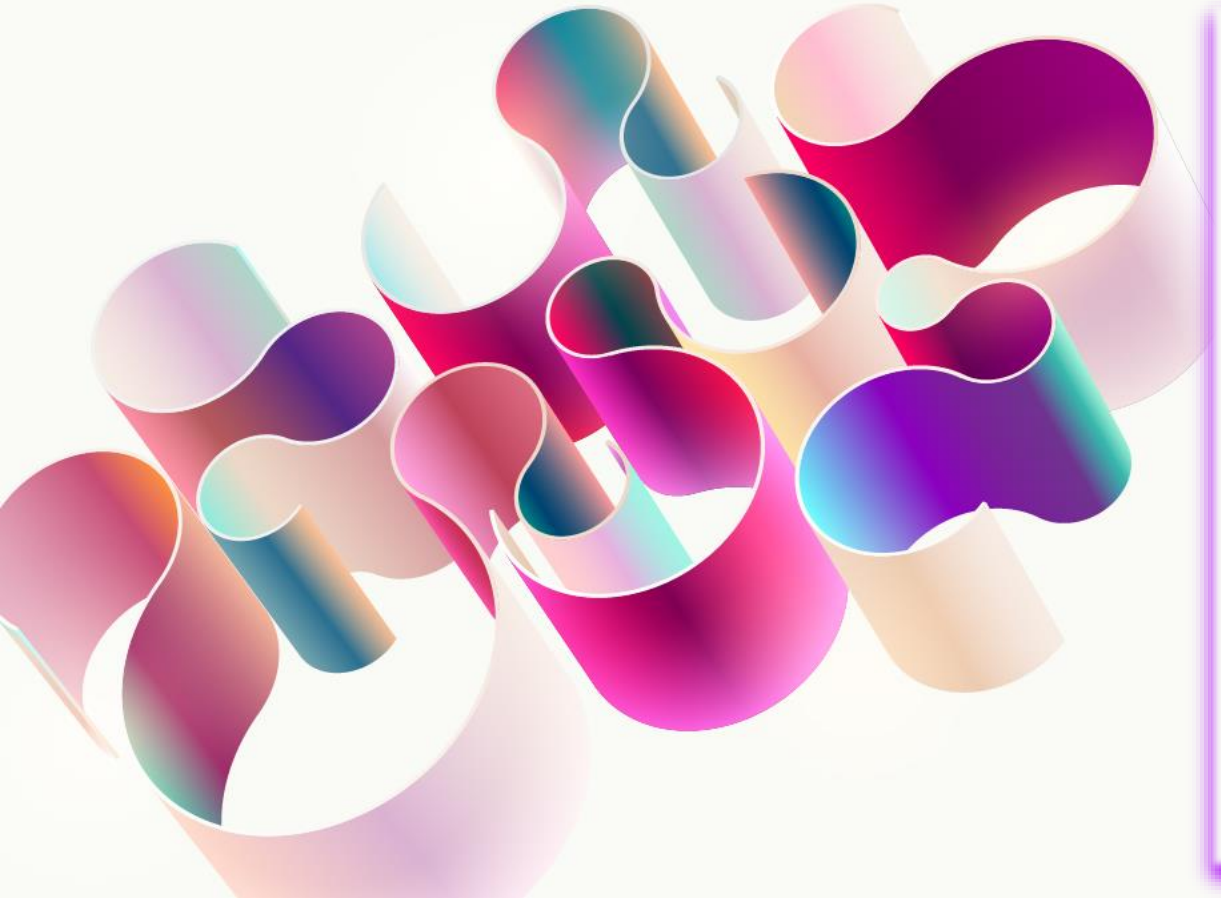
APEX connect by DOAG



# DBMS_CLOUD on premises: S3 Buckets direkt anzapfen

**Dienstag, 23.04.2024 | 13:00 - 13:45 | Amsterdam**

**Robert Marz**

DATABEE
Die IT-Architekten

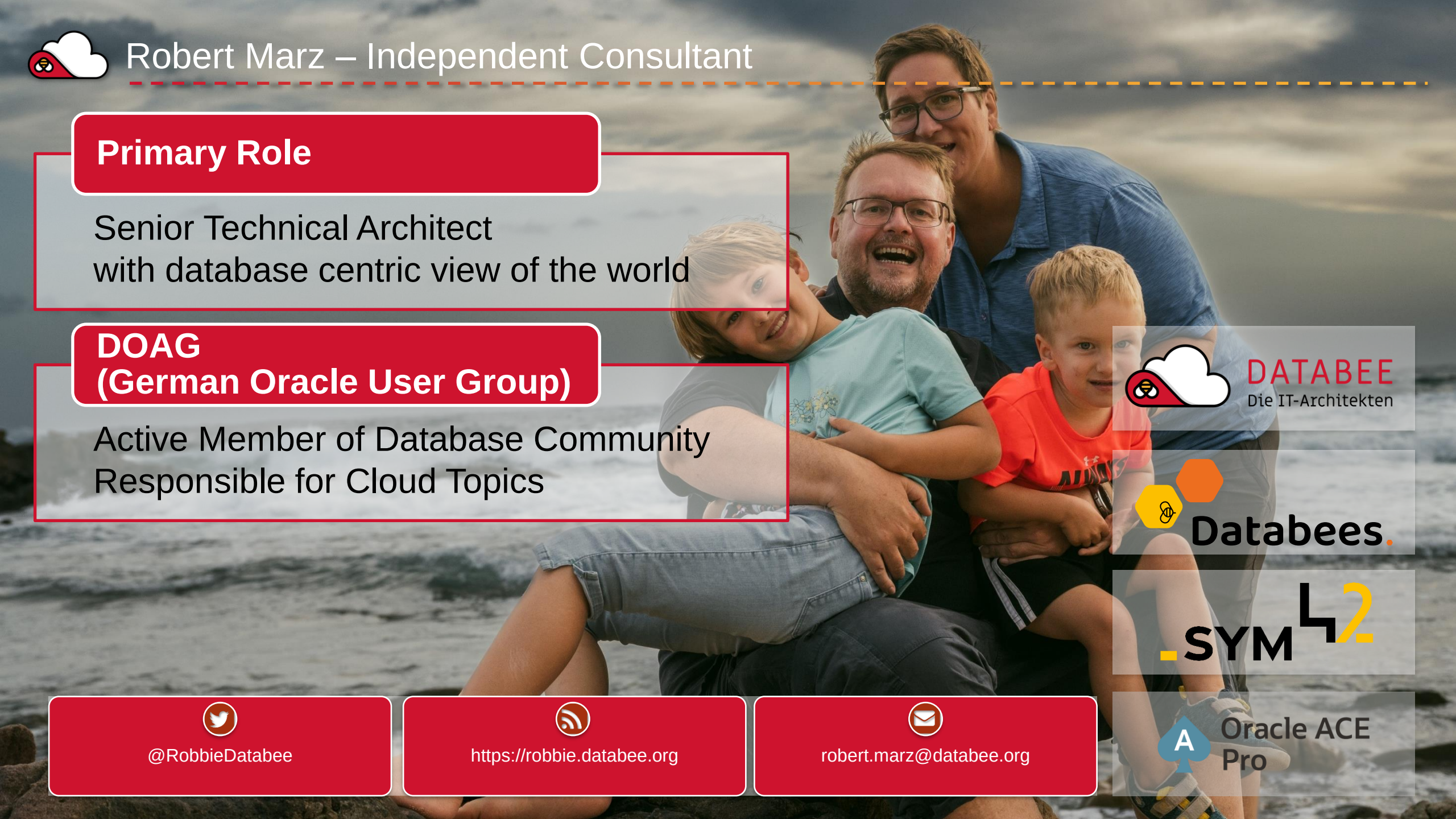
# Robert Marz – Independent Consultant

## Primary Role

Senior Technical Architect
with database centric view of the world

## DOAG (German Oracle User Group)

Active Member of Database Community
Responsible for Cloud Topics



DATABEE
Die IT-Architekten

Databees.

SYM42

Oracle ACE Pro


@RobbieDatabee

https://robbie.databee.org

robert.marz@databee.org

Oracle ACE Program

# 500+ technical experts helping peers globally

The **Oracle ACE Program** recognizes and rewards community members for their technical and community contributions to the Oracle community

## 3 membership tiers

Oracle ACE Director | Oracle ACE Pro | Oracle ACE Associate

Oracle ACE

## Nominate

**yourself or someone you know:**

ace.oracle.com/nominate

For more details on Oracle ACE Program:
ace.oracle.com

Connect: aceprogram_ww@oracle.com Facebook.com/OracleACEs @oracleace

# .SYMPOSIUM 42

Created by the community, to support the community

Sharing of reliable knowledge
Supporting the various user groups and individuals

@sym_42

https://sym42.org/


# Buckets & Object Stores

# Understanding Object Storage

## Definition of Object Storage

Stores data as objects within a flat namespace, unlike traditional file systems.

Each object includes the data itself, a globally unique ID, and metadata.

## Components

**Objects:**

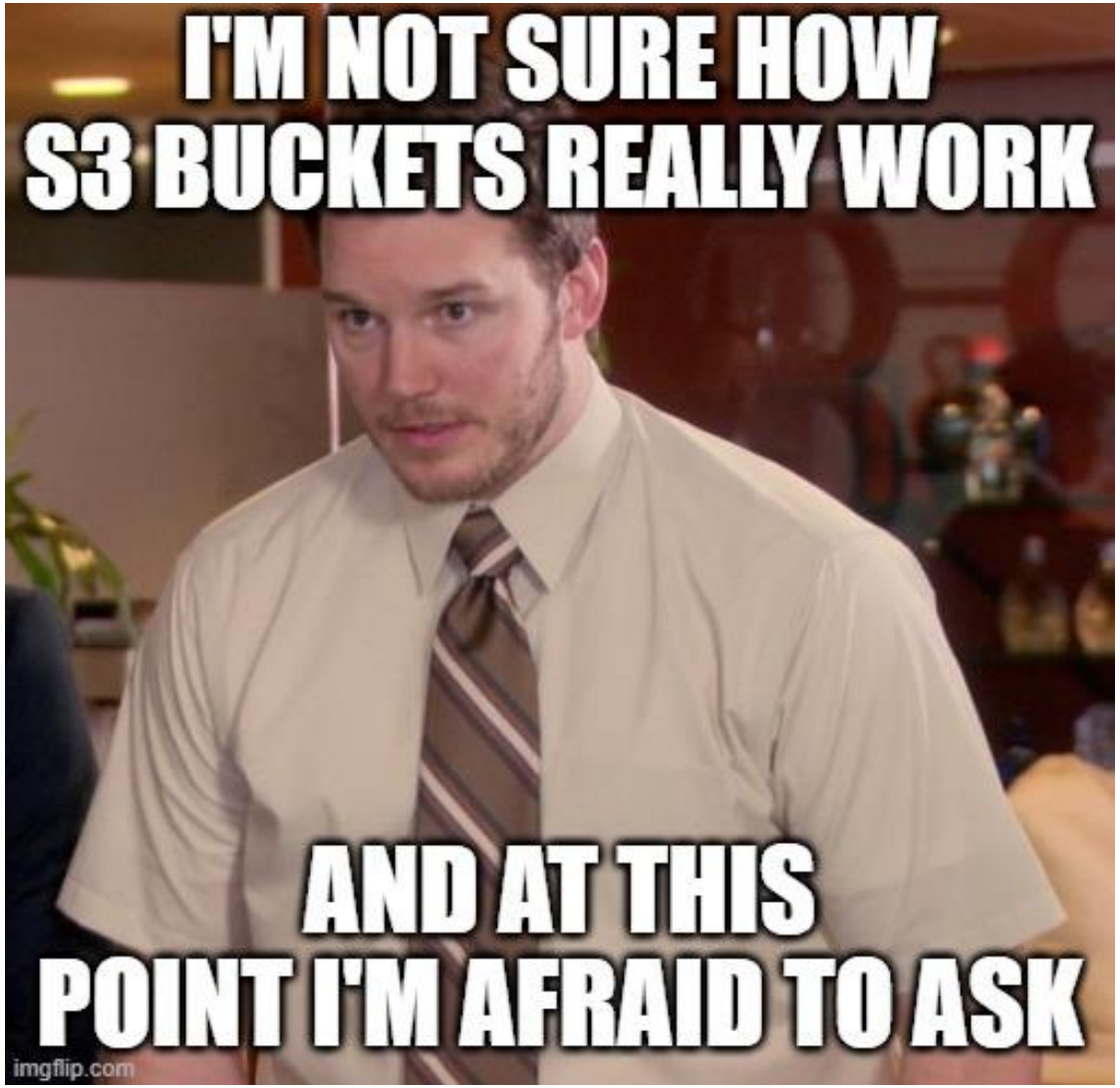
- The individual data items that are stored.

**Buckets:**

- Containers that hold objects; similar to directories but not hierarchical.

## Access Method

Typically accessed through RESTful APIs using standard HTTP methods.



# Characteristics of Object Storage

## Data Immutability

- Once an object is written, it cannot be changed (immutable); any changes require writing a new object version.

## Metadata

- Objects can have extensive metadata in addition to the data they store, which can include custom attributes.

## Scalability

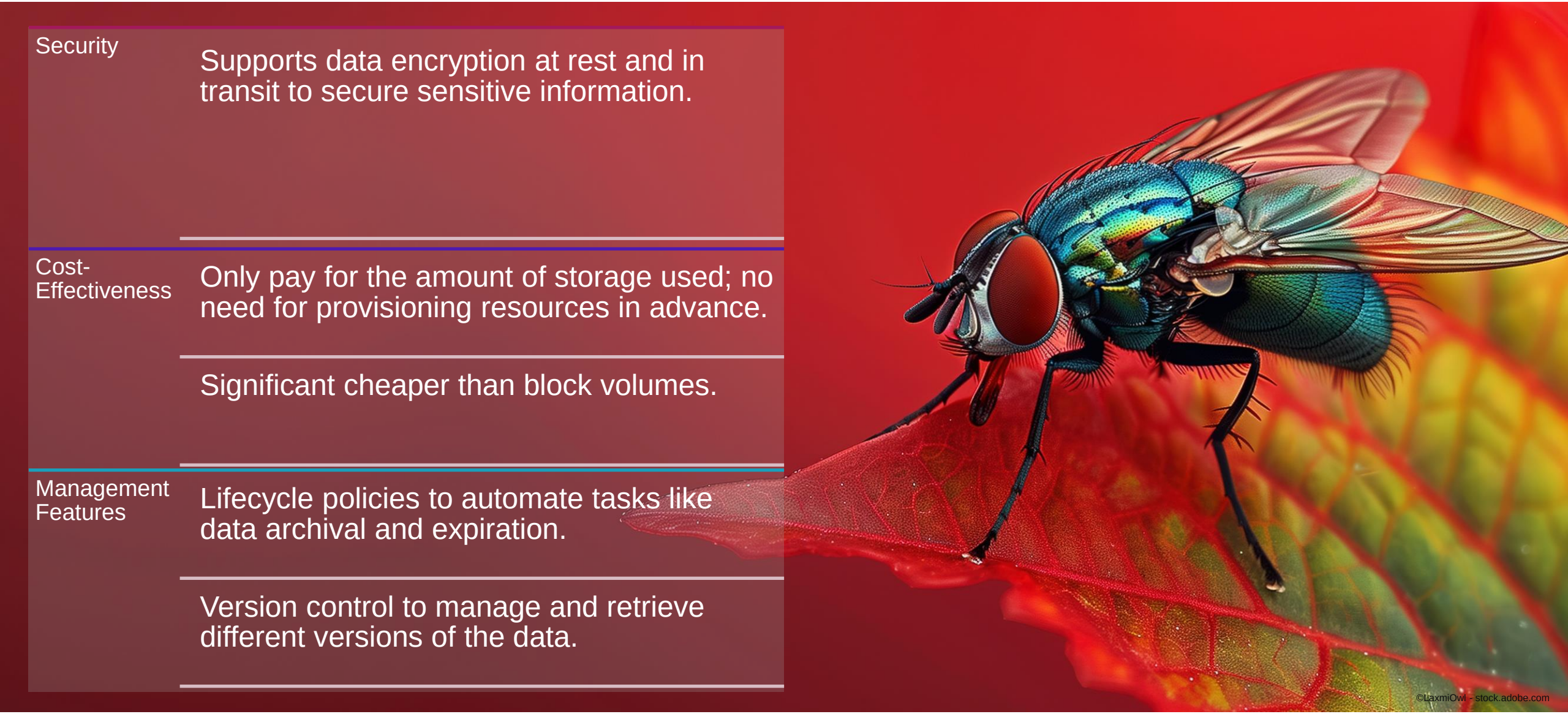
- Designed for scalability; can store an enormous amount of data with little performance impact.

## Durability

- High durability (e.g., Amazon S3 offers 99.999999999% durability), ensuring data is replicated and preserved.

# Features of Object Storage

| | |
|---|---|
| Security | Supports data encryption at rest and in transit to secure sensitive information. |
| Cost-Effectiveness | Only pay for the amount of storage used; no need for provisioning resources in advance. |
| | Significant cheaper than block volumes. |
| Management Features | Lifecycle policies to automate tasks like data archival and expiration. |
| | Version control to manage and retrieve different versions of the data. |



# Use Cases of Object Storage

**Backup and Disaster Recovery**

Ideal for offsite backups because of its durability and availability.

**Hybrid and Multi-cloud Environments**

Works well in hybrid setups, allowing data movement between on-premises data centers and the cloud or across multiple cloud environments.

**Content and Data Lakes**

Central repository for aggregating data from various sources

**Content Distribution**

Efficient at serving web content directly to users (e.g., images, videos, and static files).

# Amazon S3 Compatible Objects Store Providers

## Amazon Web Services (AWS) - Amazon S3

- **Service Name**: Amazon Simple Storage Service (S3)
- **Description**: The original service that popularized the S3 API, offering highly scalable, reliable, and low-latency data storage infrastructure.
- **URL**: Amazon S3

## Oracle Cloud Infrastructure (OCI) - Oracle Object Storage

- **Service Name**: Oracle Object Storage
- **Description**: Offers an S3-compatible API and is designed to handle large amounts of unstructured data with high durability.
- **URL**: Oracle Object Storage

## Google Cloud Platform (GCP) - Google Cloud Storage

- **Service Name**: Google Cloud Storage
- **Description**: Compatible with the S3 API via the use of adapters and offers multi-regional support, object lifecycle management, and a variety of storage classes.
- **URL**: Google Cloud Storage

## Microsoft Azure - Azure Blob Storage

- **Service Name**: Azure Blob Storage
- **Description**: Offers an S3-compatible API through Azure Blob Storage, facilitating easy migration and integration of S3 applications.
- **URL**: Azure Blob Storage

## Strato - HiDrive S3

- **Service Name**: HiDrive S3
- **Description**: German Datacenters, DSGVO (German GDPR) compliant, fast, secure.
- **URL**: Strato HiDrive S3

# DBMS_Cloud – Background & History

## SAAS / PAAS DBs like Oracle Autonomous

- don‘t have access to Filesystem
  - → No Datapump
  - → No External Tables
- → Exchange Data only via SQL-Statements and DB-Links or Applications

## DBMS_CLOUD

- was introduced with Autonomous DB in 2018
- shipped with Oracle 19.9
- continuously improved
  - latest enhancements July 2023
    - - preconfigured Object Store and REST Endpoints
    - - detectfieldorder
    - - dbms_cloud.export_data – Export Files as Text or Datapump

# DBMS_Cloud - Features

| | |
|---|---|
| Acces Management | • Credentials Handling |
| Data Handling | • Load data into tables<br>• Handle external tables<br>• (Export data) |
| File Handling | • Database Directory |
| Object Handling | • get / put / delete<br>• copy<br>• list |
| Autonomous only | • Bulk File and Object Handling<br>• Rest APIs<br>• Export Data<br>(CSV, JSON, Parquet, XML, DataPump) |

# DBMS_Cloud – Features: Data Handling

| | | |
|---|---|---|
| copy data into tables | like sqlloader | |
| | format options in JSON | |
| | type | csv with and without embedded |
| | | Avro / Parquet, datapump |
| external tables | partitioning | normal |
| | | hybrid |
| | Avro and parquet files | |
| | Validate | Data (Badfile) |
| | | Source FIles |
| export data (autonomous only) | Optional | encrypted |
| | | compressed |
| | type | CSV, JSON, XML |
| | | Datapump, parquet |

```
BEGIN
    DBMS_CLOUD.CREATE_CREDENTIAL(
        credential_name ⇒ 'OBJ_STORE_CRED',
        username        ⇒ 'user_name@oracle.com',
        password        ⇒ 'password'
        );

    DBMS_CLOUD.COPY_COLLECTION(
        table_name      ⇒ 'ORDERS',
        schema_name     ⇒ 'TEST_SCHEMA',
        credential_name ⇒ 'OBJ_STORE_CRED',
        file_uri_list   ⇒ 'https://objectstorage.us-phoenix-1.oraclecloud
        format          ⇒  json_object('ignoreblanklines' value TRUE,
                                       'rejectlimit' value '0',
                                       'dateformat' value 'yyyy-mm-dd',
                                       'regexuri' value TRUE)
            );
END;
/
```

# DBMS_Cloud – Features: File- and Object Management

## File Handling

- Database Directories
- delete
- list (DBFS and Oracle Files System only)

## Object Handling

- mimics REST methods
- get / put / delete
- copy
- list

# Oracle DBMS_Cloud in Autonomous DB

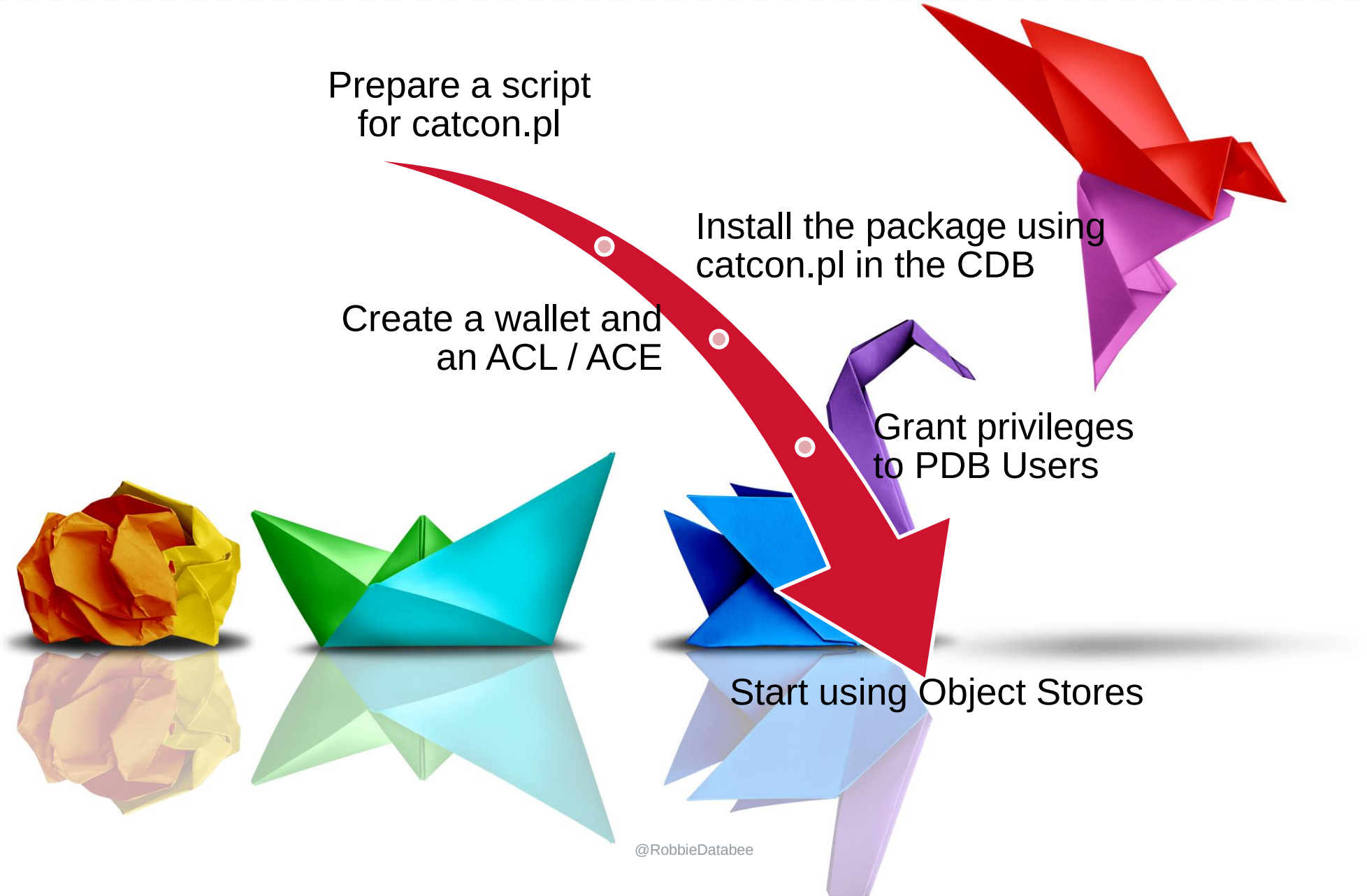

DBMS_CLOUD on premises
The rocky road
©nightman - stock.adobe.com

# DBMS_CLOUD on Premises – Prerequires

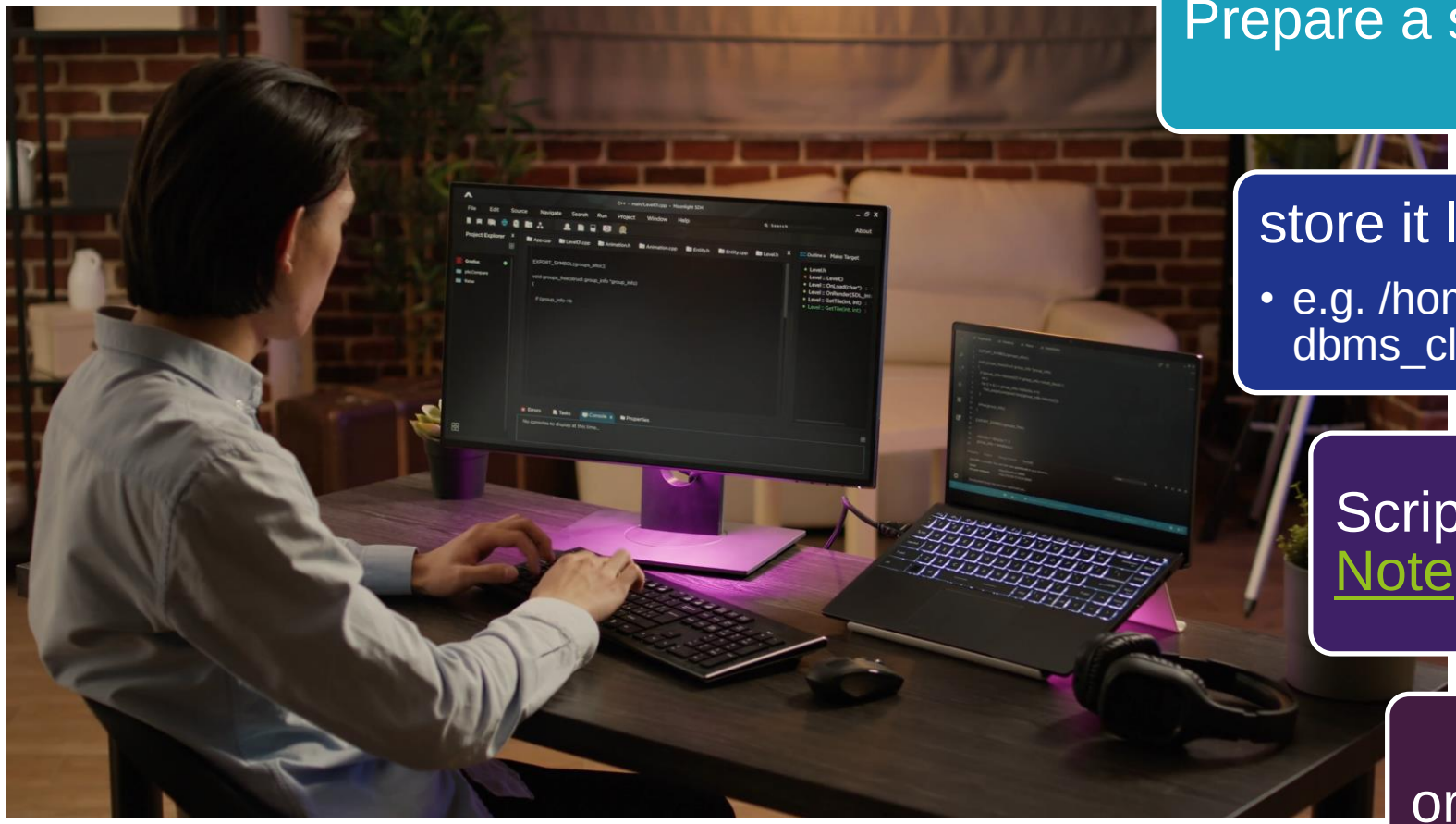
## Prerequires

| | |
|---|---|
| Oracle 19.9 or higher | Part of 19.9 RU |
| | Only CDB |
| Non-CDB since | 19.15 |
| | 21.3 |
| Installation | see Mos ID 2748362.1 |

# DBMS_CLOUD on Premises - Outline



# DBMS_CLOUD on Premesis – prepare a script for catcon.sql



Prepare a script

store it locally

- e.g. /home/oracle/dbc dbms_cloud_install.sql

Script contents in MOS Note

or download it here

# DBMS_CLOUD on Premesis – script dbms_cloud_install.sql 01

```
@$ORACLE_HOME/rdbms/admin/sqlsessstart.sql

set verify off
-- you must not change the owner of the functionality to avoid future issues
define username='C##CLOUD$SERVICE'

create user &username no authentication account lock;
```

# DBMS_CLOUD on Premesis – script dbms_cloud_install.sql 02

```
REM Grant Common User Privileges
grant INHERIT PRIVILEGES on user &username to sys;
grant INHERIT PRIVILEGES on user sys to &username;
grant RESOURCE, UNLIMITED TABLESPACE, SELECT_CATALOG_ROLE to &username;
grant CREATE ANY TABLE, DROP ANY TABLE, INSERT ANY TABLE, SELECT ANY TABLE,
CREATE ANY CREDENTIAL, CREATE PUBLIC SYNONYM, CREATE PROCEDURE, ALTER SESSION, CREATE JOB to &
username;
grant CREATE SESSION, SET CONTAINER to &username;
grant SELECT on SYS.V_$MYSTAT to &username;
grant SELECT on SYS.SERVICE$ to &username;
grant SELECT on SYS.V_$ENCRYPTION_WALLET to &username;
grant read, write on directory DATA_PUMP_DIR to &username;
grant EXECUTE on SYS.DBMS_PRIV_CAPTURE to &username;
grant EXECUTE on SYS.DBMS_PDB_LIB to &username;
grant EXECUTE on SYS.DBMS_CRYPTO to &username;
grant EXECUTE on SYS.DBMS_SYS_ERROR to &username;
grant EXECUTE ON SYS.DBMS_ISCHED to &username;
grant EXECUTE ON SYS.DBMS_PDB_LIB to &username;
grant EXECUTE on SYS.DBMS_PDB to &username;
grant EXECUTE on SYS.DBMS_SERVICE to &username;
grant EXECUTE on SYS.DBMS_PDB to &username;
grant EXECUTE on SYS.CONFIGURE_DV to &username;
grant EXECUTE on SYS.DBMS_SYS_ERROR to &username;
grant EXECUTE on SYS.DBMS_CREDENTIAL to &username;
```

```
grant EXECUTE on SYS.DBMS_RANDOM to &username;
grant EXECUTE on SYS.DBMS_SYS_SQL to &username;
grant EXECUTE on SYS.DBMS_LOCK to &username;
grant EXECUTE on SYS.DBMS_AQADM to &username;
grant EXECUTE on SYS.DBMS_AQ to &username;
grant EXECUTE on SYS.DBMS_SYSTEM to &username;
grant EXECUTE on SYS.SCHED$_LOG_ON_ERRORS_CLASS to &username;
grant SELECT on SYS.DBA_DATA_FILES to &username;
grant SELECT on SYS.DBA_EXTENTS to &username;
grant SELECT on SYS.DBA_CREDENTIALS to &username;
grant SELECT on SYS.AUDIT_UNIFIED_ENABLED_POLICIES to &username;
grant SELECT on SYS.DBA_ROLES to &username;
grant SELECT on SYS.V_$ENCRYPTION_KEYS to &username;
grant SELECT on SYS.DBA_DIRECTORIES to &username;
grant SELECT on SYS.DBA_USERS to &username;
grant SELECT on SYS.DBA_OBJECTS to &username;
grant SELECT on SYS.V_$PDBS to &username;
grant SELECT on SYS.V_$SESSION to &username;
grant SELECT on SYS.GV_$SESSION to &username;
grant SELECT on SYS.DBA_REGISTRY to &username;
grant SELECT on SYS.DBA_DV_STATUS to &username;
```

# DBMS_CLOUD on Premesis – script dbms_cloud_install.sql 03

```
alter session set current_schema=&username;
REM Create the Catalog objects
@$ORACLE_HOME/rdbms/admin/dbms_cloud_task_catalog.sql
@$ORACLE_HOME/rdbms/admin/dbms_cloud_task_views.sql
@$ORACLE_HOME/rdbms/admin/dbms_cloud_catalog.sql
@$ORACLE_HOME/rdbms/admin/dbms_cloud_types.sql
```

# DBMS_CLOUD on Premesis – script dbms_cloud_install.sql 04

```
REM Create the Package Spec
@$ORACLE_HOME/rdbms/admin/prvt_cloud_core.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_task.plb
@$ORACLE_HOME/rdbms/admin/dbms_cloud_capability.sql
@$ORACLE_HOME/rdbms/admin/prvt_cloud_request.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_internal.plb
@$ORACLE_HOME/rdbms/admin/dbms_cloud.sql
@$ORACLE_HOME/rdbms/admin/prvt_cloud_admin_int.plb

REM Create the Package Body
@$ORACLE_HOME/rdbms/admin/prvt_cloud_core_body.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_task_body.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_capability_body.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_request_body.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_internal_body.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_body.plb
@$ORACLE_HOME/rdbms/admin/prvt_cloud_admin_int_body.plb
```

# DBMS_CLOUD on Premesis – script dbms_cloud_install.sql 05

```
-- Create the metadata
@$ORACLE_HOME/rdbms/admin/dbms_cloud_metadata.sql

alter session set current_schema=sys;

@$ORACLE_HOME/rdbms/admin/sqlsessend.sql
```

# DBMS_CLOUD on Premesis – Install the package

```
$ORACLE_HOME/perl/bin/perl \
    $ORACLE_HOME/rdbms/admin/catcon.pl \
        -u sys/<your_sys_password> \
        --force_pdb_mode 'READ WRITE' \
        -b dbms_cloud_install \
        -d /home/oracle/dbc \
        -l /home/oracle/dbc \
        dbms_cloud_install.sql
```

```
catcon.pl parameter:
-u = user name, usually SYS

-b = Base name for log files, can be any meaningful name

-d = directory where the script file to be run is located.

-l - directory to use for spool log files
```

# DBMS_CLOUD on Premesis – package validation

```
REM from within ROOT to see all containers
select con_id, owner, object_name
     , status, sharing, oracle_maintained
  from cdb_objects
 where object_name = 'DBMS_CLOUD'
 order by con_id;

REM within an individual container only
select owner, object_name
     , status, sharing, oracle_maintained
  from dba_objects
 where object_name = 'DBMS_CLOUD';
```

# DBMS_CLOUD on Premesis: Updating


## After update to RU with new DBMS_CLOUD deployment

- Re-run the installation procedure on top of your existing procedure
- The installation is idempotent, no need to uninstall and re-install

# DBMS_CLOUD on Premesis – create wallet for https access

**https access**

- requires a wallet with CA Certs

**Oracle does not ship certs as part of RUs**

- Download and unpack dbc_certs.tar

**Create new wallet**

- or use existing

**Add certificates to wallet**

**Tell Oracle to use the wallet**

- add to sqlnet.ora

```
#! /bin/bash

# This is our wallet dir - choose your own location
walletdir="/opt/oracle/dcs/commonstore/wallets/ssl"
password=Welcome1    # Wallet password - choose your own

orapki wallet create        \
    -wallet "$walletdir"    \
    -pwd "$(password)"      \
    -auto_login
# unpack the certs
mkdir -p ./dbc_certs; cd $_; tar -xvf dbc_certs.tar
for i in 'ls ./dbc_certs/*cer'
do
    orapki wallet add          \
        -wallet "$walletdir"   \
        -trusted_cert          \
        -cert $i               \
        -pwd $password
done
```

```
#add to $ORACLE_HOME/network/admin/sqlnet.ora
WALLET_LOCATION=(
    SOURCE=(METHOD=FILE)
    (METHOD_DATA=
        (DIRECTORY=/opt/oracle/dcs/commonstore/wallets/ssl)
    )
)
```

# DBMS_CLOUD on Premesis – Create Access Control Entries (ACEs)

## Create file dbc_aces.sql

for proxy use, see MOS note.

## Run script in cdb

```
conn / as sysdba
@@./dbc_aces.sql
```

```
@$ORACLE_HOME/rdbms/admin/sqlsessstart.sql
-- you must not change the owner of the functionality to avoid future issues
define clouduser=C##CLOUD$SERVICE
-- CUSTOMER SPECIFIC SETUP, NEEDS TO BE PROVIDED BY THE CUSTOMER
-- - SSL Wallet directory
define sslwalletdir=/opt/oracle/dcs/commonstore/wallets/ssl
-- Create New ACL / ACE s
begin
    -- Allow all hosts for HTTP/HTTP_PROXY
    dbms_network_acl_admin.append_host_ace(
        host =>'*',
        lower_port => 443,
        upper_port => 443,
        ace => xs$ace_type(
            privilege_list => xs$name_list('http', 'http_proxy'),
            principal_name => upper('&clouduser'),
            principal_type => xs_acl.ptype_db)
    );
    -- Allow wallet access
    dbms_network_acl_admin.append_wallet_ace(
        wallet_path => 'file:&sslwalletdir',
        ace => xs$ace_type(privilege_list =>
            xs$name_list('use_client_certificates',
                         'use_passwords'),
        principal_name => upper('&clouduser'),
        principal_type => xs_acl.ptype_db));
    end;
end;
/
@$ORACLE_HOME/rdbms/admin/sqlsessend.sql
```

# DBMS_CLOUD on Premesis – PDB: Role CLOUD_USER

**DBMS_CLOUD is invokers right**

Grant rights to role or user

```
set verify off
-- target sample role
define userrole='CLOUD_USER'
-- target sample user
define username='SCOTT'
create role &userrole;
grant cloud_user to &username;

REM the following are minimal privileges to use DBMS_CLOUD
REM - this script assumes core privileges
REM - CREATE SESSION
REM - Tablespace quote on the default tablespace for a user
REM for creation of external tables,
REM    e.g. DBMS_CLOUD.CREATE_EXTERNAL_TABLE()
grant CREATE TABLE to &userrole;

REM for using COPY_DATA()
REM - Any log and bad file information is written into
REM  this directory
grant read, write on directory DATA_PUMP_DIR to &userrole;

REM
grant EXECUTE on dbms_cloud to &userrole;
```

# DBMS_CLOUD on Premesis – PDB: ACE

## ACL / ACE

Add role cloud_user to ACE

```
@$ORACLE_HOME/rdbms/admin/sqlsessstart.sql
-- target sample role
define cloudrole=CLOUD_USER

-- CUSTOMER SPECIFIC SETUP, NEEDS TO BE PROVIDED BY THE CUSTOMER
-- - SSL Wallet directory
define sslwalletdir=<Set SSL Wallet Directory>

begin
-- Allow all hosts for HTTP/HTTP_PROXY
dbms_network_acl_admin.append_host_ace(
    host =>'*',
    lower_port => 443,
    upper_port => 443,
    ace => xs$ace_type(
    privilege_list => xs$name_list('http', 'http_proxy'),
    principal_name => upper('&cloudrole'),
    principal_type => xs_acl.ptype_db)
);

-- Allow wallet access
dbms_network_acl_admin.append_wallet_ace(
    wallet_path     => 'file:&sslwalletdir',
    ace         => xs$ace_type(privilege_list =>
                    xs$name_list('use_client_certificates',
                                 'use_passwords'),
    principal_name  => upper('&cloudrole'),
    principal_type  => xs_acl.ptype_db)
);
end;
/

@$ORACLE_HOME/rdbms/admin/sqlsessend.sql
```

# Setting up DBMS_Cloud on premesis

# Further Reading


**Oracle 19c PL/SQL Packages and Types Reference**

- https://docs.oracle.com/en/database/oracle/oracle-database/19/arpls/DBMS_CLOUD.html

**DBMS_CLOUD in Amazon RDS Custom**

- https://aws.amazon.com/de/blogs/database/use-the-dbms_cloud-package-in-amazon-rds-custom-for-oracle-for-direct-amazon-s3-integration/

**DBMS_CLOUD Installation on 19c and 21c On-Prem Databases (Oracle Base by Tim Hall)**

- https://oracle-base.com/articles/21c/dbms_cloud-installation

**Using Oracle Autonomous Database Serverless**

- https://docs.oracle.com/en/cloud/paas/autonomous-database/serverless/adbsb/index.html

# DBMS_CLOUD

- Very Useful
- Not only in the Cloud
- Installation unnecessary complex
- Worth the hassle