ORACLE

Oracle Database 23 – the next Oracle database long-term release

April 25, 2024

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

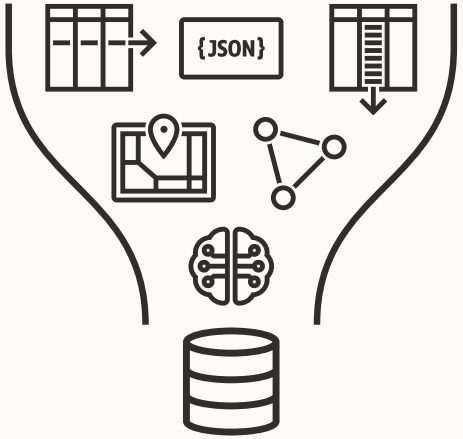
The materials in this presentation pertain to Oracle Health, Oracle, Oracle Cerner, and Cerner Enviza which are all wholly owned subsidiaries of Oracle Corporation. Nothing in this presentation should be taken as indicating that any decisions regarding the integration of any EMEA Cerner and/or Enviza entities have been made where an integration has not already occurred.

Make it easy to develop and run modern apps and analytics for all use cases at any scale

Oracle Database Vision

How We Deliver the Vision –

Complete and Simple Platform for All Data Management Needs



Converged Database

Complete support for all modern data types, workloads, and development styles

Completely consistent, scalable, available, and secure

Autonomous Database

Converged DB delivered as a self-driving, self-securing, self-repairing Cloud Database

Simplest DB for developing and running **any** apps or analytics at **any** scale or criticality

Continual improvements in each release...

2013

12c

- Multitenant Architecture
- JSON Support
- In-Memory
- Application Continuity
- Online Tablespace Encryption
- Sharding

Long Term Support Release

2018

18c

- Active Directory Integration
- Sharded RAC
- Multitenant Snapshot Carousel
- Private Temporary Tables
- Mem-Optimized Row Store
- In-Memory for External Tables

Innovation Release

2019

19c

- High Speed Data Ingestion
- Active Data Guard DML Redirect
- Automatic Indexing
- Hybrid Partition Tables
- JSON Improvements
- Security Enhancements

Long Term Support Release

2021

21c

- Native Blockchain Tables
- AutoML
- Per-PDB Data Guard
- Native JSON Type
- SQL Macros
- JavaScript Stored Logic

Innovation Release

23c App Simple

The Next Long Term Support Release

- Continues to deliver converged database innovations for both cloud and on-premises
- Brings new functionality to simplify development of data-driven apps for any scale or criticality

Milestones

- **October 2022**
  Oracle **Database 23c Beta** code named "App Simple".
- **April 2023**
  Oracle **Database 23c Free** - Developer Release
- **September 2023**
  Oracle Database 23c for Oracle Base Database Service Updated **Oracle Database 23c Free**
- **1H CY2024**
  Availability on other platforms - cloud and on-premises

Now Available: Oracle Database 23c *Free*

Develop, Learn, and Run for Free

- Release 23.3: Updated in September
- Developers get a head-start on building apps with innovative 23c features
- Simplifies development of modern data-driven apps
- Permitted features: 16 PDBs, Oracle Database In-Memory, VPD, Advanced Security, Partitioning, Advanced Compression etc.

Oracle Database 23c Free Released

- Docker image, VirtualBox VM, Linux RPM file

Available under Free Use Terms & Conditions license

- Download straight from the web - no user or login
- Capped database resources for 12GB storage, 2GB memory, and 2 CPU cores

https://www.oracle.com/database/free/

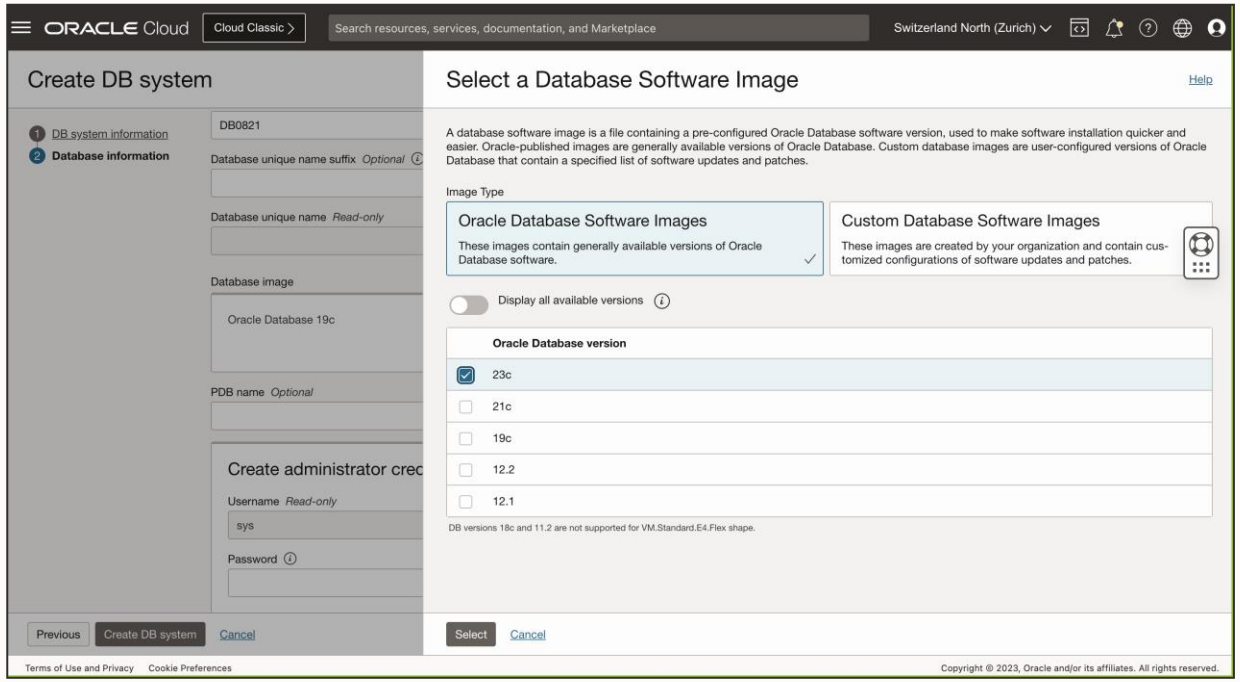
Now Available: Oracle Database 23c on OCI BaseDB

Develop, test, and deploy modern production database apps

- Available since September 2023
- Includes quick and easy-to-use cloud automation
- Try out various new features of Database 23c in the cloud
- Run Database 23c on latest Oracle Linux version 8
- Limitations: No Oracle Database Standard Edition, Multi-node RAC DB System, Upgrades to Oracle Database 23c from earlier versions …

Oracle Database 23c

- 23.3 on OCI as Base DB



Oracle Base Database Service Documentation

Announcing Oracle Database 23c – The next Long Term Support Release

New In 23c

SQL Domains

Schema Privileges

Oracle Database 23c App Simple

Read-Only Per-PDB Standby

Operational Property Graphs

Real-time SQL Plan Management

JSON Schema

Microservice Support

JSON Relational Duality

AI Vector Search

True Cache

SQL Firewall

Priority Transactions

Globally Distributed Database

Rolling Patching

JS Stored Procedures

Developer Role

Shrink Tablespace

Boolean Datatype

SQL:2023 Standard is available

© Gerd Altmann

- A joint committee of ISO and IEC standardizes SQL with the participation of national standardization bodies such as ANSI or DIN.
- The first SQL standard appeared in 1986/87
- SQL:2023 (exact designation ISO/IEC 9075-1:2023) as the successor to SQL:2016 is now available
- 11 parts are available for purchase in the ISO Store with the entirely new part 16, SQL/PGQ
  - It defines ways for the SQL language to represent and interact with property graphs.
- Oracle once again was actively involved in driving new features into the Standard

- **Many of these features are already supported in the Oracle Database 23c Free - Developer Release!**

SQL, SQL, SQL … SQL just got easier …

```
SQL> drop table if exists emails; -- suppress errors
Table dropped.
SQL> drop table if exists emails; -- suppress errors
Table dropped.
```

```
SQL> insert into emp_test (emp_id, emp_name)
     values (4,'WITOLD'),(5,'ULRIKE'), (6,'STEPHANE');
3 rows created.
```

```
SQL> with t1(e1, e2) as (values
       (1,'SCOTT'), (2,'SMITH'), (3,'JOHN'))
     select * from t1;

        E1 E2
---------- -----
         1 SCOTT
         2 SMITH
         3 JOHN
```

```
SQL> create table emp1
    (id number, first_name varchar2(30) default 'test');
Table created.
SQL> update emp1 set first_name = default where id=1;
1 row updated.
```

```
SQL> select sysdate;
SYSDATE
---------
13-FEB-24
```

```
SQL> delete emp e from dept d
     where e.deptno=d.deptno and d.dname ='RESEARCH';
5 rows deleted.
```

More Oracle Database Data Types

Built-In

- CHAR(n), VARCHAR2(n),
- NUMBER, BINARY_FLOAT, BINARY_DOUBLE, FLOAT
- DATE, TIMESTAMP, INTERVAL_YEAR
- ROWID, UROWID
- LONG, RAW, LONG RAW
- BLOB, CLOB, NCLOB, BFILE, SECUREFILE
- JSON
- BOOLEAN
- VECTOR (*)

(*) coming soon

Supplied

- Any Types : ANYTYPE …
- XMLTYPE
- URI Types: HTTPURIType …
- Spatial: SDO_GEOMETRY SDO_GEORASTER SDO_TOPO_GEOMETRY

User-Defined

- Object Types
- VARRAY (Varying Array)
- Nested Table

SQL Language Reference: Data Types

Data Types: JSON, BOOLEAN, VECTOR (*)

- JSON data can be stored in textual or with JSON data type - the native Binary JSON storage

```
create table emp (empno     number,
                  first     varchar2(50),
                  last      varchar2(50),
                  salary    number,
                flex BLOB,  check(flex IS JSON));
```

```
create table emp (empno     number,
                  first     varchar2(50),
                  last      varchar2(50),
                  salary    number,
                  flex      JSON);
```

- Native BOOLEAN data type with operations on Booleans that return Booleans: NOT, AND, OR …

```
create table if not exists emails (email varchar2(500), active BOOOLEAN);
```

```
select * from emails where active;
```

- Natively implemented AI Vector Search in Oracle Database with VECTOR data type, operations, and index

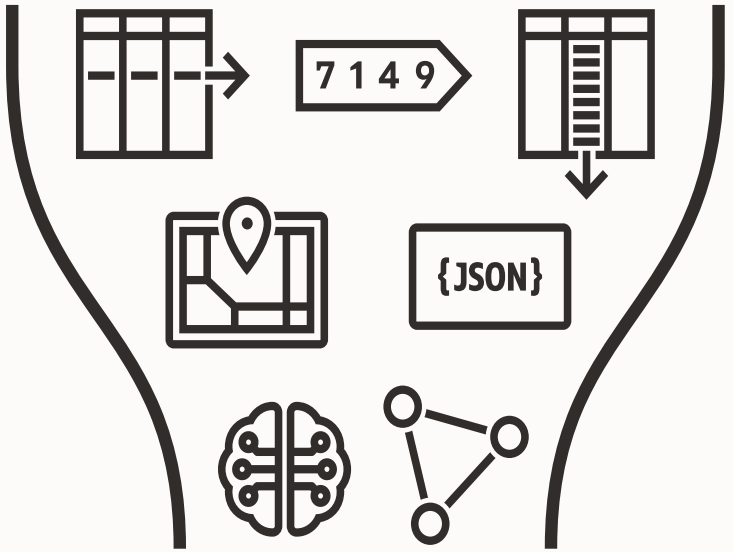
```
create table jobs_postings (id number, title varchar2(100), sal number, loc varchar2(100), job_vec VECTOR);
```

```
select * from job_postings
where title = 'Software Engineer' and loc = 'New York'
order by VECTOR_DISTANCE(job_vec, :resume_vector)
fetch first 10 rows only;
```

(*) coming soon

Oracle's Converged Database

- **Multi-model**
- **Multi-workload**
- **Multi-tenant**




```
select p.id, p.name,
   sl.json_doc.quantity amount
from product p, customer c, store s, sales sl
where
sl.json_value(json_doc,'$.product_id')= p.id
and
contains(p.notes,fuzzy('shipng costs'), 1) > 0
and
sl.json_value(json_doc,'$.CUSTNumber' returning
number) = c.cust_ids
and
s.store_id = 1234
and
sdo_within_distance(c.location,>
s.location,'distance=20')='TRUE'
order by
vector_distance(prod_vector, :input_vector);(*)
```

JSON
TEXT
SPATIAL
VECTOR (*) coming soon

Multi-Model

Enterprise Grade Operations and Security by Consolidating to Database As A Service

Cross Model

Consistent "Holistic View" of Real Time Production Data

Enhanced Oracle Converged Database in 23

JSON-Relational Duality

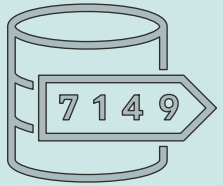
Native Property Graph in the database

Oracle Spatial

Blockchain Table

Javascript in the database

Ubiquitous Database Search



Native AI Vector Search (*)

(*) coming soon

New | Oracle Database 23c JSON Relational Duality

Storage Format

first_name	order	quantity
Jake	{ “product” : ”TV”, “price” : 250 }	1

Access Formats

```
{
 "first_name" : “Jake",
 "order" :
    {
        "product" : "TV",
        "price" :  250
    },
 "quantity" : 1
}
```

JSON Relational Duality delivers the simplicity of JSON for accessing data plus the simplicity of relational for storing and manipulating data

New | Oracle Database 23c Operational Property Graphs

- **First** commercial database to implement SQL/PGQ standard
- Enables easy use of graph analytics in **transactional** systems

Before SQL Property Graph

```
-- Snippet from a 50+ Line SQL Query
SELECT "v1.id" AS "THOM_ID", "v2.id" AS "LARRY_ID"
FROM (SELECT "V1"."ID" AS "v1.id", "V2"."ID" AS "v2.id"
FROM "GRAPHUSER"."USERS" "V1", "GRAPHUSER"."USERS" "V2", (WITH t1(src_table, src_key,
         dst_table, dst_key,
         exp, lvl)
AS (
  -- Anchor member.
  SELECT "src_table" AS src_table, "src_key" AS src_key, "src_table" AS dst_table, "src_key" AS dst_key,
  '' AS exp, 0 AS lvl   FROM(SELECT 'USERS' AS "src_table", v1.id AS "src_key"
FROM "GRAPHUSER"."USERS" "V1"
WHERE ("V1"."ID" = 1))
  UNION ALL
  -- Recursive member.
  SELECT t1.src_table, t1.src_key,
        t2."dst_table" AS dst_table, t2."dst_key" AS dst_key,
        t1.exp || t2."exp" AS exp,
        t1.lvl + 1 AS lvl
    FROM (SELECT 'USERS' AS "src_table", "anonymous_1".follower_id
    AS "src_key", 'USERS' AS "dst_table", "anonymous_1".followed_id AS "dst_key", '' AS "edge_table",
    '' AS "edge_key", (('<EXPRESSIONS>' || '') || '</EXPRESSIONS>') AS "exp"
FROM "GRAPHUSER"."FOLLOWS" "anonymous_1"
WHERE (NOT("anonymous_1"."FOLLOWER_ID" IS NULL) AND NOT("anonymous_1"."FOLLOWED_ID" IS NULL))) t2, t1
... 40+ lines
```

SQL Property Graph

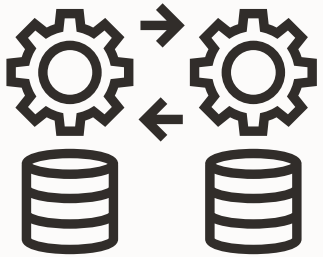
```
SELECT DISTINCT Thom_ID, Harry_ID from graph_table(social_graph
MATCH (v1)-[IS follows]->{1,5}(v2)
WHERE v1.id = 1 AND v2.id = 15
COLUMNS (v1.id AS Thom_ID, v2.id AS Harry_ID));
```

New | Oracle Database 23c (23.3) for the Developer

SQL Enhancements

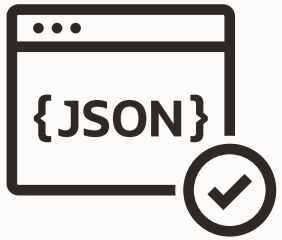
SQL Domains

Lock-free Column Value Reservations



Microservices Support

JavaScript Development



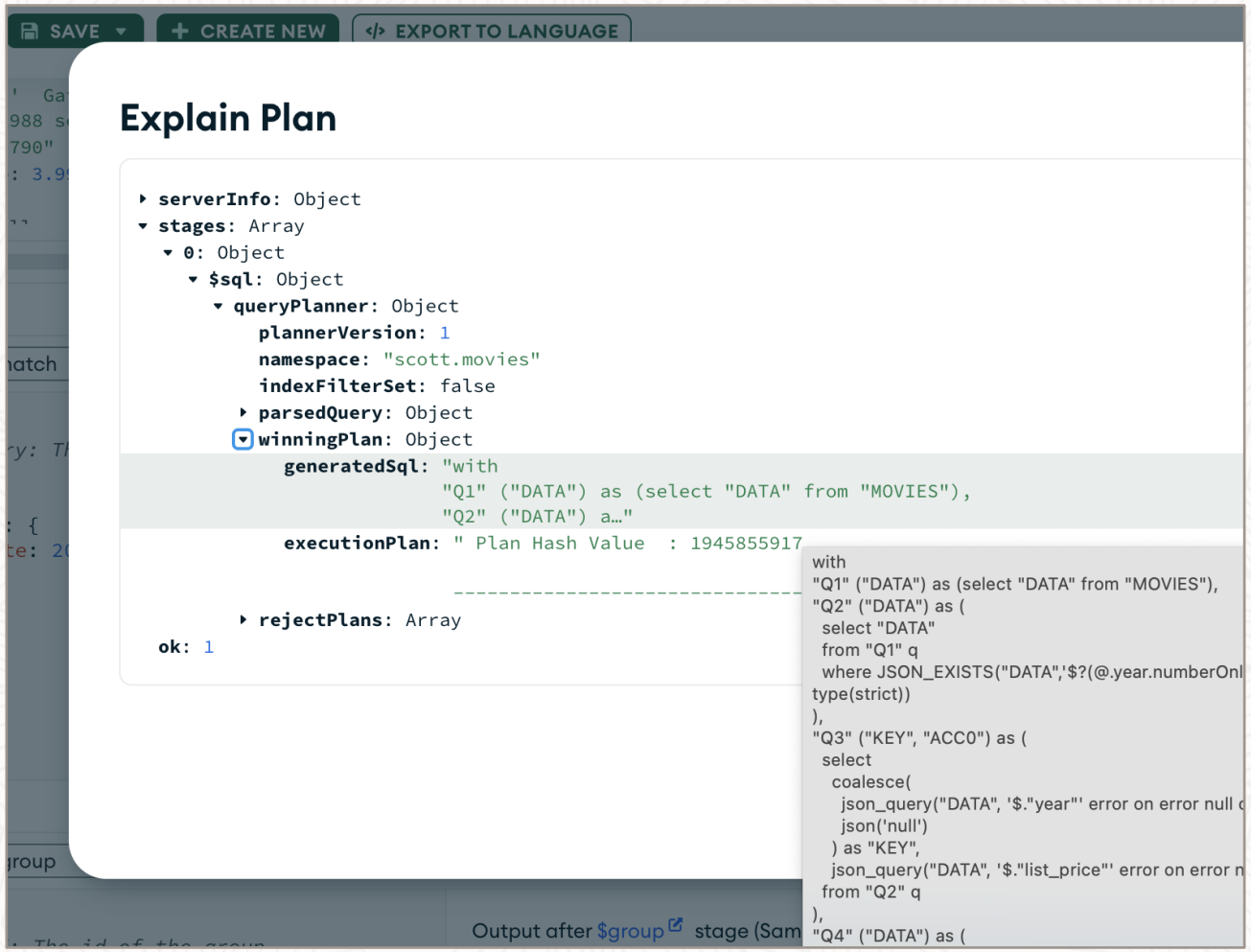
JSON Schema

New | Oracle Database 23c (23.3) MongoDB API Support for Aggregation Pipelines

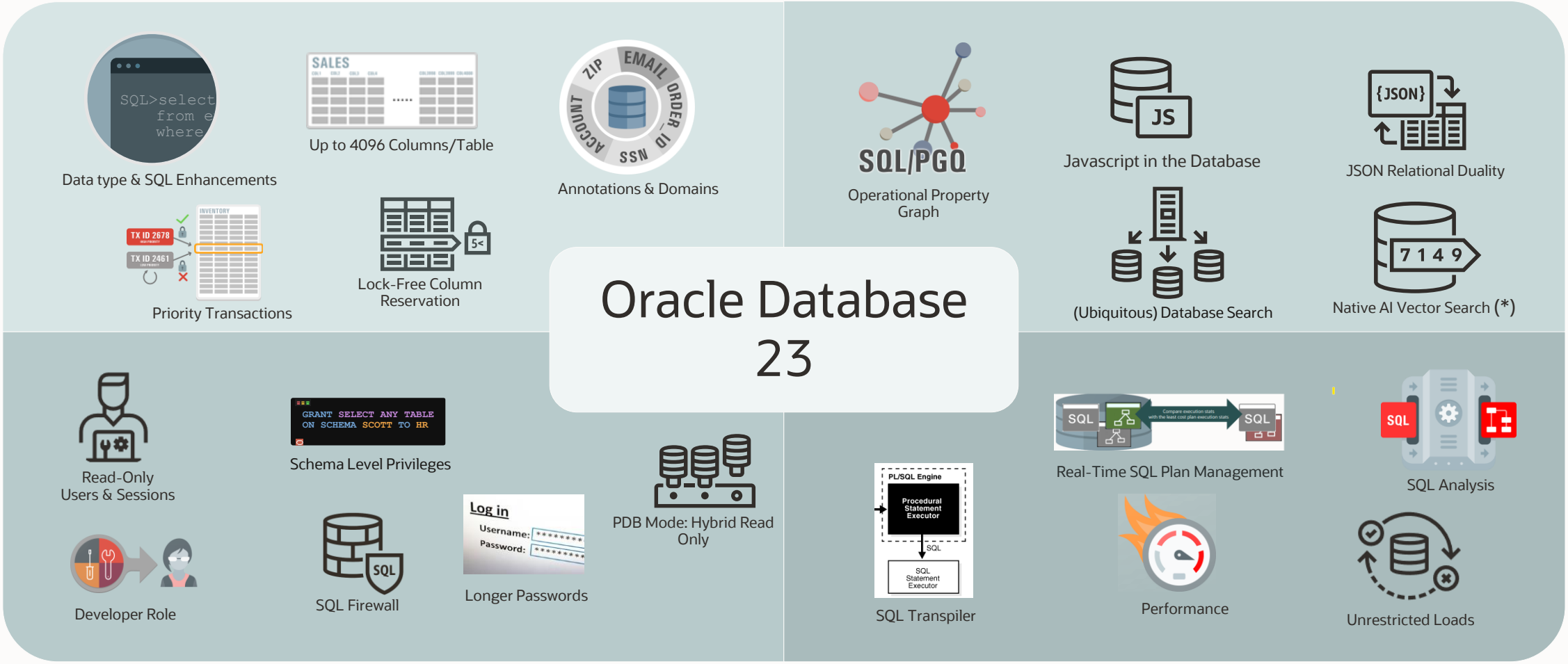
Translated to Oracle analytical SQL

- Mongo pipelines operations equivalent to SQL operations
- Translation to SQL streamlines processing and improves performance

Supports commonly stages and expressions
(`$group, $match, $project, $sort,` etc.)

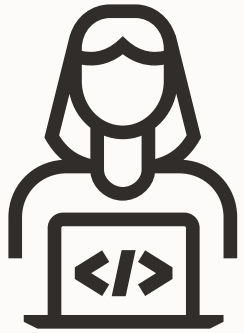


Oracle Database 23, making it more simple and more complete



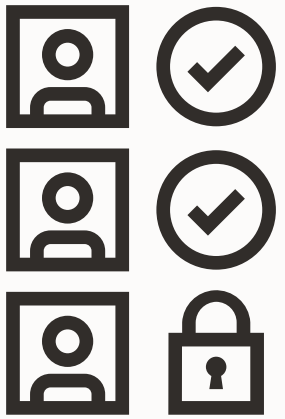
New | Oracle Database 23c (23.3) for Security

SQL Firewall



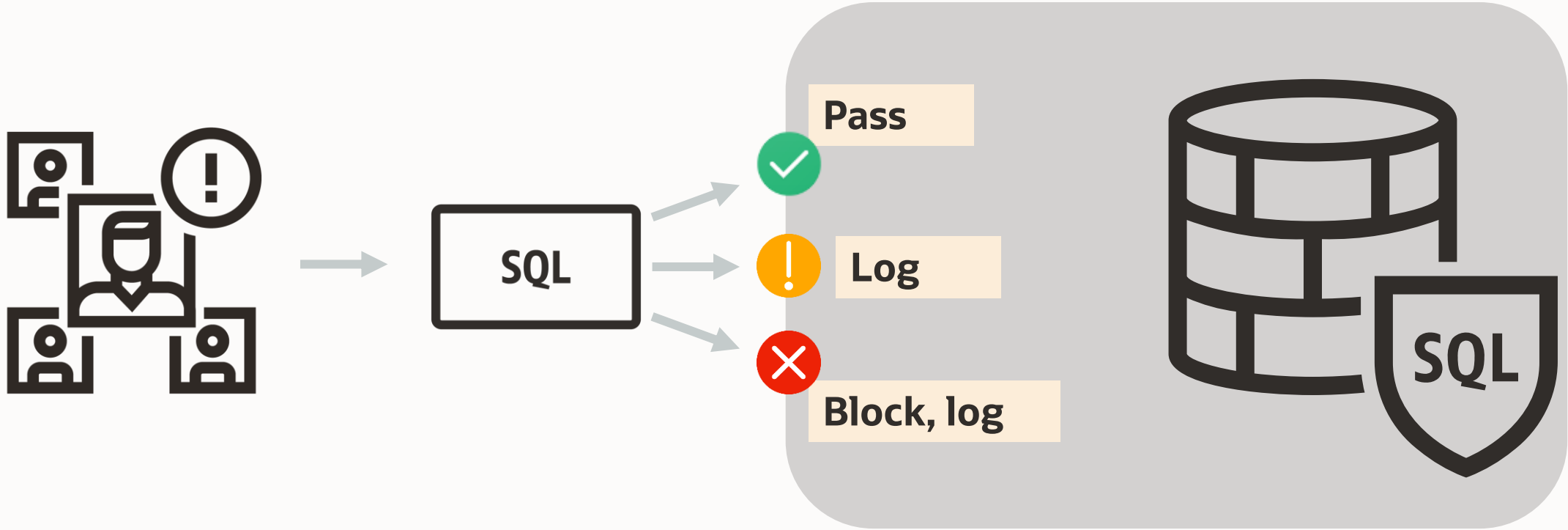
Developer Role

Blockchain Table



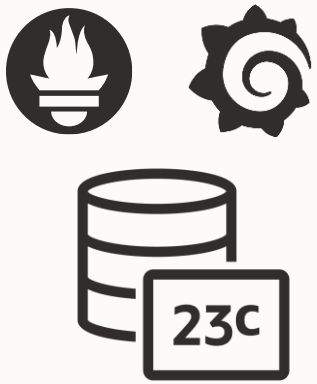
Schema Privileges

New | Oracle Database 23c SQL Firewall

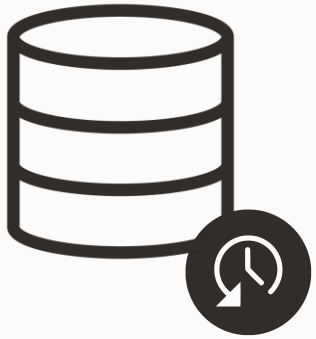


- SQL Firewall is **built into** the database, ensuring that it **cannot be bypassed**
- Oracle SQL Firewall offers protection against common database attacks by monitoring and blocking **“unauthorized SQL”** and SQL injection attacks from inside the database

New | Oracle Database 23c (23.3) for Manageability and Diagnosability

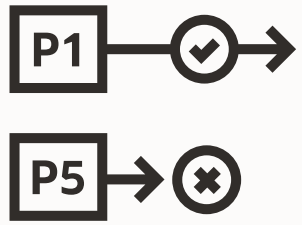


Prometheus/ Grafana for Oracle



Backup Improvements

Error Messages And Logging

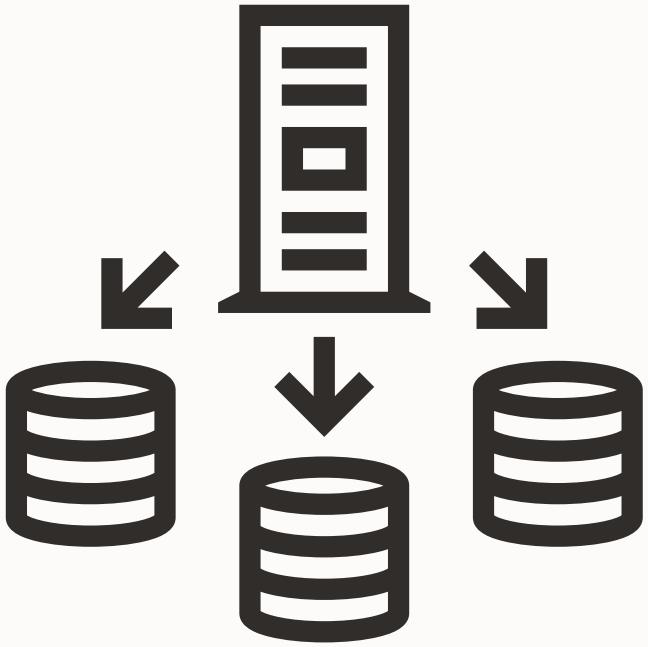


Priority Transactions

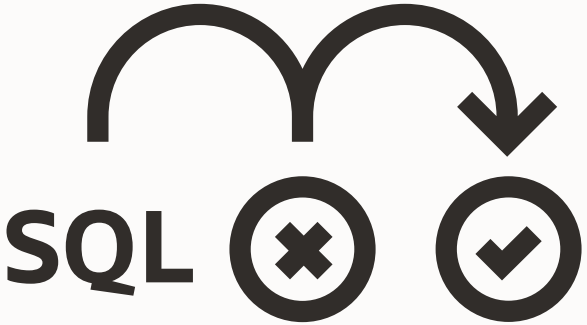
Plus, hundreds of other enhancements to manageability and diagnosability

- Automatic Transaction Quarantine
- New Oracle Call Interface (OCI) APIs
- Set events for a user-specified durations
- In memory diagnostics improvements
- SQLHC and SQLT Diagnostics Tools merged
- Enhancements to [g]v$ views for diagnosis
- Added functionality to hanganalyze dumps
- Improved snapshot too old errors
- RAC related enhancements to diagnostic and auto-mitigation
- Transparent Database Encryption diagnostic enhancements
- Reuse dropped con#
- Improved performance and stability of SMON
- Improved PDB open scalability
- Inter-instance communication for global locks enhancements
- Improvements to excessive tracing across the RDBMS
- New RMAN progress reporting
- SQL Monitor enhancements across multiple PDBs
- Query optimizer enhancements for SQL performance
- Cluster Health Monitor improved diagnosability
- New GoldenGate statistics for performance
- ASM Scrub improvements
- Enhance automatic hang resolution and diagnostics
- Diagnosis and repair of SQL Exceptions
- Auto-healing of CRS resources
- Improve corruptions in encrypted tablespaces
- Enhancement of redo dumps
- Dictionary health check in RDBMS
- Simplify identification of corrupted objects
- Improved background action processing
- Improvements to Major Block Change Tracking (BCT)
- … and more!

New | Oracle Database 23c (23.3) for High Availability & Scalability



Globally Distributed Database with Raft-replication

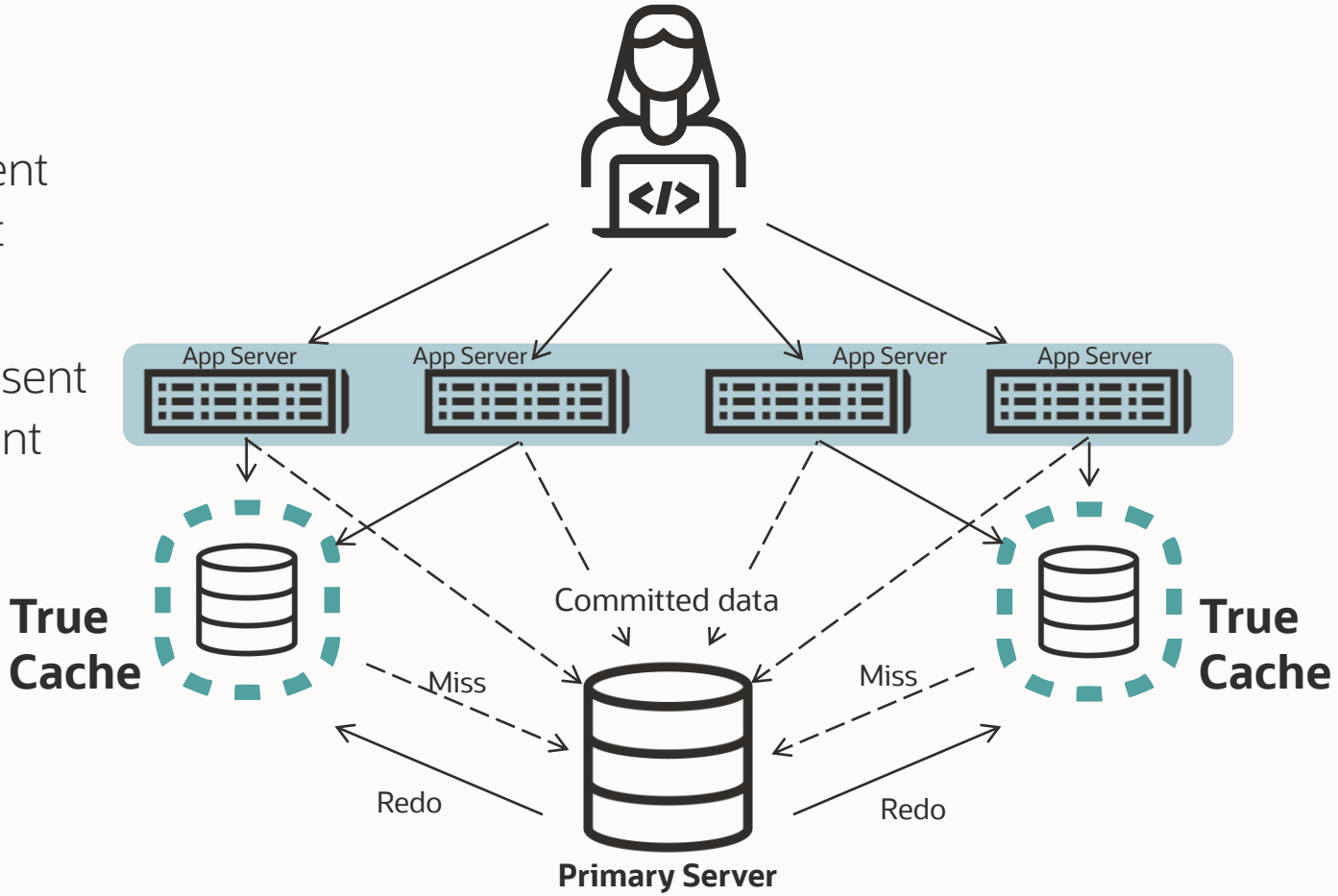


Real-time SQL Plan Management

Oracle Database 23^{c}, What's Coming

Next | Oracle Database 23c True Cache

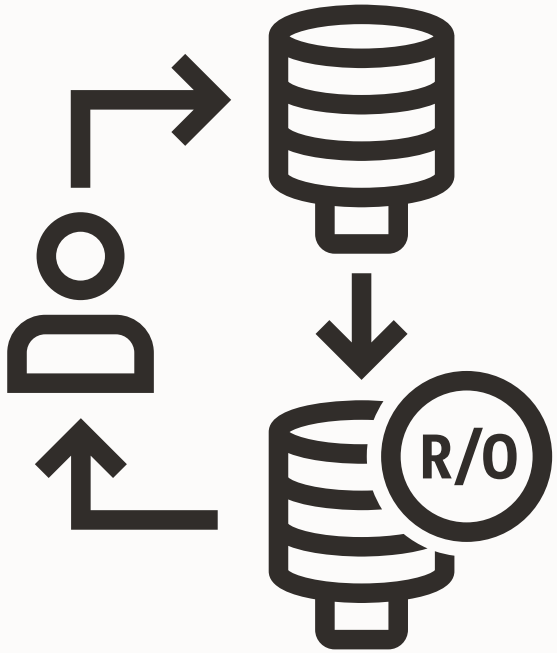
- Solid lines represent relatively frequent requests
- Dotted lines represent relatively infrequent requests



App connect to True Cache and perform SQL queries

True Cache is an in-memory, consistent, and automatically managed full SQL cache

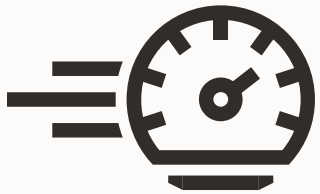
Next | Oracle Database 23c for High Availability & Scalability



Read-Only Per-PDB Standbys

Rolling Patching

Performance Enhancements



Vectorized Query Processing : **1.2 – 2 x** improvement on some queries in TPC-H and Star Schema benchmarks

OCI Pipelining Support : Improves throughput and responsiveness of applications. Can enable better CPU utilization

Ordered Sequences on RAC: Observations of up to **2x** increase in insert performance

Window Functions : More parallelization, better skew handling etc. Resulting in up to **50%** to **1000%** improvements on some queries

OFS: Parallel File System Operations : DBMS_FS operations parallelized with multiple worker threads results in improved performance for mount, create, delete

In-Memory Improvements : Multi-bit Bloom Filters resulted in **50%** improvement on some queries. And other enhancement across query execution

And many, many more…

Make it easy to generate and run modern apps and analytics for all use cases at any scale

Oracle Database Vision

with Generative AI

Key Components of Generative AI

Large Language Model (LLM)

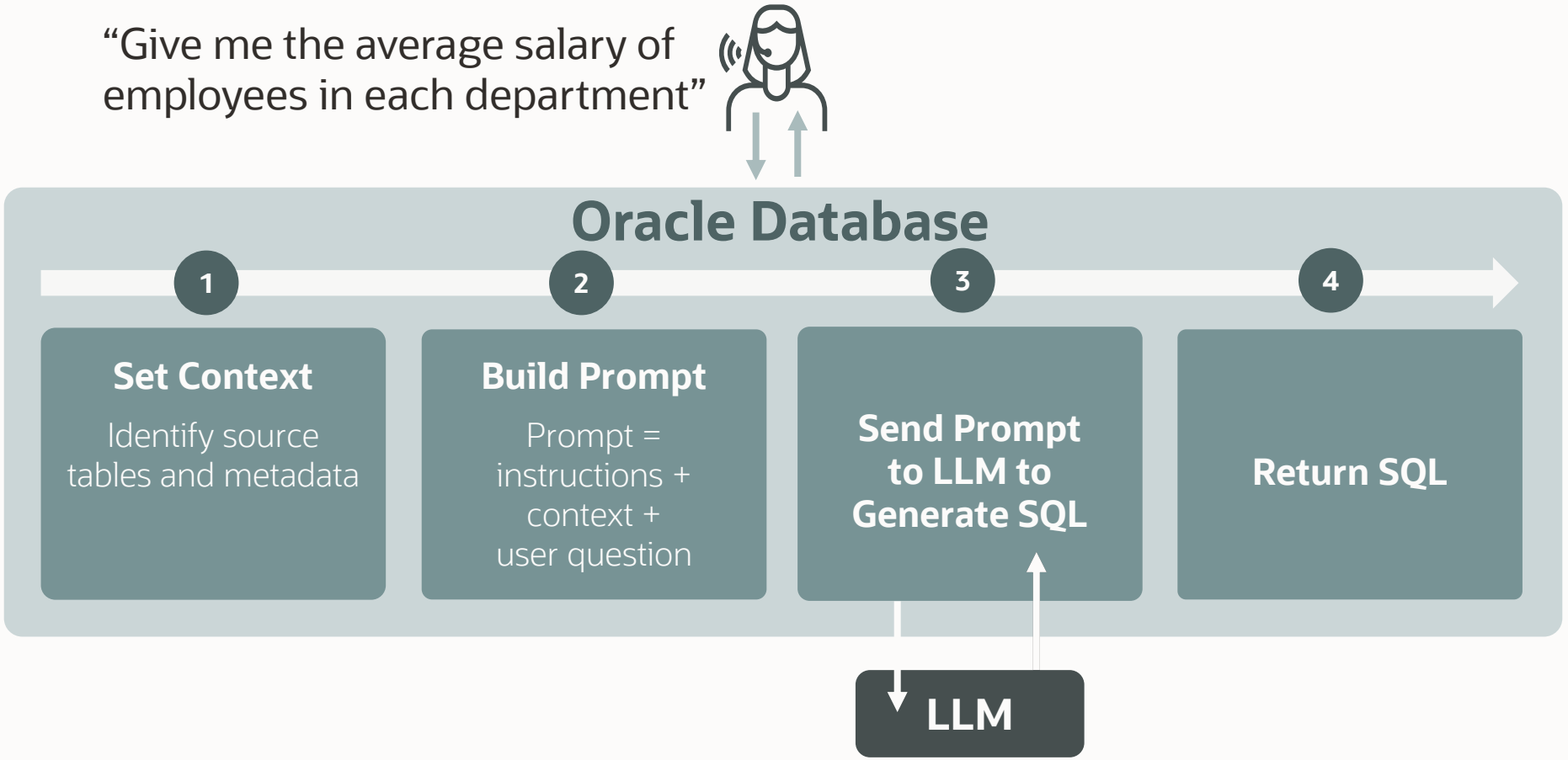
- AI technology designed to understand and generate human language
- Trained on enormous amounts of text data
- Grasps the relationships between words, sentences, and even entire documents
- Accepts natural language input
- Produces coherent and contextually appropriate responses, stories, or other forms of text

Vector Database

- Type of database designed to handle and efficiently search for vectors which can be used to represent documents, images, videos, and audio
- Engineered to store and index vectors in a way that allows for fast and accurate similarity searches
- Useful in tasks like recommendation systems, image and video analysis, natural language processing

Next | SQL Generation from Natural Language using LLM

Generative AI Use Case



Vectors in AI Represent Semantics of Unstructured Data

Such as images, documents, videos, etc.

A vector is a sequence of numbers, called dimensions, used to capture the important "features" of the data

Vectors represent the semantic content of data, not the underlying words or pixels

Vectors are generated using deep learning embedding models

A technology called **Vector Search**
enables user questions
to be mapped to relevant data
in your database

Business Use Cases of Vector Search

Similarity Search

Match candidates with jobs
Image and video retrieval
Find patients with similar issues
Legal e-discovery

Content-Based Filtering

Personalized recommendation
Find item(s) based on image

Natural Language Processing

Text classification and clustering
SQL generation

Data Analytics

Anomaly detection
Pattern recognition

Computer Vision

Face recognition
Biometric identification
Object detection

Biomedical Research

Gene/DNA similarity analysis
Molecular structure search

Geographic Information Systems

Spatial analysis
Map rendering

Industrial Applications

Quality control
Predictive maintenance
Machinery malfunction detection

Concerns with LLMs

Hallucinations

Plausible-sounding false information

Confident response not justified by training data

Security

Model access control: IP theft or model manipulation

Cybersecurity risks to steal data or behave unpredictably

Misinformation and deepfakes

Memory

Trained on snapshot of data in time

No knowledge of private or enterprise data

Ethical AI

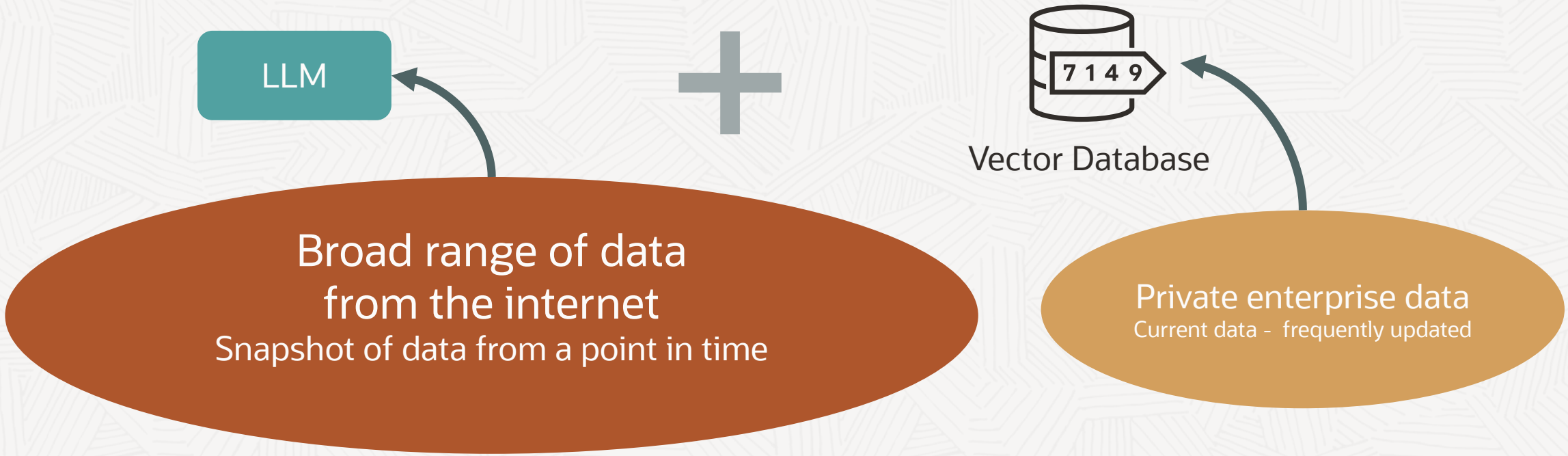
Bias and fairness

Transparency and privacy

Accountability

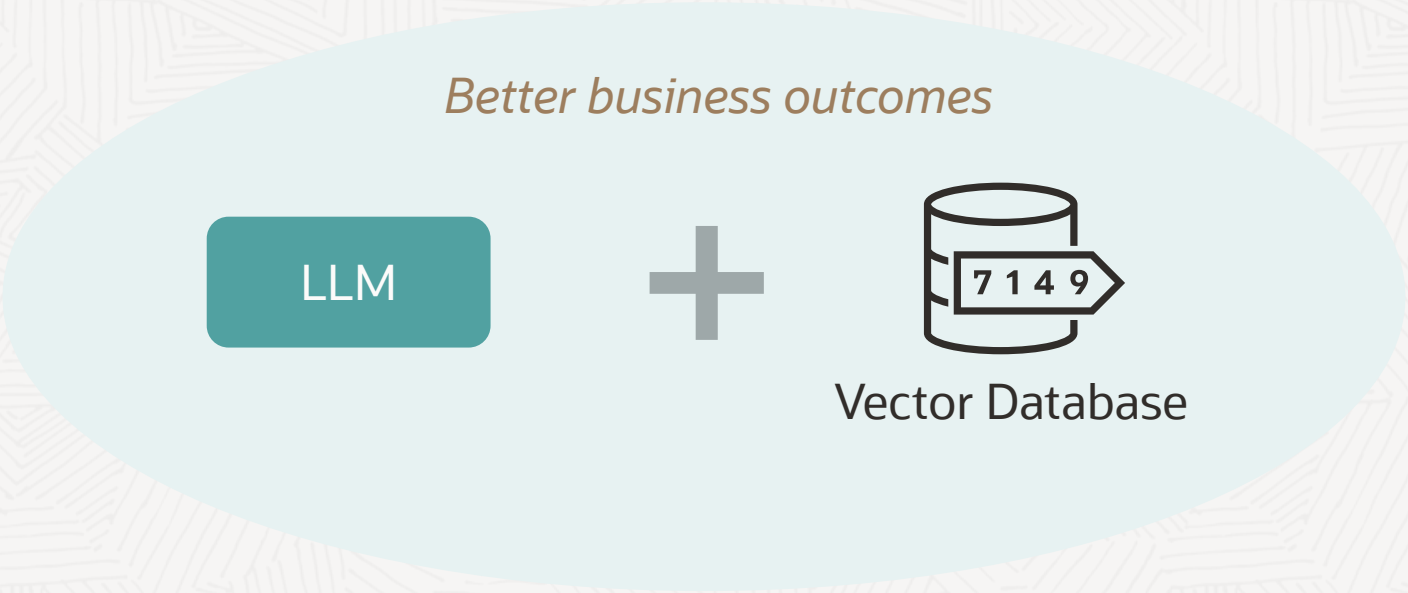
Role of Vector Databases with LLMs

Address the hallucination and memory problem inherent in LLM responses
Augment prompt with enterprise-specific content to produce better responses
Avoid exceeding LLM token limits by using most relevant content



Role of Vector Databases with LLMs

Avoid using sensitive customer data for LLM training and fine-tuning
Cheaper than fine-tuning LLMs, which can be expensive to update
Real-time updated knowledgebase
Cache previous LLM prompts/responses to improve performance and reduce costs



Introducing Oracle AI Vector Search

New set of capabilities coming in Oracle Database 23c

Designed to be simple to use and easy to understand

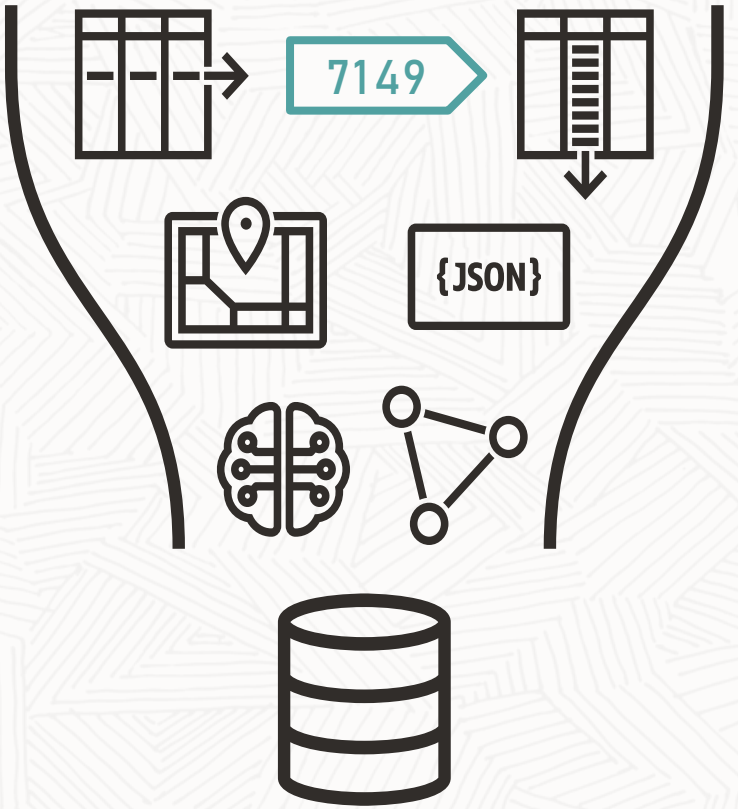
New VECTOR data type for storing vector embeddings

New SQL syntax and functions expresses similarity search with ease

New Approximate search indexes packaged and tuned for high performance and quality

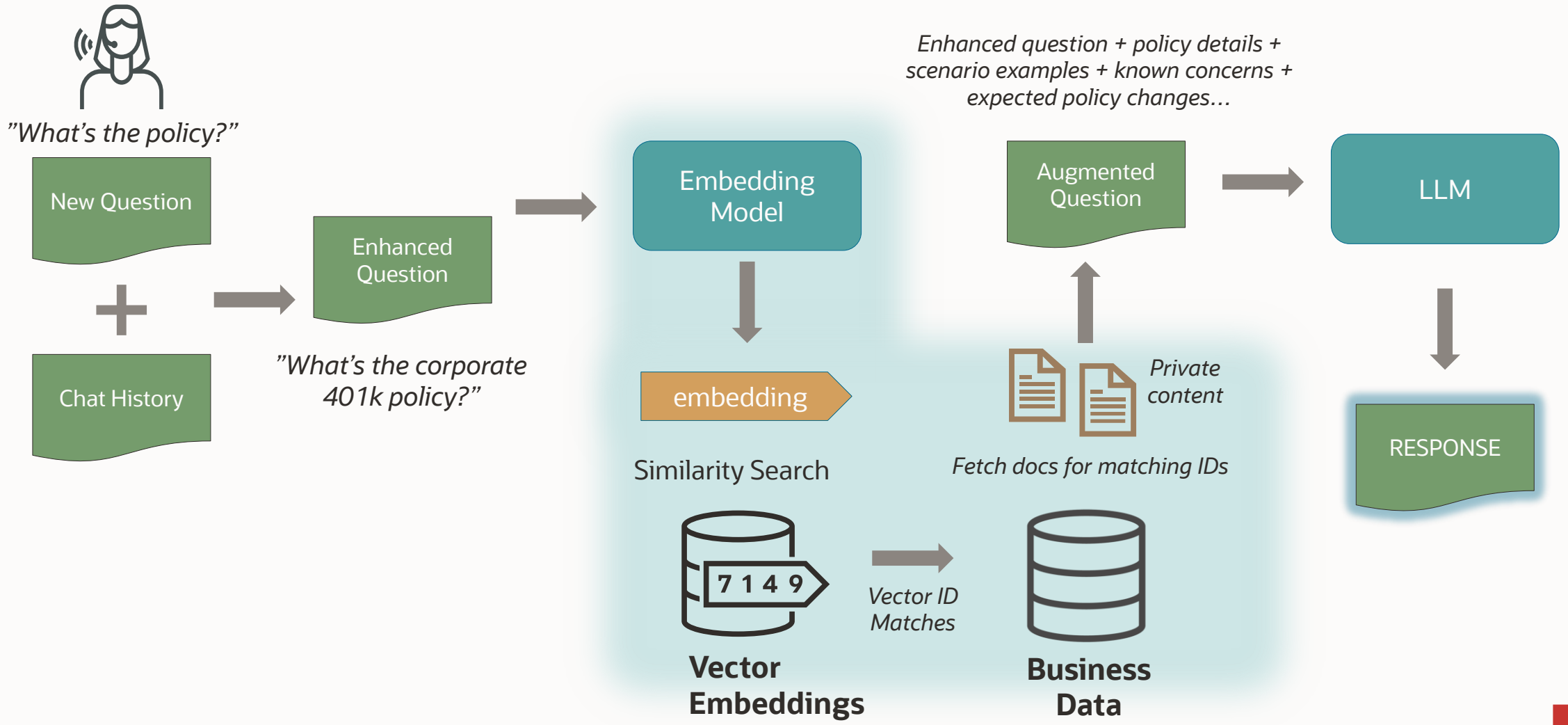
Perform vector search in queries alongside business data about customers and products

Handle vector and other workloads in same database



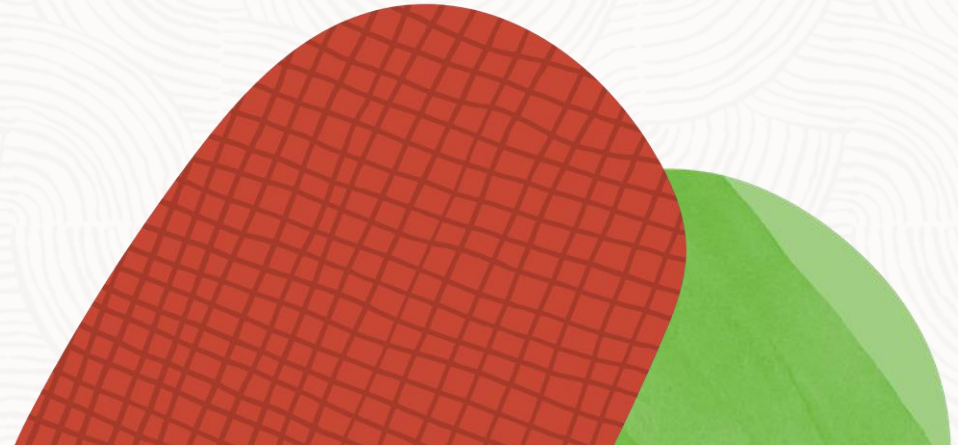
LLM-based Chatbot with “Enterprise Knowledge”

Generative AI Use Case



Upgrading to Oracle Database 23c

App Simple



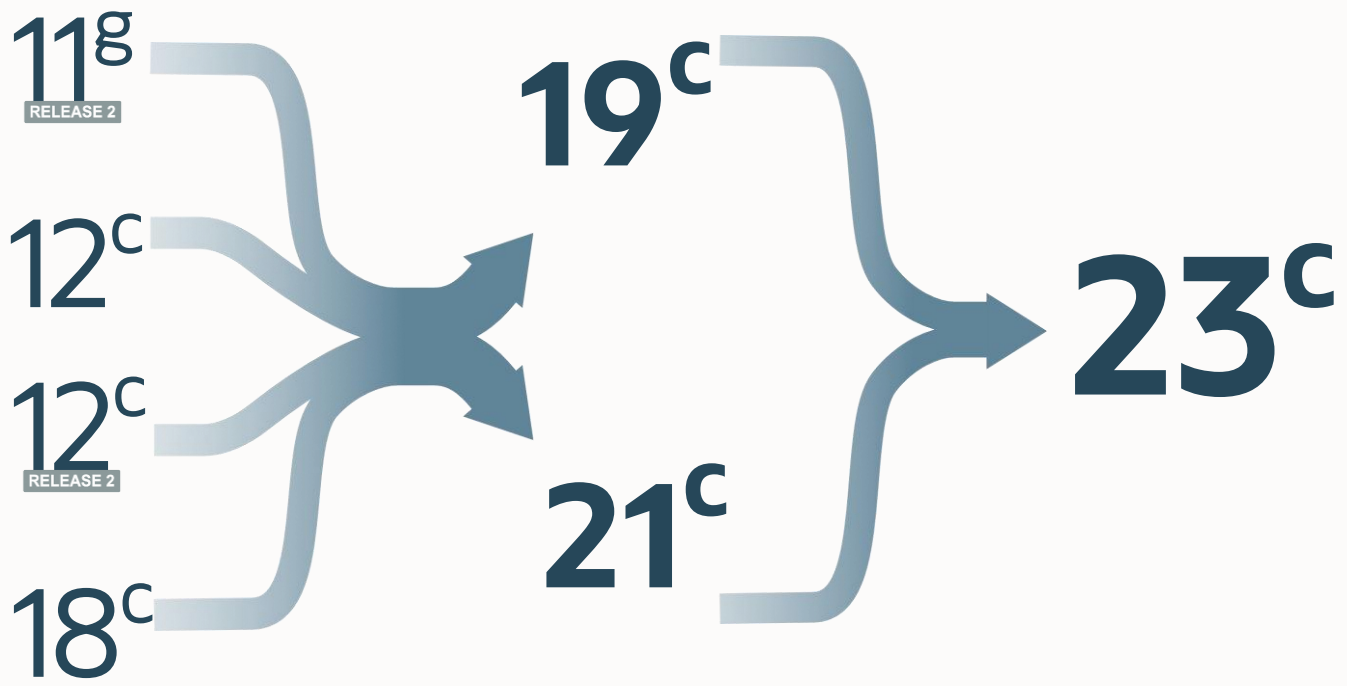
Oracle Database 23c Release Roadmap

Oracle Database 23c (23.3) on Base Database Service

- No RAC, no Standard Edition
- Available September 19, 2023

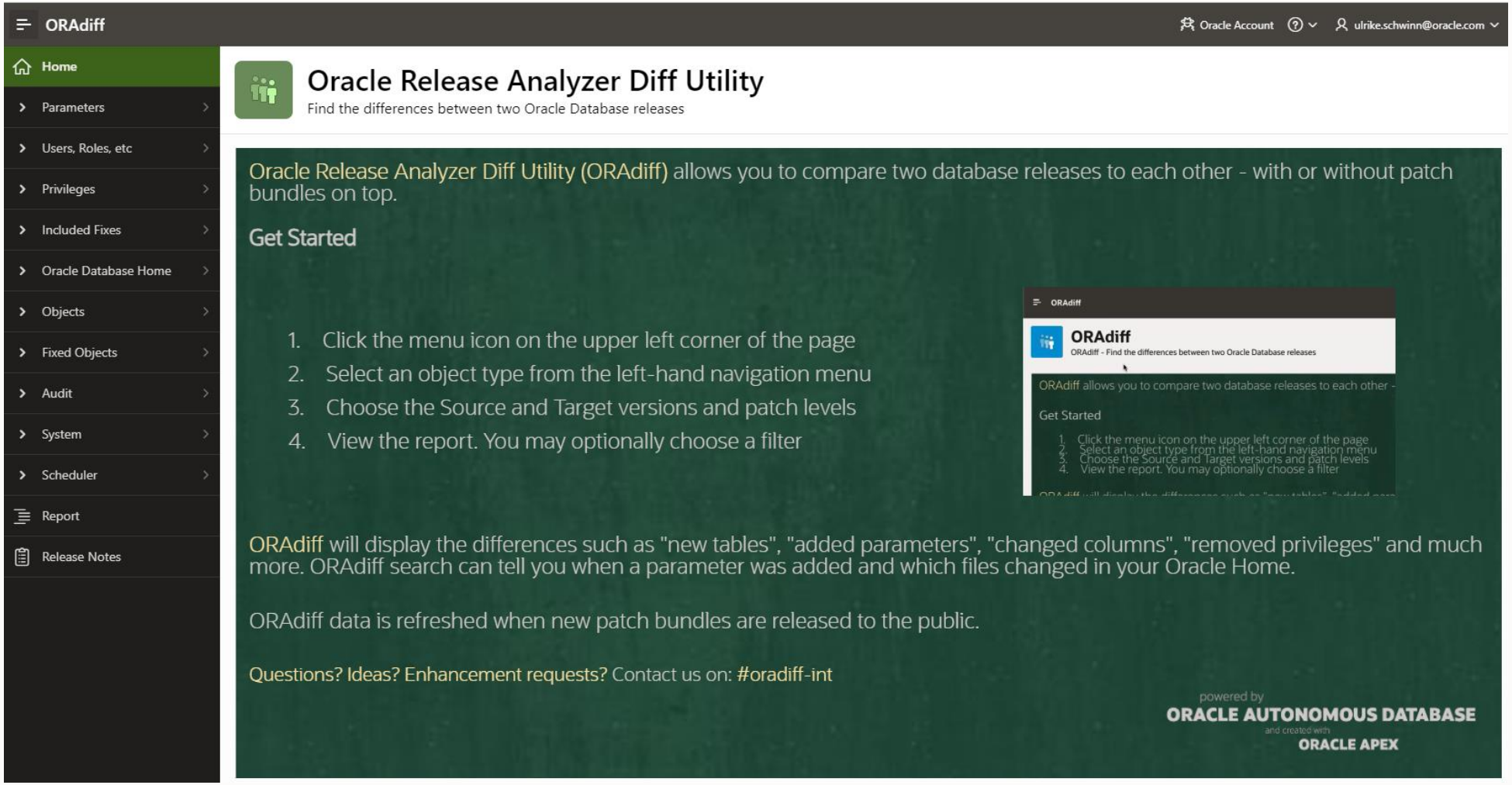
Oracle Database 23c (23.3) Free

- Oracle Linux 8 / Redhat Linux 8
- Available September 19, 2023



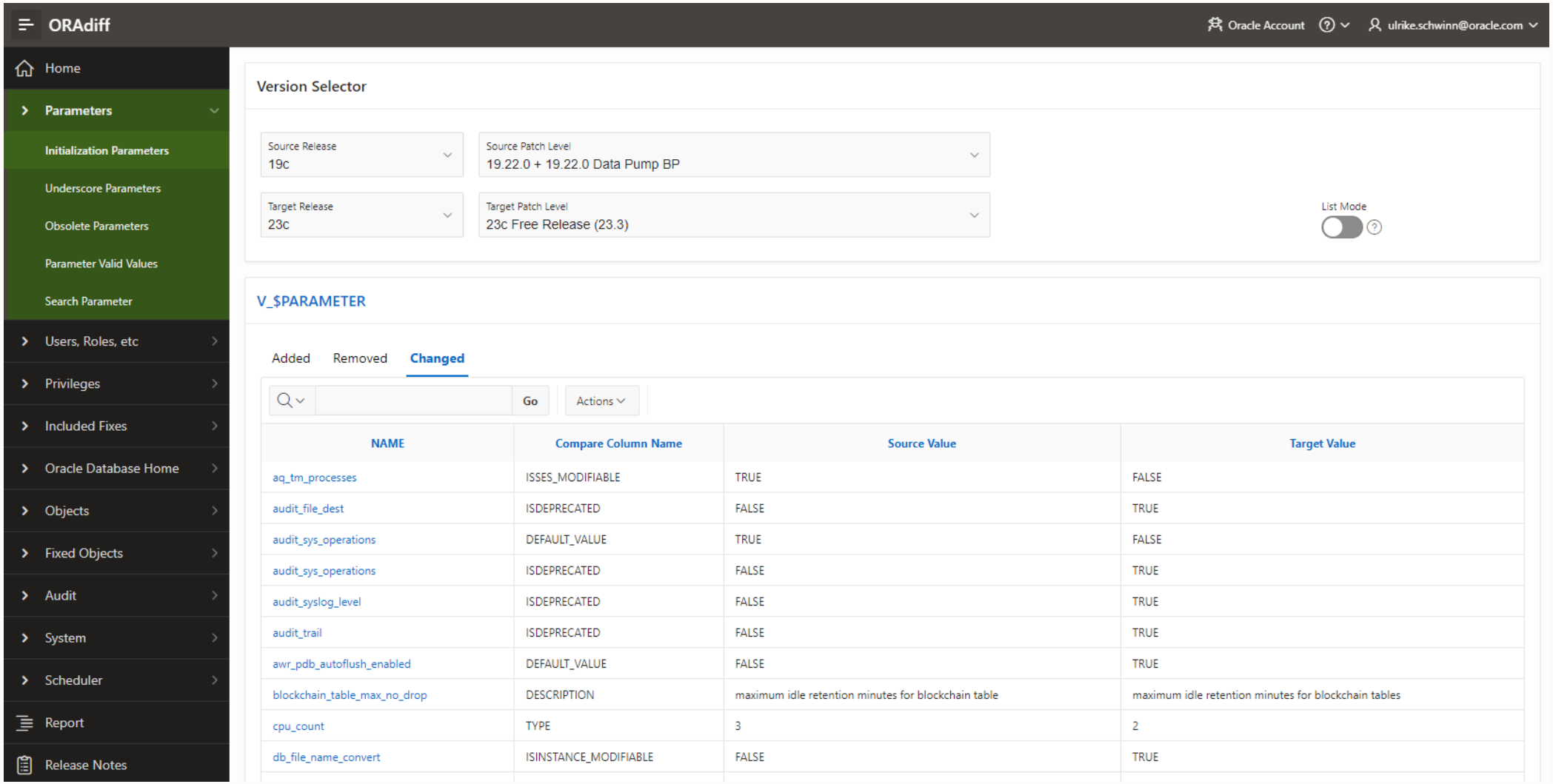
Note: Rolling releases starting in 1H CY2024 for on-premises and in cloud

OraDiff - Find the differences between two Oracle Database releases



https://oradiff.oracle.com/

OraDiff - Find the differences between two Oracle Database releases



NAME	Compare Column Name	Source Value	Target Value
aq_tm_processes	ISSES_MODIFIABLE	TRUE	FALSE
audit_file_dest	ISDEPRECATED	FALSE	TRUE
audit_sys_operations	DEFAULT_VALUE	TRUE	FALSE
audit_sys_operations	ISDEPRECATED	FALSE	TRUE
audit_syslog_level	ISDEPRECATED	FALSE	TRUE
audit_trail	ISDEPRECATED	FALSE	TRUE
awr_pdb_autoflush_enabled	DEFAULT_VALUE	FALSE	TRUE
blockchain_table_max_no_drop	DESCRIPTION	maximum idle retention minutes for blockchain table	maximum idle retention minutes for blockchain tables
cpu_count	TYPE	3	2
db_file_name_convert	ISINSTANCE_MODIFIABLE	FALSE	TRUE

Check Features: Deprecated and Desupported Features

Deprecated Features	Desupported Feature
"Features that are no longer being enhanced, but are still supported for the full life of this release of Oracle Database. "	" Features that are no longer supported by fixing bugs related to that feature. Often, Oracle can choose to remove the code required to use the feature. "

Deprecated Features in Oracle Database 23c Database Upgrade Guide

23

- **Original Export Utility (EXP) Desupported**
  Oracle recommends that you use Oracle Data Pump Export
- **Oracle Enterprise Manager Database Express Desupported**
  Choose EM, SQL Developer, PerfHub, DB Management Service
- **DBUA and Manual Upgrade Desupported**
  Oracle recommends using AutoUpgrade for db upgrades
- **Oracle OLAP Desupported**
  Oracle recommends that you consider analytic views
- **Traditional Auditing**
  Oracle recommends that you use unified auditing

21c

- **Desupport of Non-CDB Oracle Databases**
  Starting with Oracle Database 21c, installation of non-CDB Oracle Database architecture is no longer supported.

Oracle Database Architectures

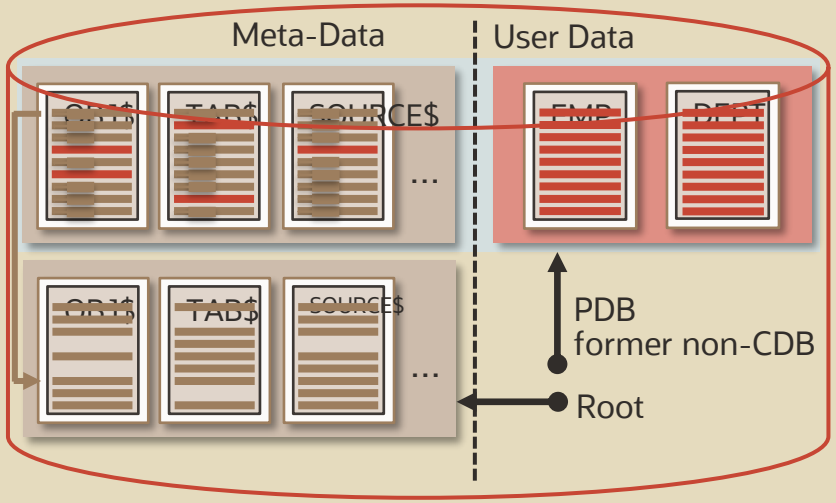
19c

- Non-CDB still supported
- All database features supported with PDB
- Up to three PDBs are free

21c and later

- Multitenant is the only architecture
- (Non-CDB no longer supported)
- Up to three PDBs are free

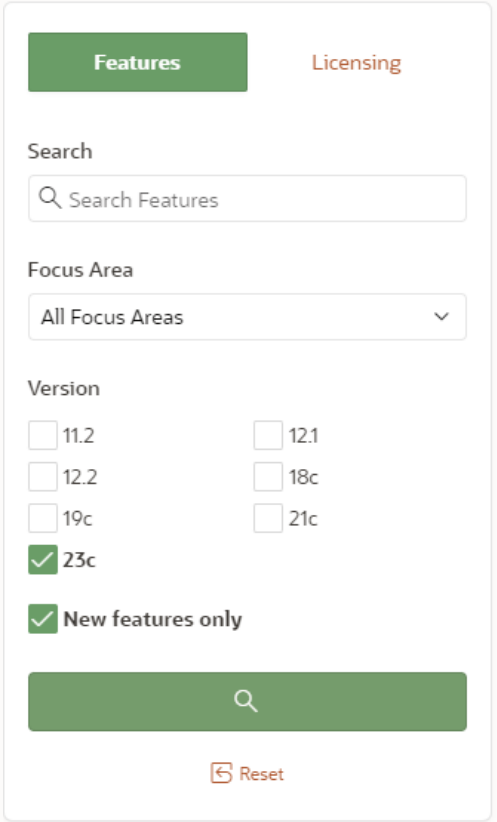
- **Doc ID 2808317.1**



New Features and Release Updates

New Features Guide Chapter 2

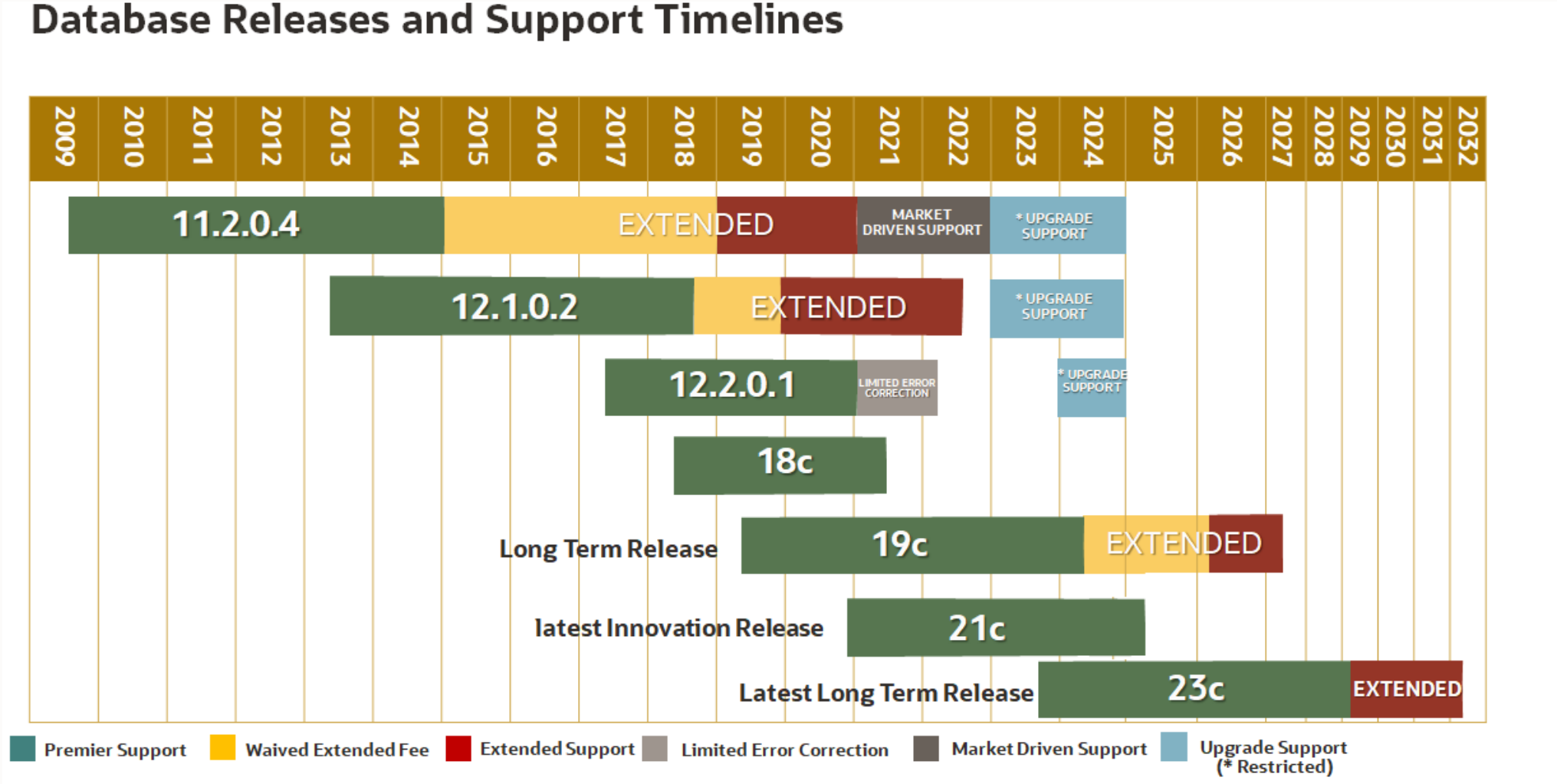
Features and Licensing App



Examples

- SQL Macros (19.7)
- Database in-Memory Base Level (19.8)
- Oracle Blockchain Table (19.10) und Immutable Tables (19.11)
- Integration von Cloud Ressourcen mit DBMS_CLOUD auf Non Autonomous (19.10)
- Immutable Tables (19.11)
- Gradual Database Password Rollover for Applications (19.12)
- In-Memory Eligibility Test (19.20)
- License Change for Automatic SQL Plan Management (19.22)
- DBMS_SPM.ADD_VERIFIED_SQL_PLAN_BASELINE (19.22)

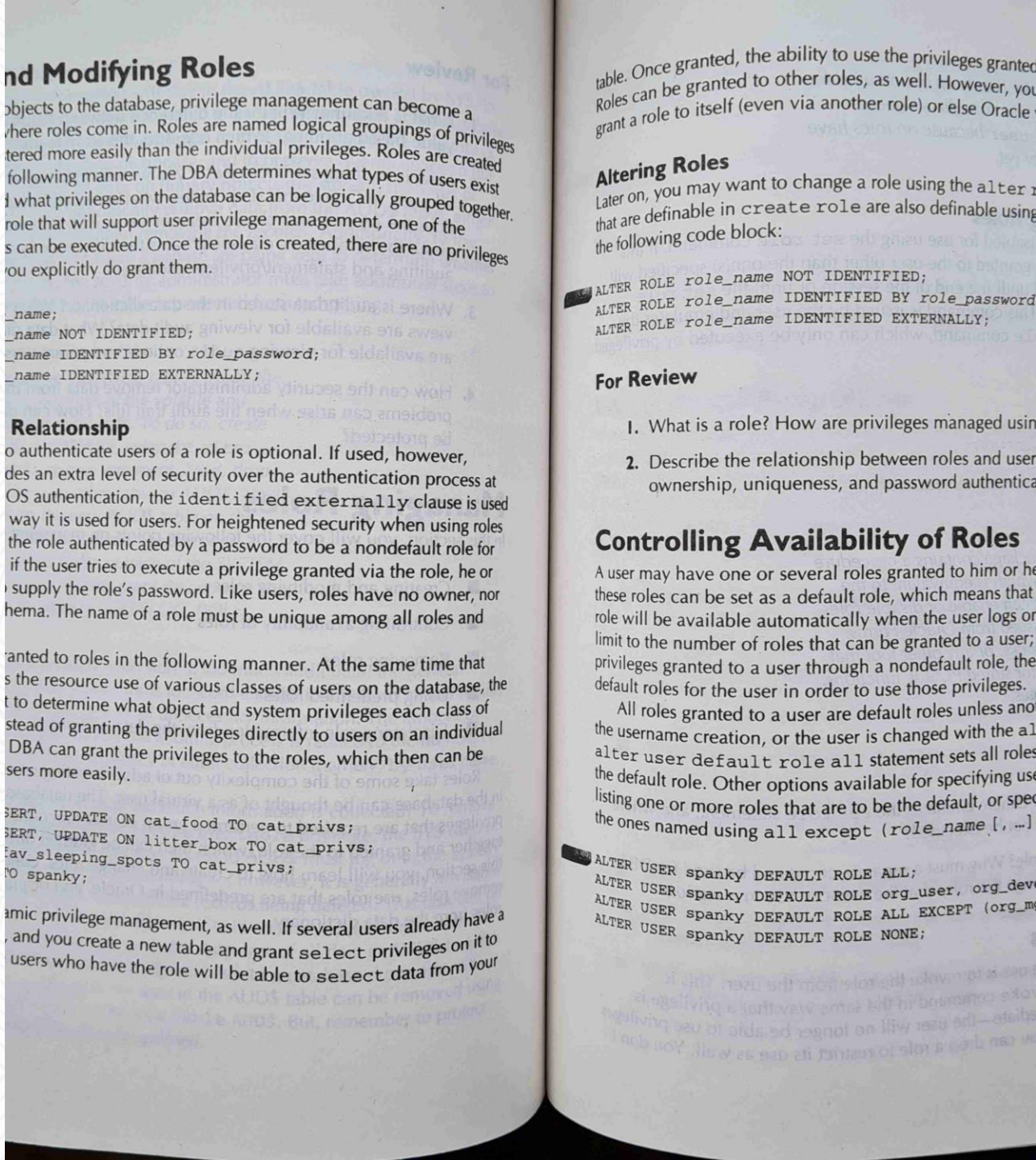
Release Schedule of Current Database Releases (Doc ID 742060.1)



Where to get more information?

Documentation, Postings, Websites, Classes …

- Documentation: Oracle Database 23c
- Product Page: Oracle Database 23c
- Classes
  - Oracle DBA Masterclasses
  - Oracle Tech Lounge
  - Livelabs
- Postings
  - Oracle Database Insider
  - AskTom
  - 23c on blogs.oracle.com/coretec



nd Modifying Roles

bjects to the database, privilege management can become a
here roles come in. Roles are named logical groupings of privileges
tered more easily than the individual privileges. Roles are created
following manner. The DBA determines what types of users exist
what privileges on the database can be logically grouped together.
role that will support user privilege management, one of the
s can be executed. Once the role is created, there are no privileges
ou explicitly do grant them.

```
_name;
_name NOT IDENTIFIED;
_name IDENTIFIED BY role_password;
_name IDENTIFIED EXTERNALLY;
```

Relationship

o authenticate users of a role is optional. If used, however,
des an extra level of security over the authentication process at
OS authentication, the identified externally clause is used
way it is used for users. For heightened security when using roles
the role authenticated by a password to be a nondefault role for
if the user tries to execute a privilege granted via the role, he or
supply the role's password. Like users, roles have no owner, nor
hema. The name of a role must be unique among all roles and

anted to roles in the following manner. At the same time that
s the resource use of various classes of users on the database, the
t to determine what object and system privileges each class of
stead of granting the privileges directly to users on an individual
DBA can grant the privileges to the roles, which then can be
sers more easily.

```
ERT, UPDATE ON cat_food TO cat_privs;
ERT, UPDATE ON litter_box TO cat_privs;
av_sleeping_spots TO cat_privs;
O spanky;
```

amic privilege management, as well. If several users already have a
and you create a new table and grant select privileges on it to
users who have the role will be able to select data from your

table. Once granted, the ability to use the privileges granted
Roles can be granted to other roles, as well. However, you
grant a role to itself (even via another role) or else Oracle

Altering Roles

Later on, you may want to change a role using the alter
that are definable in create role are also definable using
the following code block:

```
ALTER ROLE role_name NOT IDENTIFIED;
ALTER ROLE role_name IDENTIFIED BY role_password
ALTER ROLE role_name IDENTIFIED EXTERNALLY;
```

For Review

1. What is a role? How are privileges managed usin
2. Describe the relationship between roles and user
   ownership, uniqueness, and password authentica

Controlling Availability of Roles

A user may have one or several roles granted to him or he
these roles can be set as a default role, which means that
role will be available automatically when the user logs or
limit to the number of roles that can be granted to a user;
privileges granted to a user through a nondefault role, the
default roles for the user in order to use those privileges.

All roles granted to a user are default roles unless ano
the username creation, or the user is changed with the al
alter user default role all statement sets all roles
the default role. Other options available for specifying use
listing one or more roles that are to be the default, or spec
the ones named using all except (role_name [, …]

```
ALTER USER spanky DEFAULT ROLE ALL;
ALTER USER spanky DEFAULT ROLE org_user, org_dev
ALTER USER spanky DEFAULT ROLE ALL EXCEPT (org_m
ALTER USER spanky DEFAULT ROLE NONE;
```