ORACLE

ORACLE

Base de Datos Convergente:
Machine Learning, Spatial and Graph Workshop

Manel Moreno
Andrés Araújo
Daniel Villaverde
Francisco Alvarez

8, 9 y 10 de marzo 2022
Zoom sessions

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.

Agenda

Agenda

Single-Purpose vs. Multi-Purpose

Instead of

Phones,
Messaging,
Camera, Calendar,
Music, Navigator,
Notepad,
Calculator…

Smart Phone

Converged Database

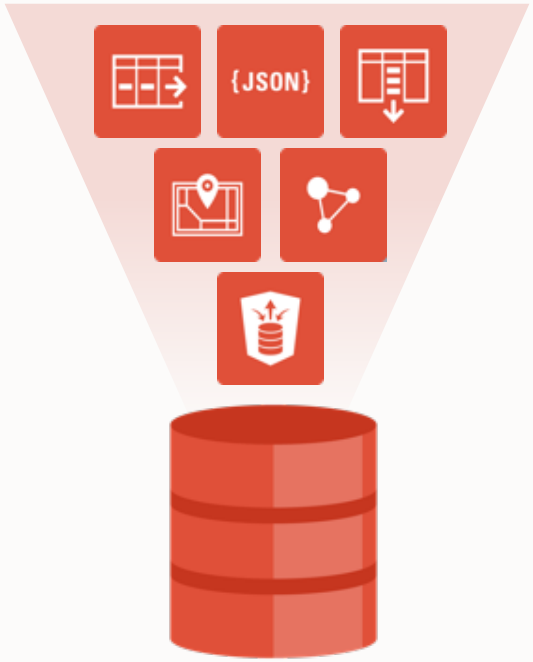
Instead of

Relational, No-SQL,
JSON, XML,
Transactional,
Analytics, In-Memory,
IoT, ML, Blockchain,
Spatial, Sharding…

Oracle Converged Database
Multi-Model and Multi-Workload

Converged Database

Multi-Model



Multi-Workload

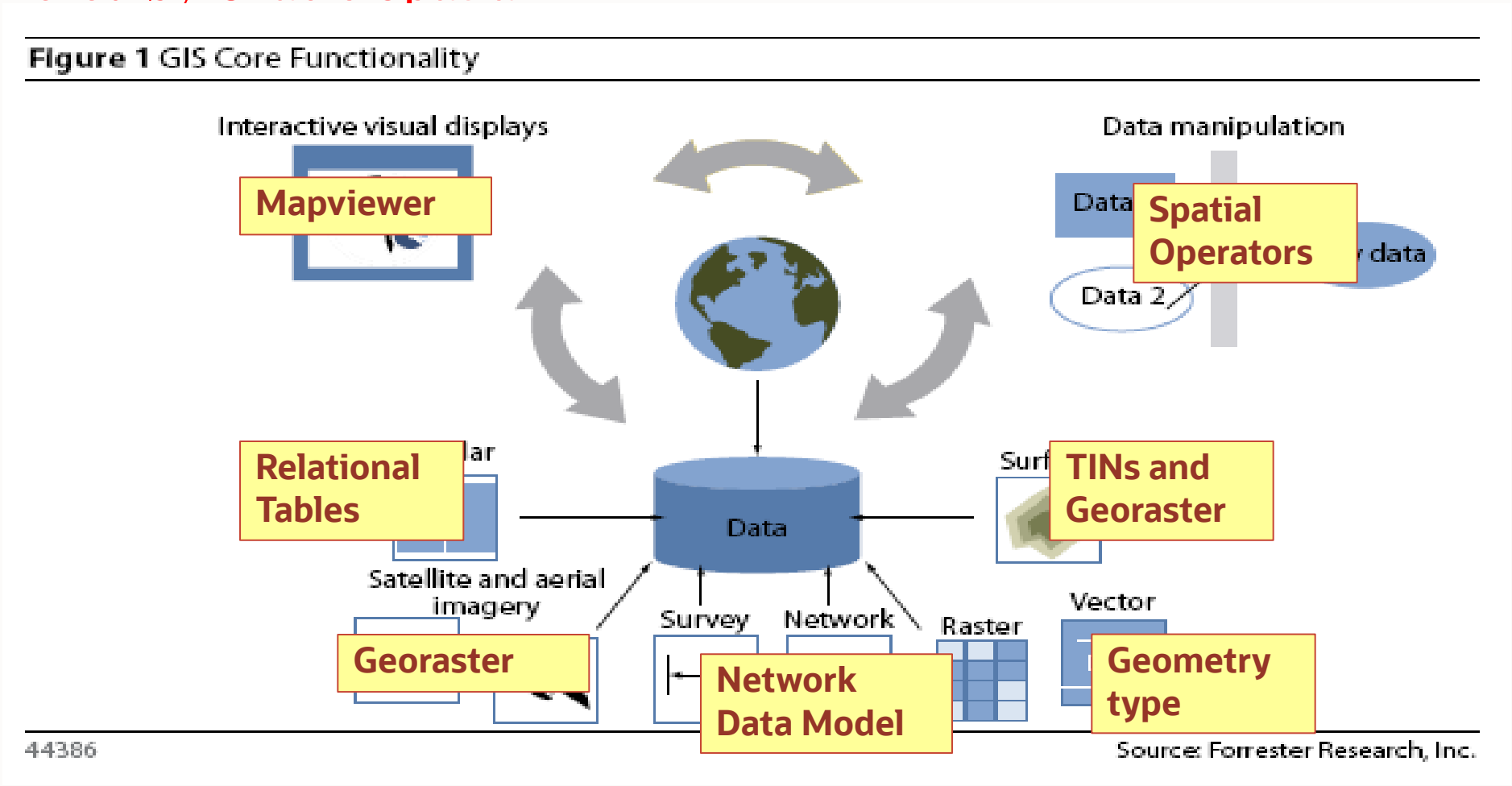
Multiple Data Types
(models and semantics)
Relational, Document, JSON, XML, OLAP, Spatial, Graph, Object-Oriented, Text, etc.

Multiple Application Types
(workloads and technologies)
Operational, Analytics, **Translytics**, Transactional, IoT, ML, In-Memory, Block-Chain, **HTAP**, etc.

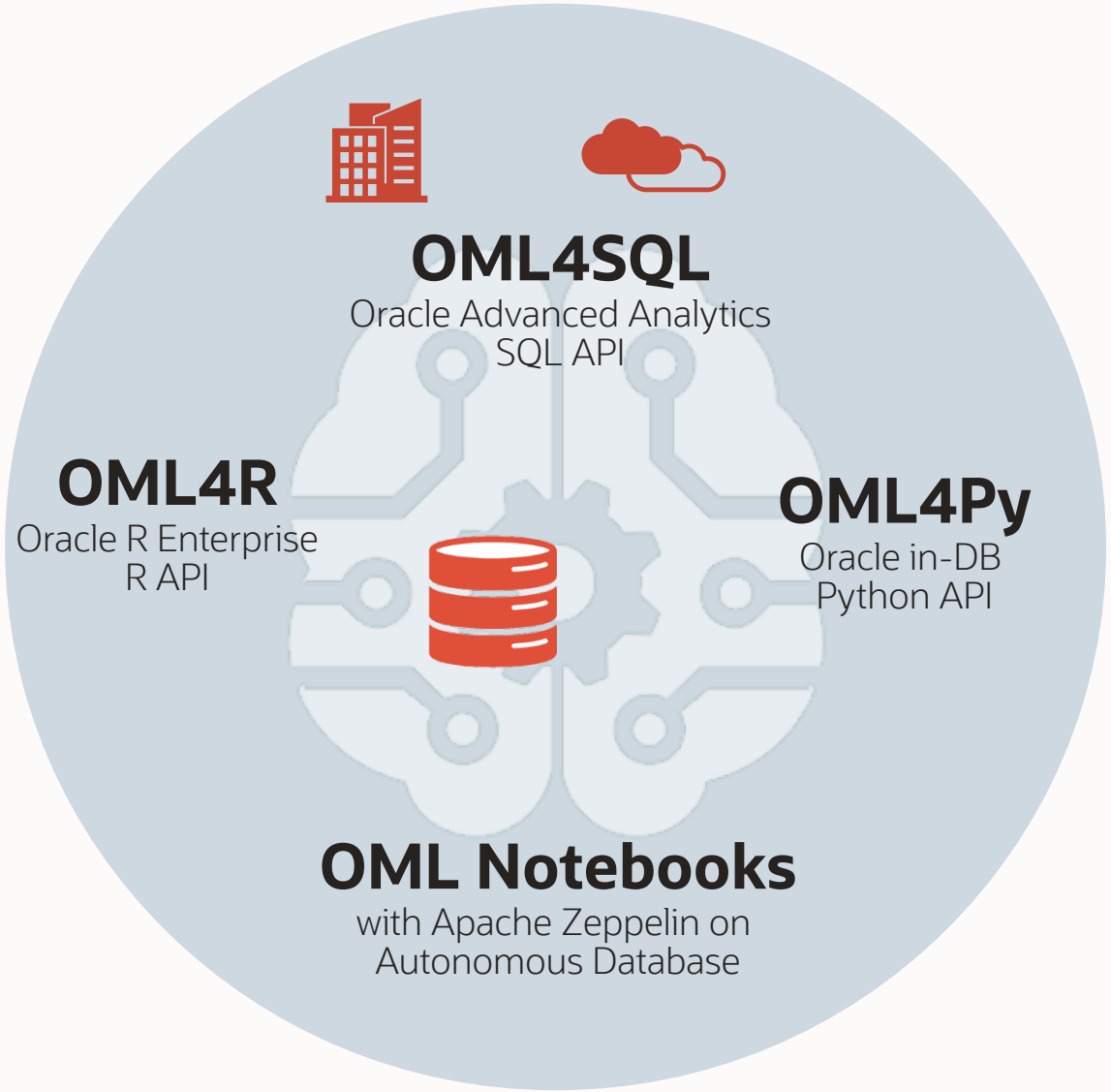
Oracle runs one **Converged Multi-Purpose Database** supporting multiple data types and workloads
Avoid running many **Specialized Single-Purpose Databases** for each data type and workload

GIS Core Functionality

All Covered by Oracle Spatial



Oracle Machine Learning

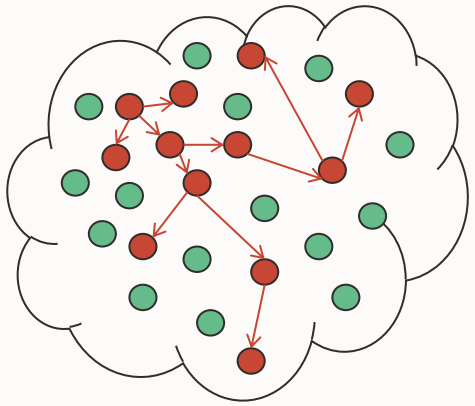


- In-DB Parallel ML Framework
- Python, R or PLSQL
- Cloud Notebook Interface
- Model Lifecycle Management
- Auto-ML and Model Explanation
- Leverage DB Security
- REST and SQL APIs for Scoring

Multi-Workload

Oracle Machine Learning and Graph Analytics

Graph Analytics



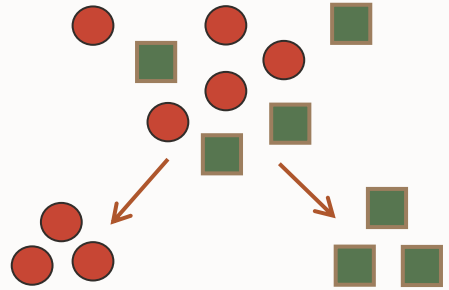
Compute graph metric(s)

Explore graph or compute new metrics using ML result

Add to structured data

Add to graph

Machine Learning



Build predictive model using graph metric

Use models to score or classify data

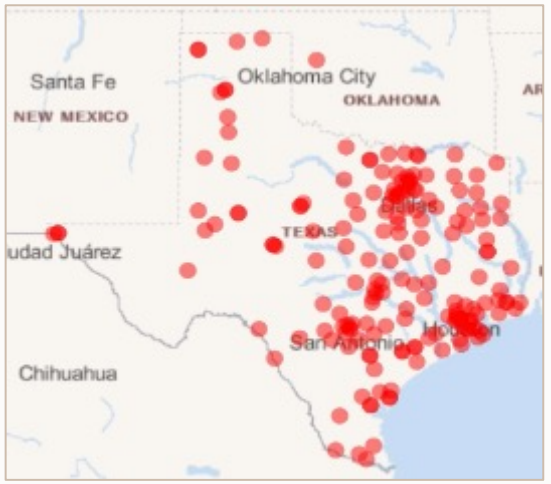
Multi-Workload

Agenda

Oracle Spatial– Key Spatial Features

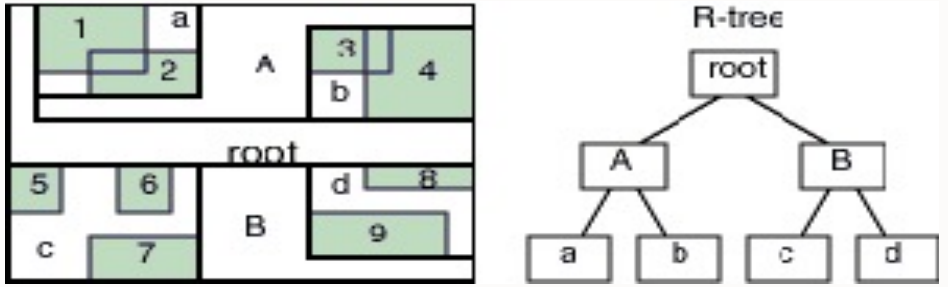
- In-database support for different kinds of geospatial data
- Vector Data (Points, Lines, Linestrings, Areas)
- Geo-referenced Raster Imagery (Orthophotos, Satellite Images, …)
- 3D Point Cloud Data (Laser scanning, Photogrammetry)
- Network Data (Road Networks, Utility Networks)
- Topology Data (Land management)
- Streaming Point Data (Location tracking)

- Deployable Services
- Map visualization
- Geocoding
- Routing
- Publishing (OGC Web Services)



Database Capabilities for Geospatial Analysis

- Data type to store points, lines, areas, solids, …
  - In two or three dimensions
  - Taking into account coordinate system
- Topological operators
  - Point-in-polygon, intersecting linestrings, overlapping areas, …
- Geometric functions
  - Calculating areas, distances, buffer zones, …
- Spatial indices
  - Fast access to relevant data



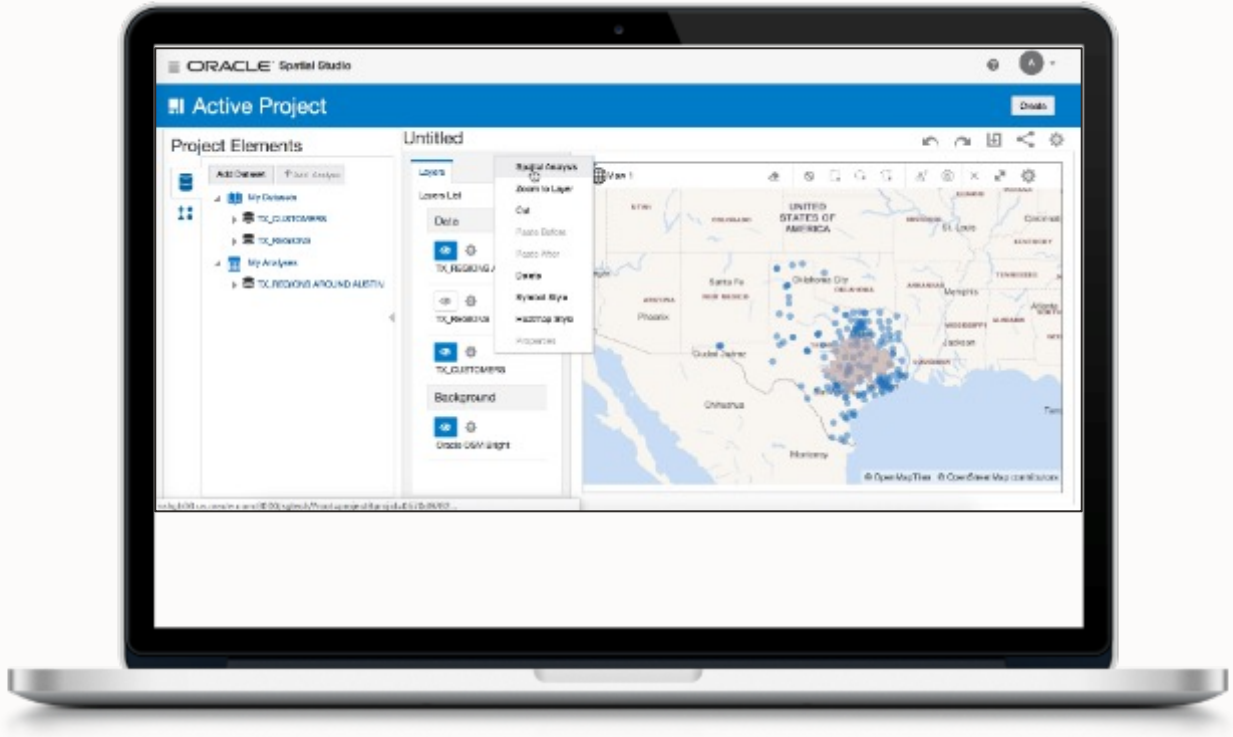

```
SELECT a.owner_name, a.acquisition_status
FROM properties a, projects b
WHERE sdo_within_distance (a.property_geom1,
b.project_geom,
    'distance = 25 unit = meter') = 'TRUE'
and b.project_id=189498;
```

Benefits of Managing Spatial Data in Oracle DB

- Multi-model database, integrating all kinds of data
  - Relational data, XML or JSON documents, spatial data, images, …
- Comprehensive server-side ETL and analytics capabilities
  - Data integration, geospatial analysis, machine learning, graph analysis, …
- Secure datastore
  - Multi-level access control, encryption, redaction, auditing, …
- Highly available, scalable infrastructure
  - Clustering, parallelization, Maximum Availability Architecture (MAA), …
- Core component of data management platform for analytics
  - Tools integration, standards support, open interfaces, Big Data connectivity, …

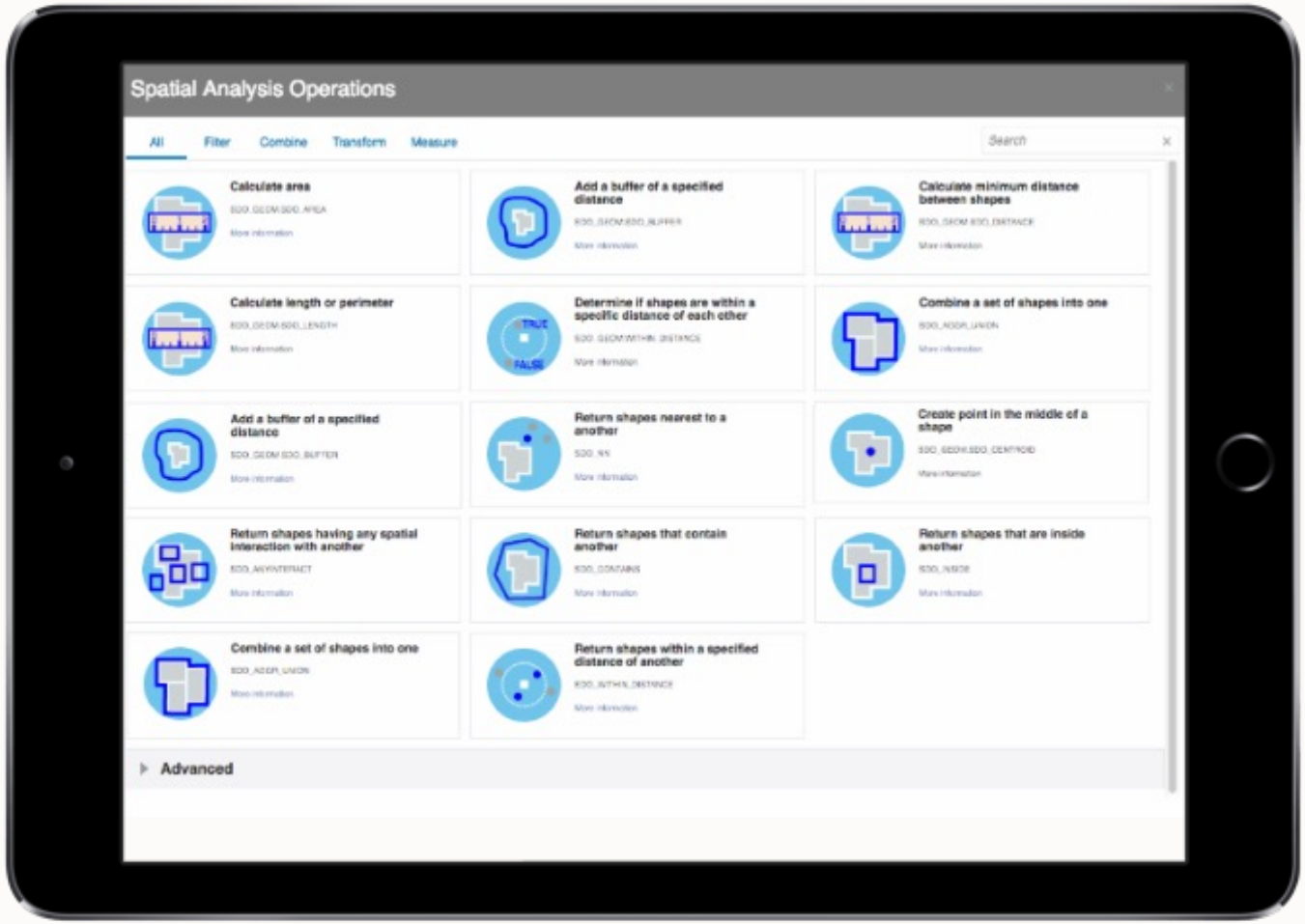
Typical Data Analysis Workflow

- Data ingestion
  - Spatial and non-spatial data
- Data enrichment
  - Address geocoding
  - Converting placenames
- Geospatial processing
  - Creating analytical workflows
- Interactive analysis
  - Map visualization
- Publication of results



Spatial Studio – Self-service spatial analytics

Spatial Studio – Simple Geospatial Analysis



Major New Spatial Features

Ease of Use

- Spatial Studio - Self-service development tool
- Improved JSON and Oracle REST Data Services
- Enhanced Location Tracking Server
- Map Visualization
- Improved web services support (CSW, WFS)
- Georaster enhancements

Performance

- Spatial index performance improvements
  - 3x faster queries for large point data sets
- Map visualization dynamic tile layer
  - Save storage overhead on large, complex queries

Improved Database Integration

- Spatial support for all partitioning methods
- Spatial support for distributed transactions
- Spatial support for database sharding
- Improved support for queries on external tables

Storing and Querying JSON

Application developers:
Store JSON using RESTful API

```
PUT /my_database/my_schema/customers HTTP/1.0
Content-Type: application/json
Body:
{
 "firstName": "John",
 “lastName”: "Smith",
 "age": 25,
 "address": {
       "streetAddress": "21 2nd Street",
       "city": "New York",
       "state": "NY",
       "postalCode": "10021“,
       "isBusiness" : false  },
       "phoneNumbers": [
       {"type": "home",
        "number": "212 555-1234“ },
       {"type": "fax",
        "number": "646 555-4567“ }  ]
}
```

Analytical tools and business users:
Query JSON using SQL

```
select
  c.document.firstName,
  c.document.lastName,
  c.document.address.city
from customers c;”
```

```
firstName    lastName     address.city
-----------  -----------  --------------
“John”       “Smith”      “New York”
```

GeoJSON: JSON for geometries

```
select json_value(
 '{
    "type": "Point",
    "coordinates": [125.6, 10.1]
  }', '$'
  returning sdo_geometry
)
from dual;
```

```
SDO_GEOMETRY(2001, 4326,
SDO_POINT_TYPE(125.6, 10.1, NULL),
NULL, NULL)
```

- Extend JSON support in the database with Spatial operations
- JSON_VALUE() supports GeoJSON and SDO_GEOMETRY
- SDO_GEOMETRY constructors extended to take JSON as input
- Support spatial index and spatial queries on JSON documents
- Coordinates are in WGS84 (4326)

SDO_GEOMETRY or GeoJSON?

Determined by application design

- Applications requiring a JSON document store, can keep geometry attributes stored as GeoJSON inside the JSON documents

Can mix JSON and spatial predicates in same query

Conversion between JSON and SDO_GEOMETRY

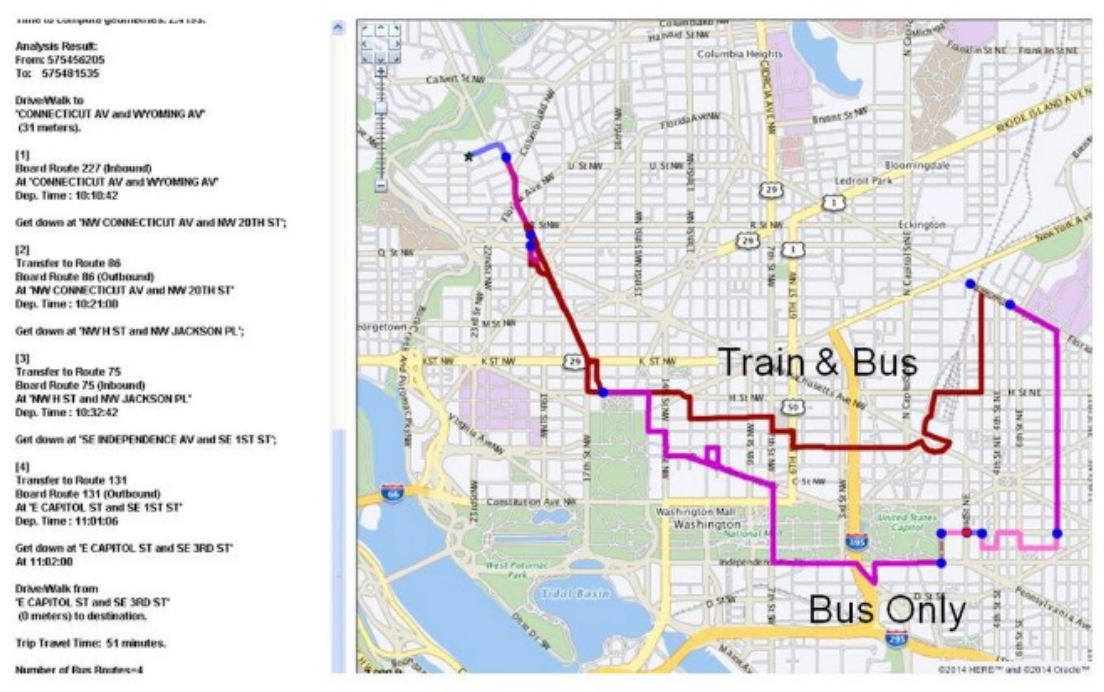
- SDO_UTIL.From_GeoJSON() and To_GeoJSON functions
- SDO_GEOMETRY Get_GeoJSON() method

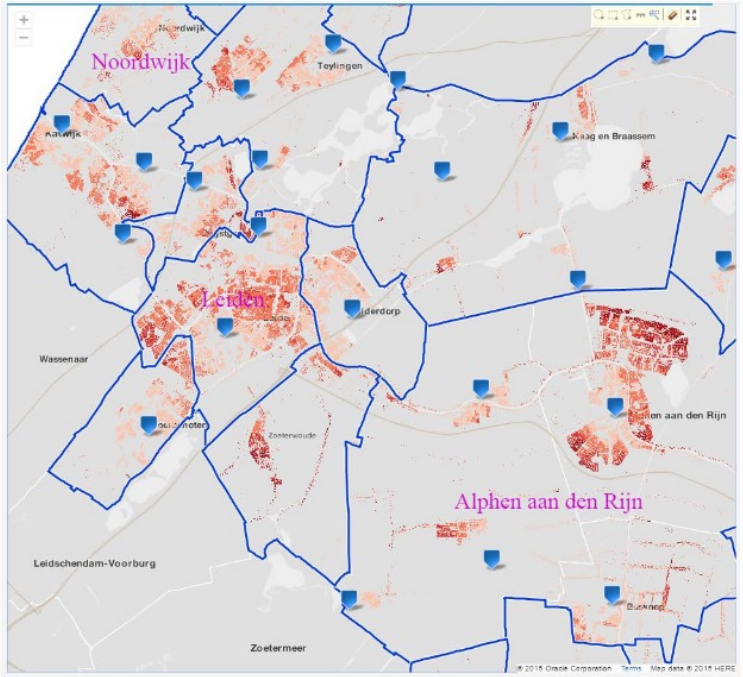
Integrate third party applications that expect SDO_GEOMETRY by

- Defining a **function-based index** on JSON returning SDO_GEOMETRY

Advanced Spatial Data Models

- **Spatial networks for roads, transport, pipelines, telcos and other geographically connected analysis**
- **Topology for mapping, land management and cadastre applications**

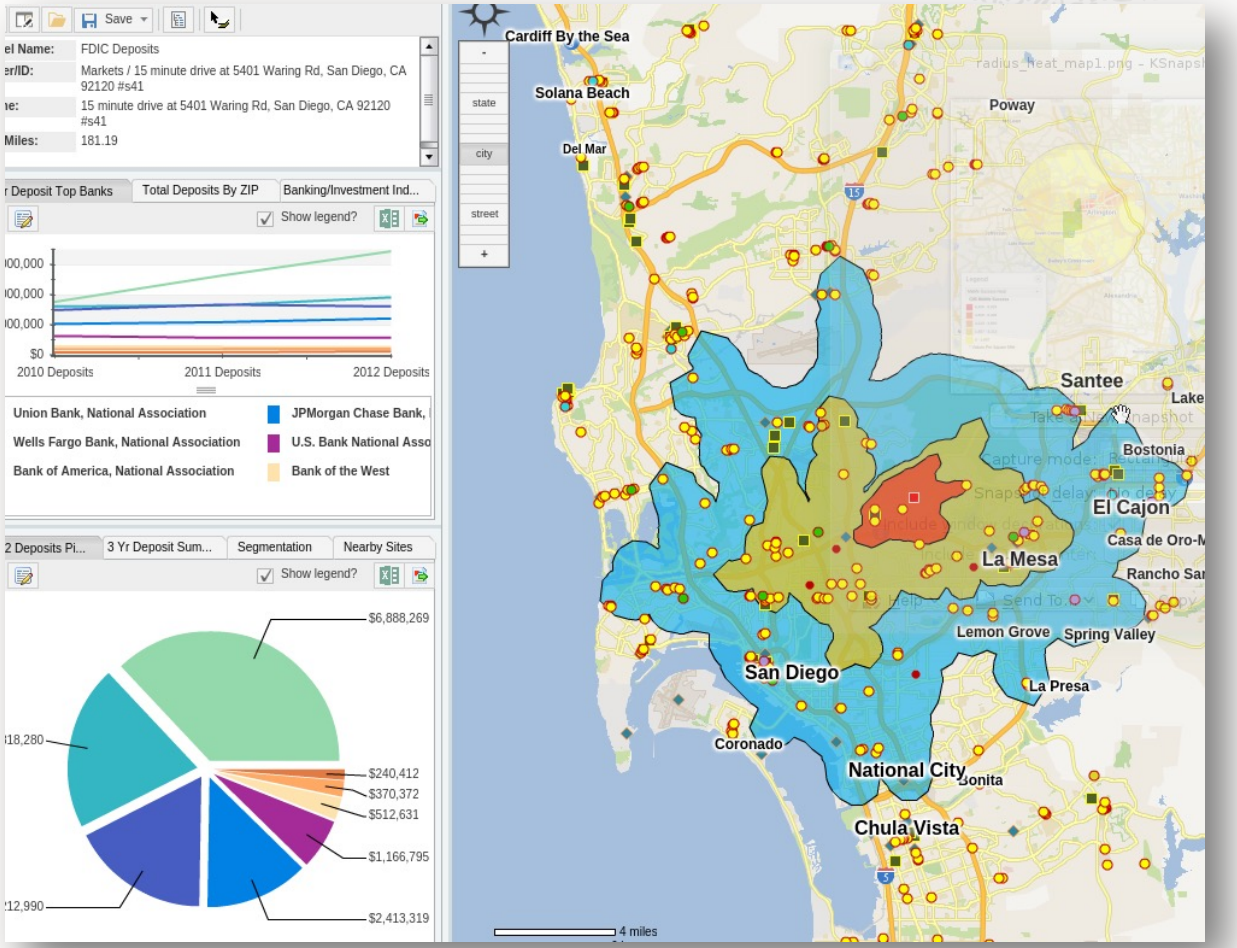




Using NDM for Drive-time Calculations

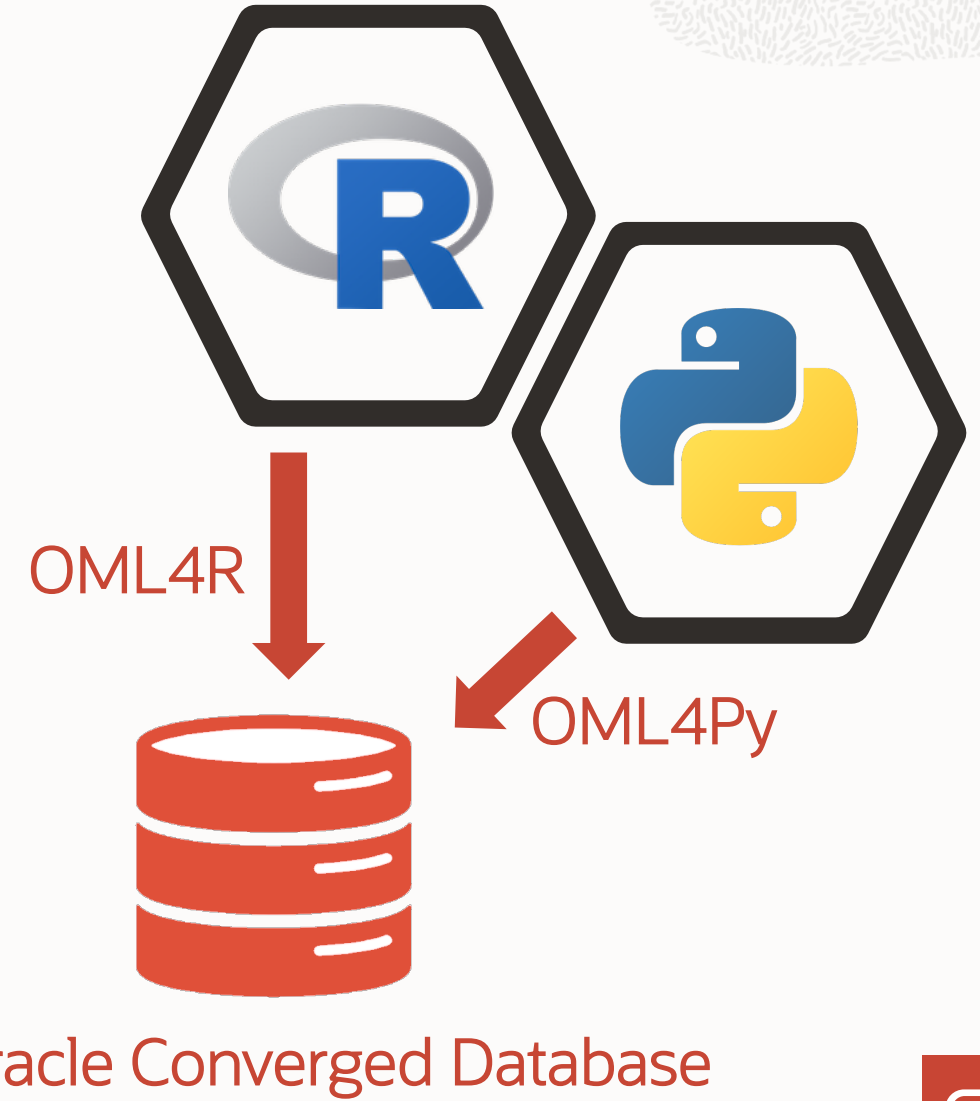
Network Data Model for site planning:

- Drive-time calculations on road network
- Measure accessibility
- Combine with other indicators

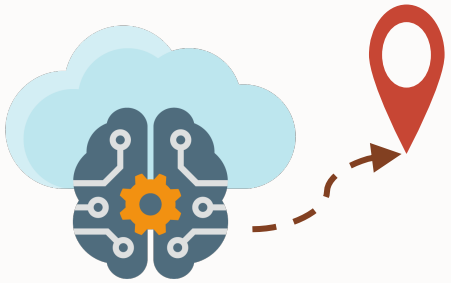


Why data scientists and data analysts use R and Python

- Powerful
- Extensible
- Graphical
- Extensive statistics
- Ease of installation and use
- Rich ecosystem
  - 1000s of open source packages
  - Millions of users worldwide
- Heavily used by data scientists
- Free

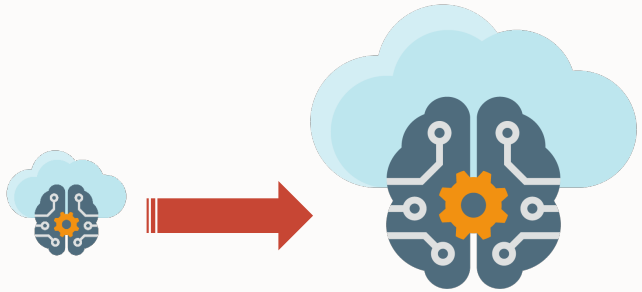


Oracle Machine Learning Key Attributes



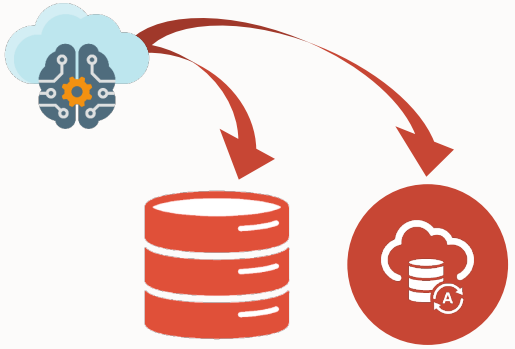
Automated

Get better results faster with less effort – even non-expert users



Scalable

Handle big data volumes using parallel, distributed algorithms – no data movement

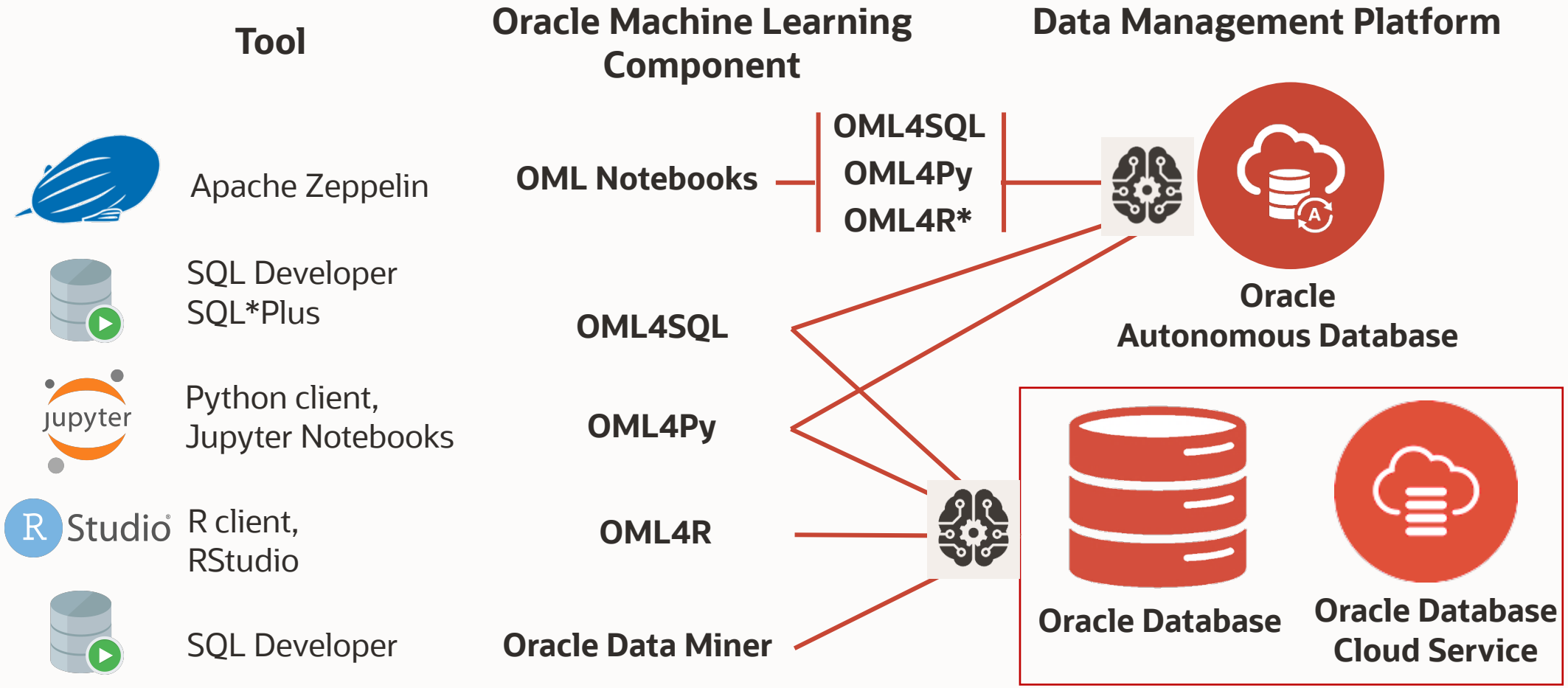


Production-ready

Deploy and update data science solutions faster with integrated ML platform

Increase productivity | Achieve enterprise goals | Innovate More

Oracle Machine Learning interfaces to Oracle Database



* coming soon

Oracle Machine Learning Algorithms and Analytics

- **Classification**
  - Naïve Bayes
  - Logistic Regression (GLM)
  - Decision Tree
  - Random Forest
  - Neural Network
  - Support Vector Machine (SVM)
  - Explicit Semantic Analysis
- **Clustering**
  - Hierarchical K-Means
  - Hierarchical O-Cluster
  - Expectation Maximization (EM)
- **Anomaly Detection**
  - One-Class SVM
- **Time Series**
  - Forecasting - Exponential Smoothing
  - Includes popular models e.g. Holt-Winters with trends, seasonality, irregularity, missing data

Regression
- Linear Model
- Generalized Linear Model (GLM)
- Support Vector Machine (SVM)
- Stepwise Linear regression
- Neural Network
- LASSO

Attribute Importance
- Minimum Description Length
- Principal Component Analysis (PCA)
- Unsupervised Pair-wise KL Div
- CUR decomposition for row & AI

Association Rules
- A priori/ market basket

- **Predictive Queries**
  - Predict, cluster, detect, features
- **SQL Analytics**
  - SQL Windows
  - SQL Patterns
  - SQL Aggregates

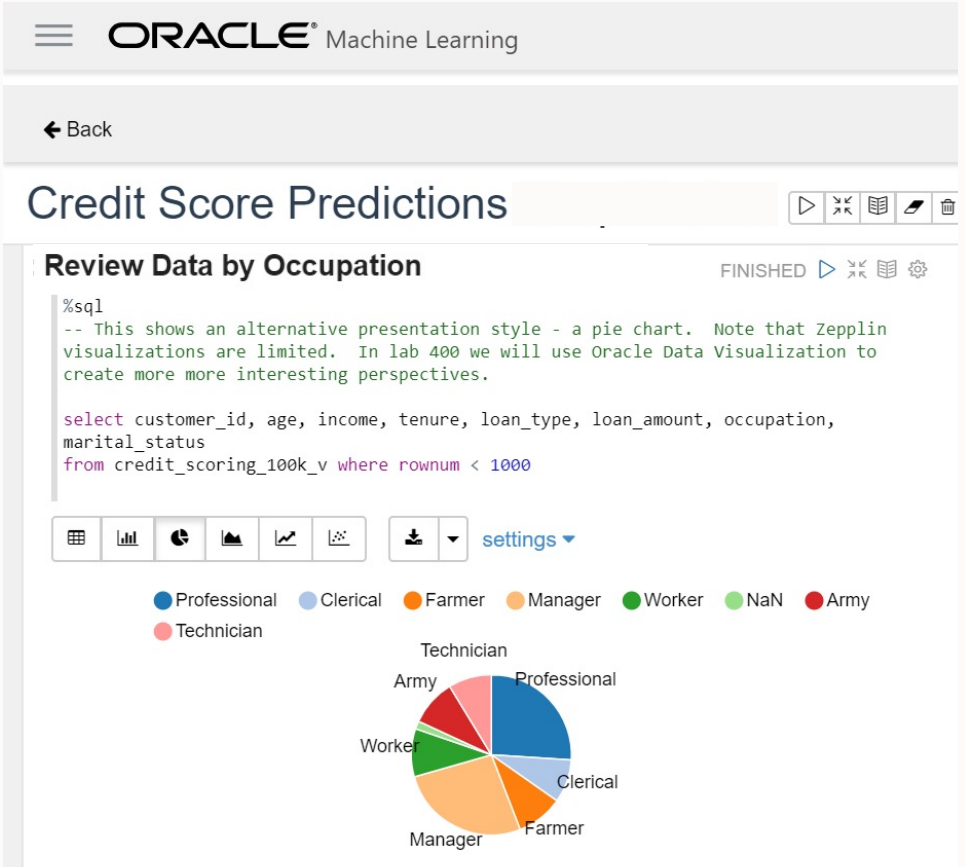
XGBoost
MSET

- **Feature Extraction**
  - Principal Comp Analysis (PCA)
  - Non-negative Matrix Factorization
  - Singular Value Decomposition (SVD)
  - Explicit Semantic Analysis (ESA)
- **Text Mining Support**
  - Algorithms support text columns
  - Tokenization and theme extraction
  - Explicit Semantic Analysis (ESA) for document similarity
- **Statistical Functions**
  - Basic statistics: min, max, median, stdev, t-test, F-test, Pearson's, Chi-Sq, ANOVA, etc.
- **R and Python Packages**
  - Third-party R and Python Packages through Embedded Execution
  - Spark MLlib algorithm integration

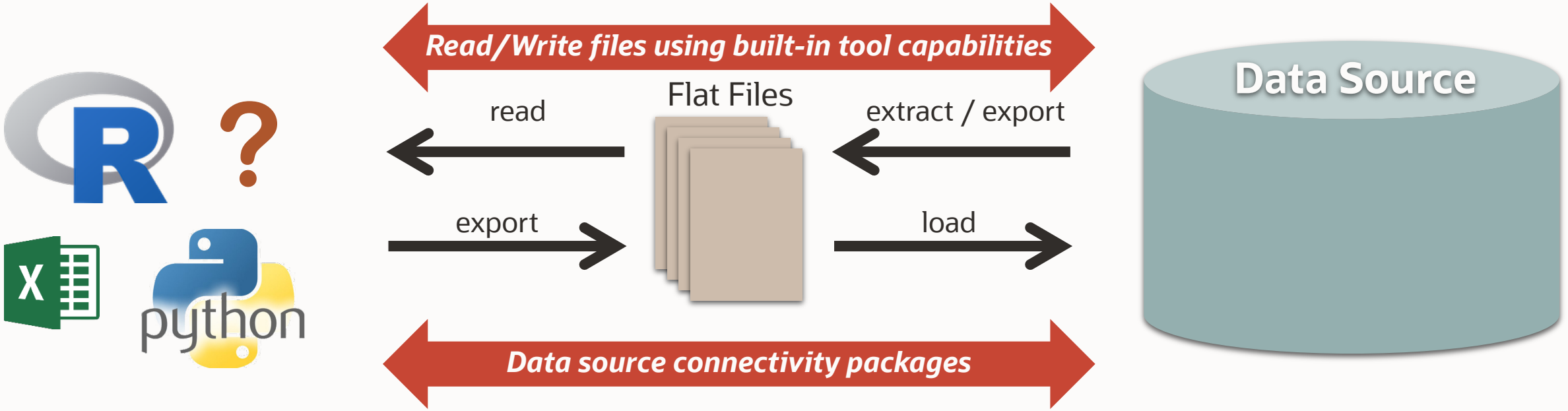
Oracle Machine Learning Notebooks

Autonomous Database as a Data Science Platform

- **Collaborative UI**
  - Based on Apache Zeppelin
  - Supports data scientists, data analysts, application developers, DBAs
  - Easy sharing of notebooks and templates
  - Edits made in one notebook immediately appear in other open shared notebooks
  - Permissions, versioning, and execution scheduling
- **Included with Autonomous Database**
  - Automatically provisioned, managed, backed up
  - In-database SQL algorithms and analytics functions
  - Explore and prepare, build and evaluate models, score data, deploy solutions
  - Python available!



Traditional Analytics and Data Source Interaction

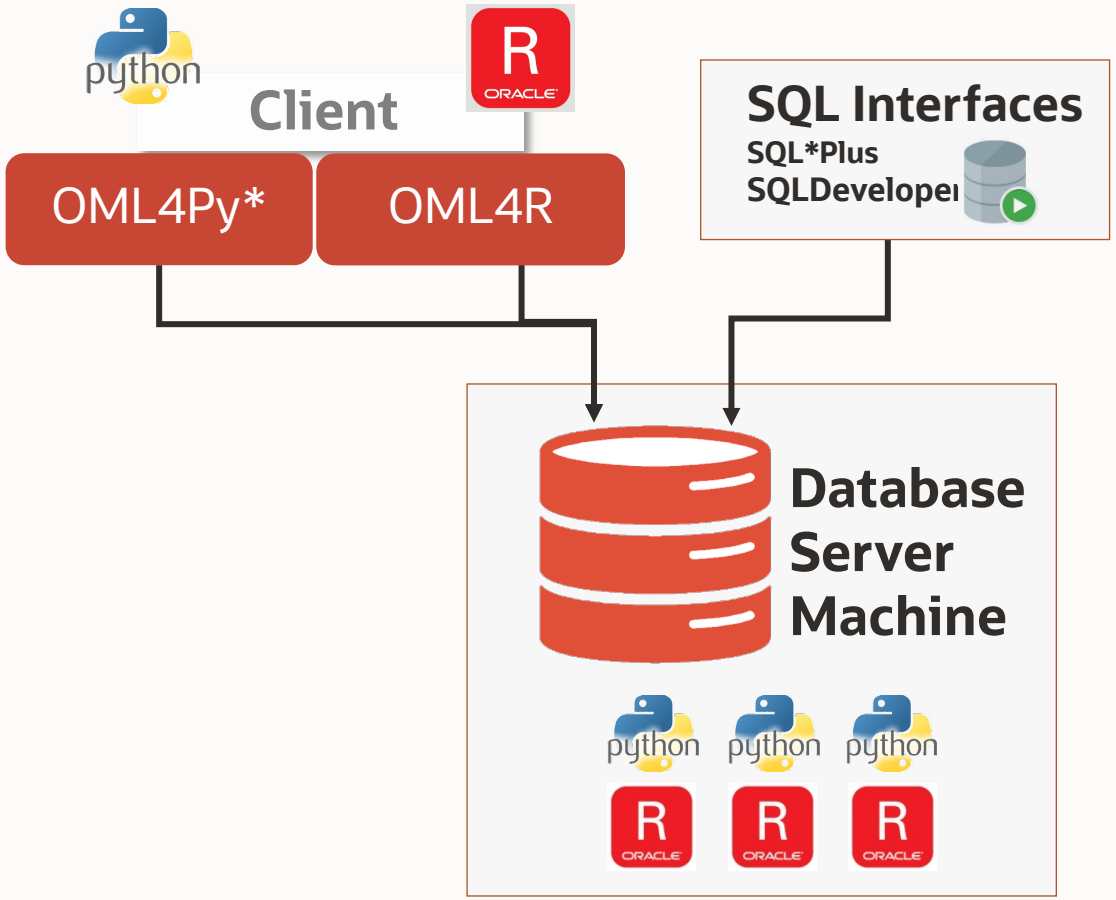


Deployment
Ad hoc
cron job

- **Access latency**
- **Paradigm shift: R/Python → *Data Access Language* → R/Python**
- **Memory limitation – data size, in-memory processing**
- **Single threaded**
- **Issues for backup, recovery, security**
- **Ad hoc production deployment**

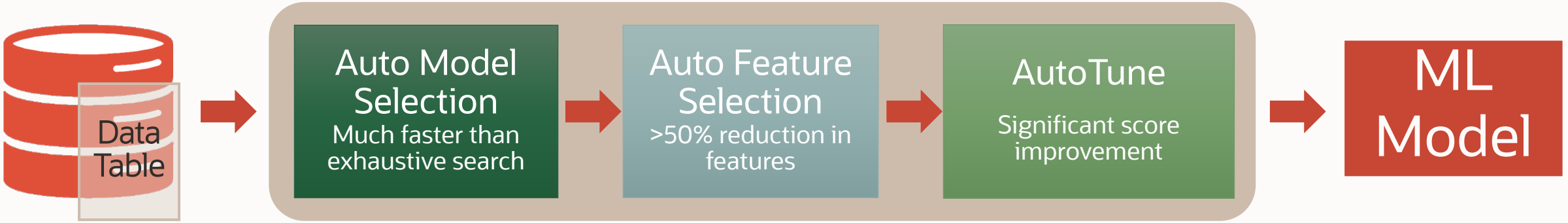
Oracle Machine Learning for R and Python

- **Transparency layer**
  - Leverage proxy objects so data remain in database
  - Overload native functions translating functionality to SQL
  - Use familiar R / Python syntax on database data
- **Parallel, distributed algorithms**
  - Scalability and performance
  - Exposes in-database algorithms available from OML4SQL
- **Embedded execution**
  - Manage and invoke R or Python scripts in Oracle Database
  - Data-parallel, task-parallel, and non-parallel execution
  - Use open source packages to augment functionality
- **OML4Py AutoML**
  - Model selection, feature selection, hyper-parameter tuning
  - Supports Classification and Regression



AutoML – *new* with OML4Py

Increase data scientist productivity – reduce overall compute time



Auto Model Selection

- Identify in-database algorithm that achieves highest model quality
- Find best model faster than with exhaustive search

• Auto Feature Selection

- Reduce # of features by identifying most predictive
- Improve performance and accuracy

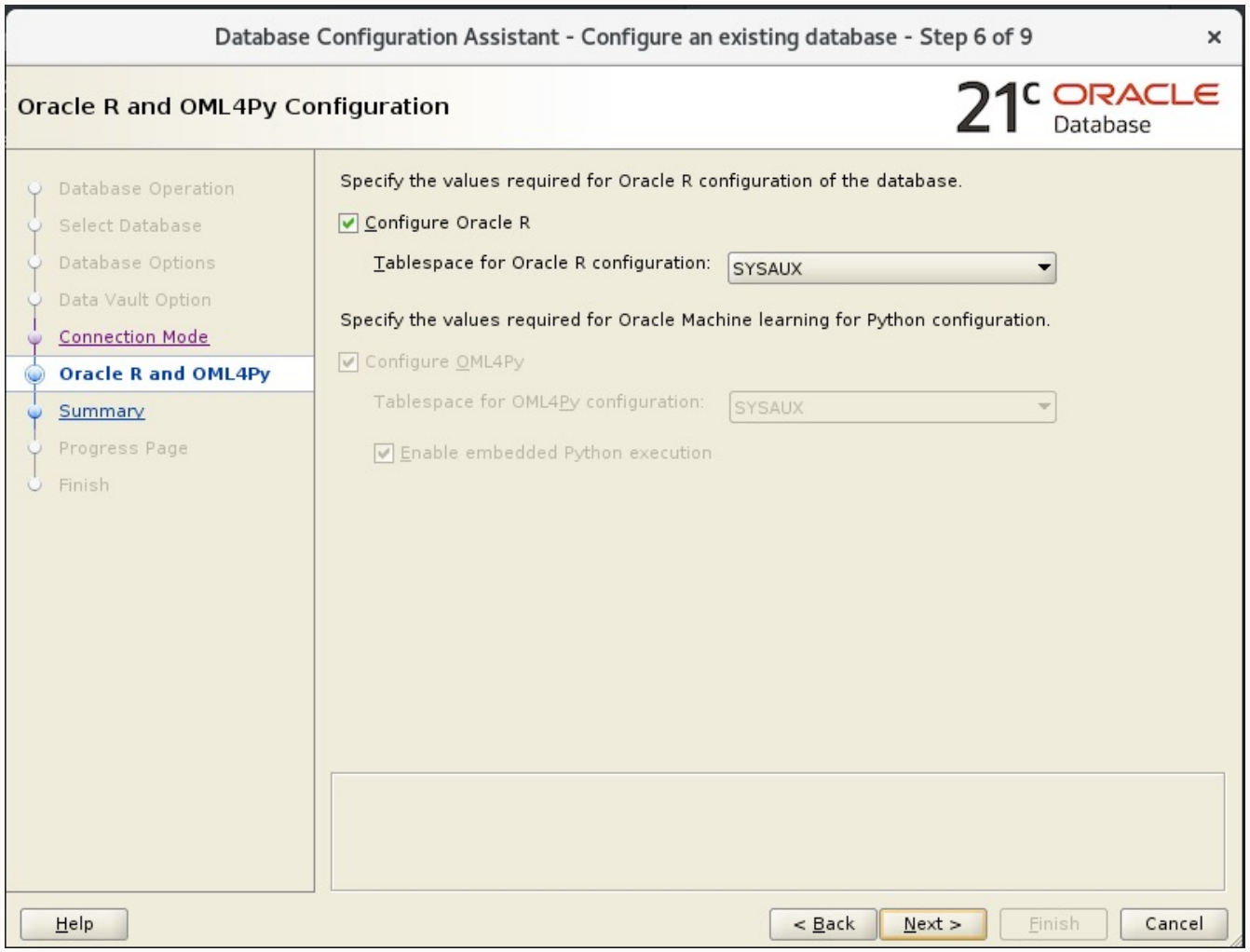
Auto Tune Hyperparameters

- Significantly improve model accuracy
- Avoid manual or exhaustive search techniques

Enables non-expert users to leverage Machine Learning

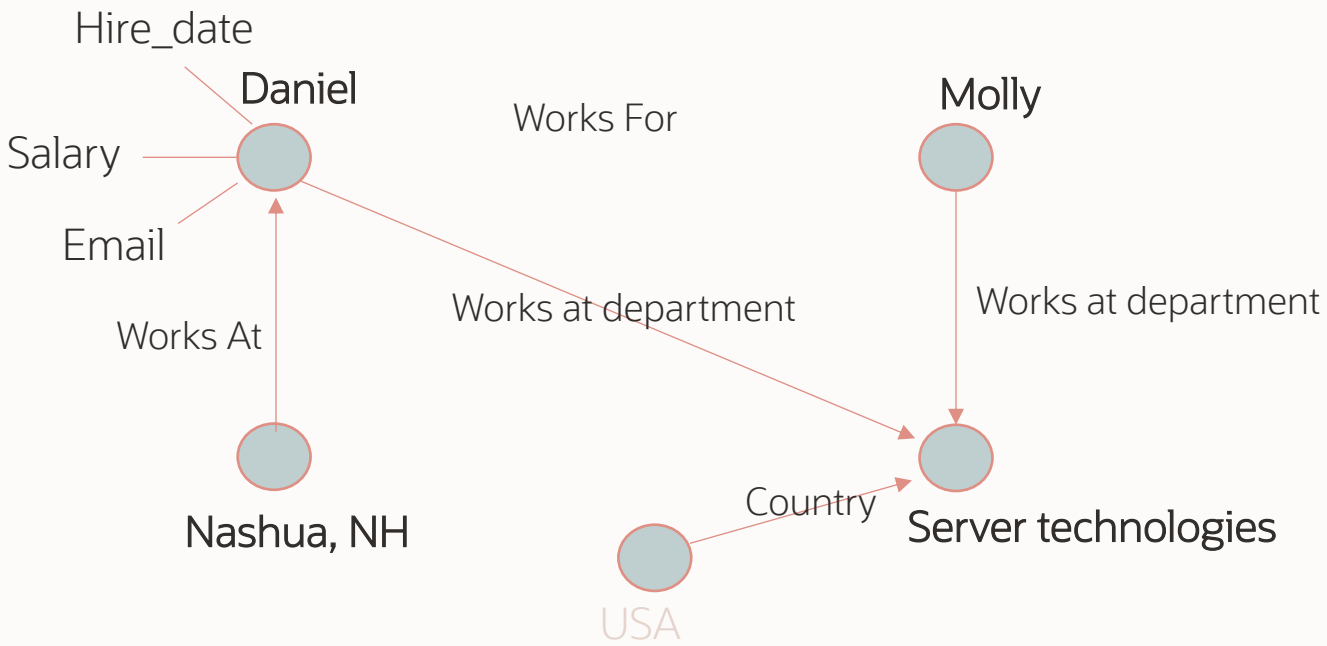
Easy to set up Now in 21c!

Configure R and Python from Database Configutation Assistant



Graph Analytics

- **Analytics based on connections and relationships between data entities**



What is Graph Analytics?

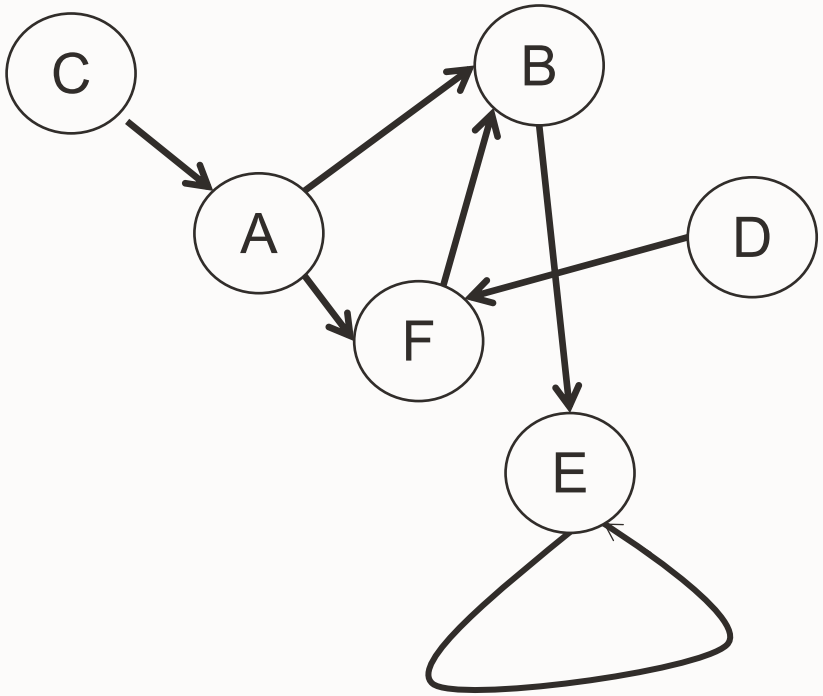
A labeled-property graph model is represented by a set of nodes, edges, properties, and labels.

What is a graph?

- Data model representing entities as **vertices** and relationships as **edges**
- Optionally including attributes

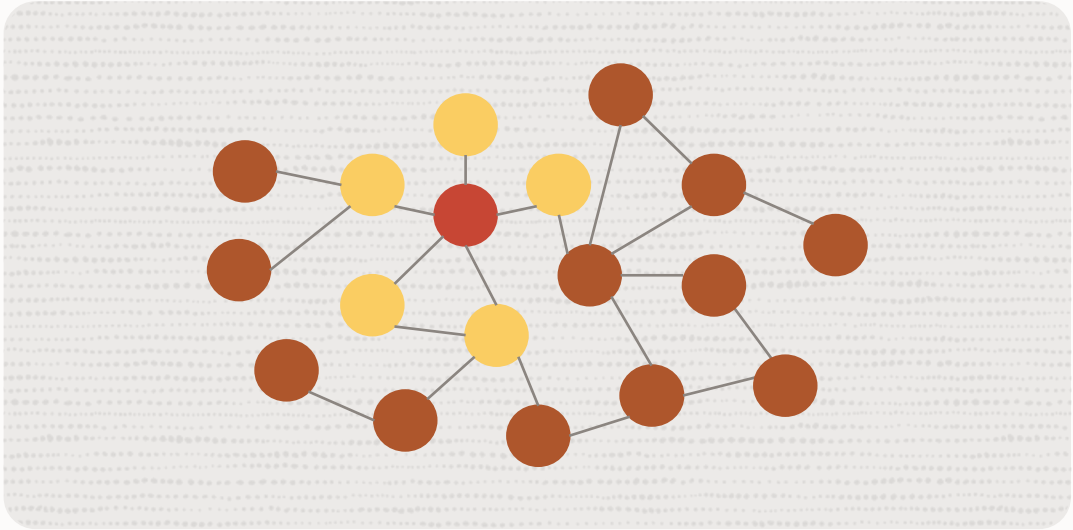
What are typical graphs?

- Social Networks
  - LinkedIn, Facebook, Google+, Twitter, …
- Physical networks, Supplier networks,…
- Knowledge Graphs
  - Apple SIRI, Google Knowledge Graph, …

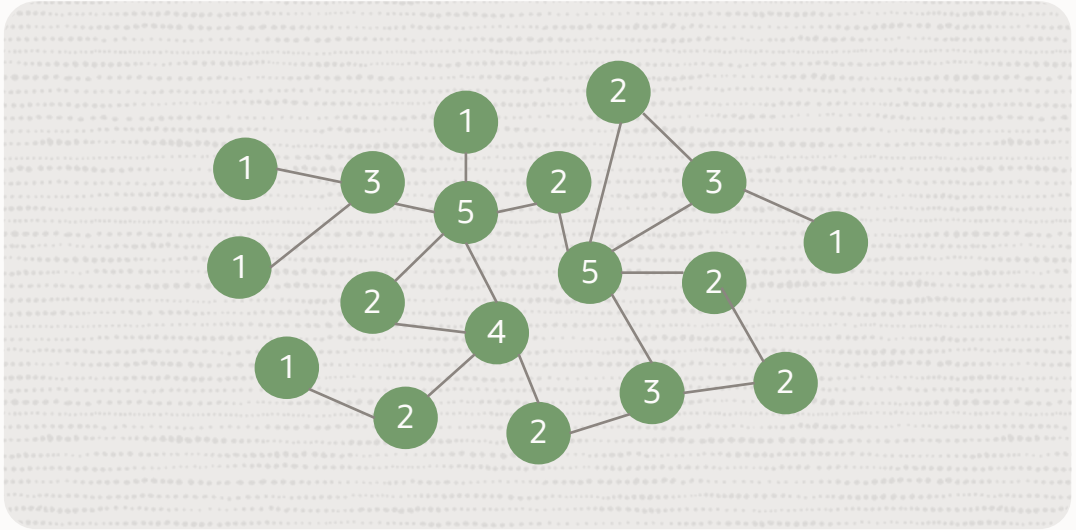


Two types of processing in graph analytics

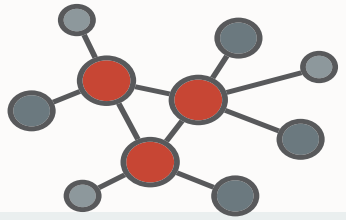
- **Query processing**
- Search for surrounding nodes
- Traverse property paths
- Pattern matching
- Extract sub-graphs



- **Executing graph algorithms**
- Rank importance of nodes
- Detect components and clusters
- Evaluate structure of communities
- Shortest paths



From tables to Property Graphs



PRODUCT_ID	BOUGHT_WITH
0	1
0	2
0	4
1	0
1	12
1	23
…	…

PGQL DDL SYNTAX:

```
CREATE PROPERTY GRAPH products

VERTEX TABLES (
    PRODUCTS KEY(PRODUCT_ID) PROPERTIES (PRODUCT_ID)
    )
EDGE TABLES(
    SOURCE KEY(PRODUCT_ID) REFERENCES PRODUCTS
    DESTINATION KEY(BOUGHT_WITH) REFERENCES PRODUCTS
)
```

- Every product id is a vertex
- Two vertices in one row are connected by an edge
- (“bought_with” relationship)

Property Graph Product Overview

Store, manage, query and analyze graphs

- Enterprise capabilities: Built on Oracle Infrastucture
- Manageability, fine-grained security, high availability, integration and more

High scalable

- In-memory query and analytics and in-database query
- 10s of billions of edges and vertices

PGQL: Powerfull SQL-like graph query language

Analytics Java API: 50+ pre-built graph analytics algorithms

Visualization:

- Light-weight web application: UI accesible from a browser

Oracle Database as a Graph Store

Database stores and manages Graph Nodes, Edges and Properties

Database provides graph traversal and query language and API's

- Java API to develop applications
- Command-line submission of graph queries
- Graph visualization tool
- APIs to update graph store
- PGQL language for Property Graph
- SPARQL language for RDF Triple Store

PGQL Graph Query Language

Graph patter matching

(person)-[:works_for]->(person)

Basic patterns and reachability patterns

Can we reach from A to B with an arbitrary number of hops?

Familiarity of SQL users

- Similar language construct and syntax

  SELECT … WHERE …

  GROUP BY … ORDER BY …

- ‘Result set’ (table) as output

PGQL Graph Query

```
1 SELECT n, n0, n1, e0, e1, e2, n.pagerank, n0.pagerank, n1.pagerank
2 MATCH (n)-[e0]-(n0)-[e1]-(n1), (n)-[e2]-(n1)
3 WHERE ID(n0) = 'IRON MAN/TONY STARK '
4 ORDER BY n.pagerank DESC, n0.pagerank DESC, n1.pagerank DESC LIMIT 30
```

Graph

PGQL Graph Pattern: MATCH Clause Syntax

- Use () to represent vertices
- (a:person) variable a, label person
- Use [] to represent edges
- [e:knows] variable e, label knows
- Edge patterns
- -[]-> outgoing edge
- <-[]- incoming edge
- -[]- anydirected edge
- Variable-length paths
- -/ ... */-> zero or more hops
- -/ ... +/-> one or more hops
- -/ ... {2,4}/-> two to four hops
- Use ON to specify graph (optional)

```
SELECT a,b,e FROM
MATCH (a)-[e]-(b)
ON graph
```

```
SELECT a,b,e FROM
MATCH (a)-[e]->(b)
ON graph
```

```
SELECT a,b,e FROM
MATCH (a:ACCOUNT)-[e:TRANSFER]->(b)
ON graph
```

```
SELECT a,b FROM
MATCH (a:ACCOUNT)-/:TRANSFER*/->(b)
ON graph WHERE a.account_no=4711
```

```
SELECT a,b,c,e1,e2,e3 FROM
MATCH (a)-[e1]->(b)-[e2]->(c)-[e3]->(a)
ON graph
```

PGQL Graph Pattern: More advanced examples

- Path queries and reachability
- -/ … /-> returns endpoint vertices only
- ANY retrieves vertices or edges along path (new in PGQL 1.4 specification)
- Shortest path/cheapest path
- Any, all, or top <k> shortest paths
- Cheapest path uses cost function
- ARRAY_AGG allows output of properties along path
- Vertex/Edge functions, eg.
- In_degree()
- Out_degree()
- Label()

```
SELECT dst.number, ARRAY_AGG(e.amount),
SUM(e.amount) FROM
MATCH ANY (src:ACCOUNT) -[e]->+ (dst:ACCOUNT)
WHERE src.number = 4711
```

```
SELECT src, SUM(e.amount), dst FROM
MATCH SHORTEST ( (src) -[e]->* (dst) )
WHERE src.create_date < dst.create_date
```

```
SELECT src, SUM(e.amount), dst FROM
MATCH TOP 3 SHORTEST ( (src) -[e]->* (dst) )
WHERE src.create_date < dst.create_date
```

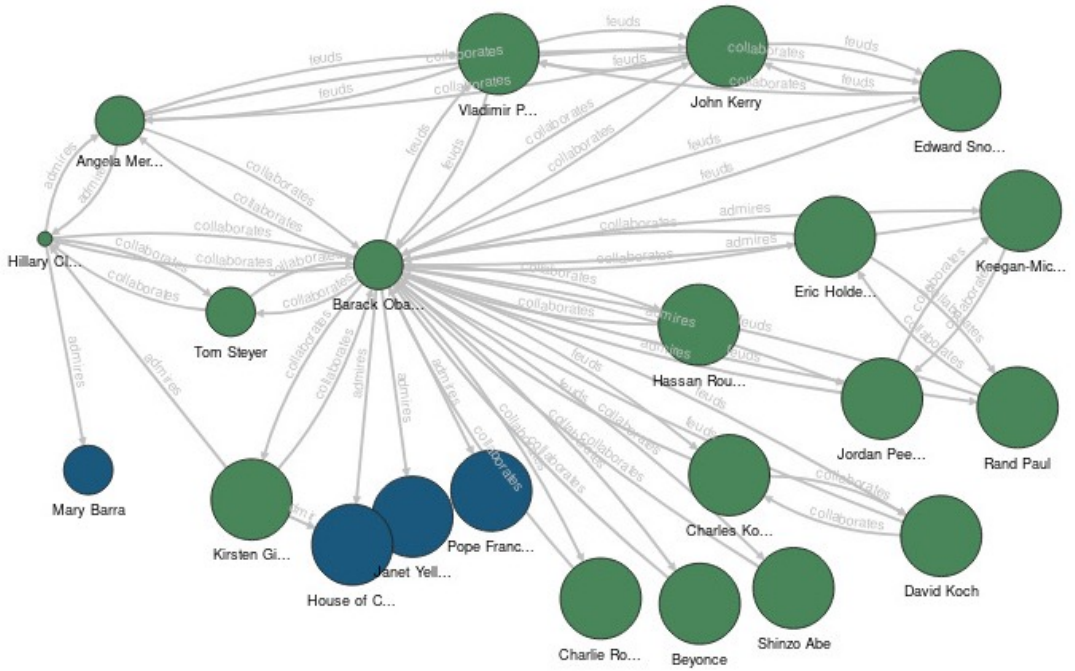
```
SELECT src, out_degree(src) AS num_txns FROM
MATCH (src:ACCOUNT) -[e]-> (dst:ACCOUNT)
ORDER BY num_txns DESC
```


GraphViz Tool

PGQL Graph Query

```
1 SELECT m,n,e
2 FROM MATCH (m)-[e]->(n)
3 where n.distance<=2 and m.distance<=2
```

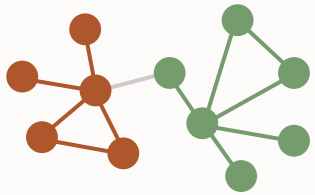
Graph: CONNEC... Parallelism: 0



Visualize PGQL query results

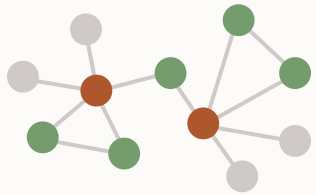
- Pre-loaded and Published graphs
- Themes, styles, layouts
- Interactive Graph manipulation

Graph analytics: 60+ parallelized, in-memory algorithms out-of-the-box



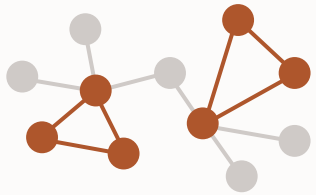
Detecting components and communities

Strongly Connected Components,
Weakly Connected Components,
Label Propagation,
Conductance Minimization,
Infomap



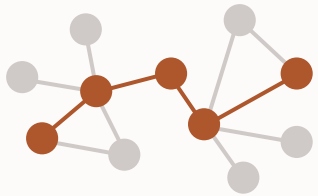
Ranking and walking

PageRank, Personalized PageRank,
Degree Centrality, Closeness Centrality,
Vertex Betweenness Centrality,
Eigenvector Centrality, HITS, SALSA,
Random Walk with Restart



Evaluating structures

Adamic-Adar Index, Conductance,
Cycle Detection, Degree Distribution,
Eccentricity, K-Core, LCC, Modularity,
Reachability Topological Ordering,
Triangle Counting



Path-finding

Shortest Path (Bellman-Ford, Dijkstra,
Bidirectional Dijkstra), Fattest Path,
Compute Distance Index,
Enumerate Simple Paths,
Fast Path Finding, Hop Distance

Link prediction and others

Twitter Whom-to-follow
Minimum Spanning-Tree,
Matrix Factorization

Machine learning

DeepWalk *
Supervised GraphWise *
Pg2Vec *

*on-premises, not yet in Graph Studio

Interaction with the Property Graph

- Access through APIs
  - Implementation of Apache Tinkerpop Blueprints APIs
  - Based on Java, REST plus SolR Cloud/Lucene support for text search
- Scripting
  - Groovy, Python, JavaScript, …
  - Apache Zeppelin integration, JavaScript (node.js) language binding
- Graphical UIs
  - Commercial tools such as TomSawyer Perspectives
  - Vis.js and D3 among others

Agenda

Converged Database Workshop Series

Converged Database Workshop Series

1. Oracle Converged Database: Multitenant, Multimodel, In-Memory
   - For DBAs, Solutions Architects and Developers, including CTOs

2. Oracle Converged Database: Multicloud ECX with Autonomous DB
   - For Data Engineers and Cloud Solutions Architects

3. Oracle Converged Database: Spatial, Graph & ML with Python and R
   - For Data Engineers, Data Scientists, Business Analysts and Solutions Architects

4. Oracle Converged Database: Security
   - For DBAs, Solutions Architects and CISOs

5. Oracle Converged Database: Continuous Availability
   - For DBAs, Solutions Architects and CTOs

Agenda

Machine Learning, Spatial and Graph

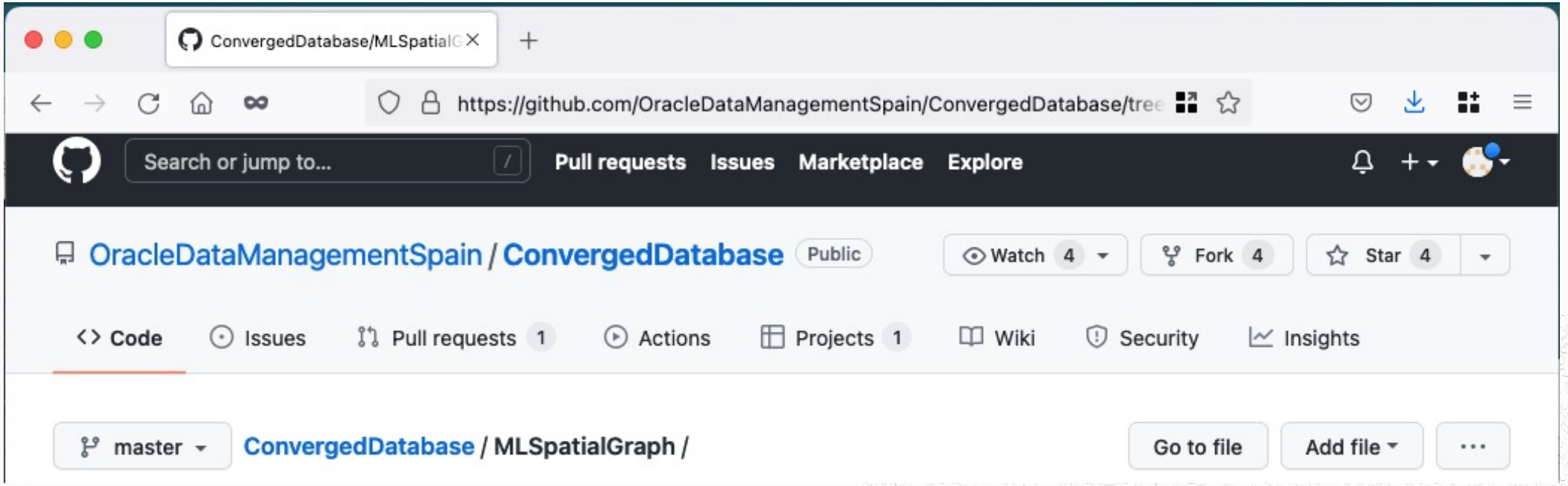
Oracle Converged Database: Machine Learning, Spatial and Graph Workshop

Hands On Labs

HOL0 – To Be Uploaded (Spatial Web Services) – Not included
HOL1 – Spatial and Spatial Studio
HOL2 – Machine Learning with Python and R
HOL3 - Graph

Materials and HOL manuals

https://github.com/OracleDataManagementSpain/ConvergedDatabase/tree/master/MLSpatialGraph



Cloud Free Tier

Cuenta Cloud gratuita

https://signup.oraclecloud.com/

500$ en créditos Cloud gratuitos

Durante 30 días

+

Servicios Always Free

Encuesta

¡Tu opinión es muy importante!

ENCUESTA

ORACLE

Workshop Virtual BD Convergente:
Machine Learning, Spatial y Graph

¡Tu opinión es muy importante!

Escanea el QR para responder o entra aquí:

https://bit.ly/3HM2x9N

Inspiration & Innovation

Our mission is to help people see data in new ways, discover insights, unlock endless possibilities.

Thank you!

Oracle Spain