

ORACLE

Deploy a Microservice Application connected to an Oracle DB

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.

Oracle Database Product Management – Oracle Platform Technology Solutions (PTS)

Who We Are

- Part of Oracle Server Technologies Development division

- Primary responsibility is technology enablement NOT a sales function

- Enable partners to rapidly, successfully, and efficiently adopt Oracle server technologies

Value to Partners

- In-depth architectural/technology reviews

- Upgrading/Migrating to latest releases and new features

- Validation support (onsite or in Oracle lab)





Pedro Torres

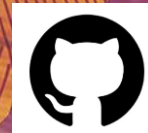
PTS – Platform Technology Solutions
Global Team – NAS, LAD, EMEA, APAC

OCP 11g, 12c, RAC Expert, 19c ATP.

LAD – Mexico City

+18 years in IT

+16 DB Solutions



<https://github.com/Ptorresr>



[@PedroToRdz](https://twitter.com/PedroToRdz)



[/in/pedro-torres-rdz](https://in/pedro-torres-rdz)



Pedro.p.Torres@oracle.com



\$ who -u



Alexandre Fagundes

alexandre.af.fagundes@oracle.com

LAD Technical Partner Advisor

20+ years in IT Industry
14+ years of Oracle (3 seasons)

- Cloud Architect
- Database Administrator
- EBS Applications DBA
- Support Engineer
- Partner Enablement

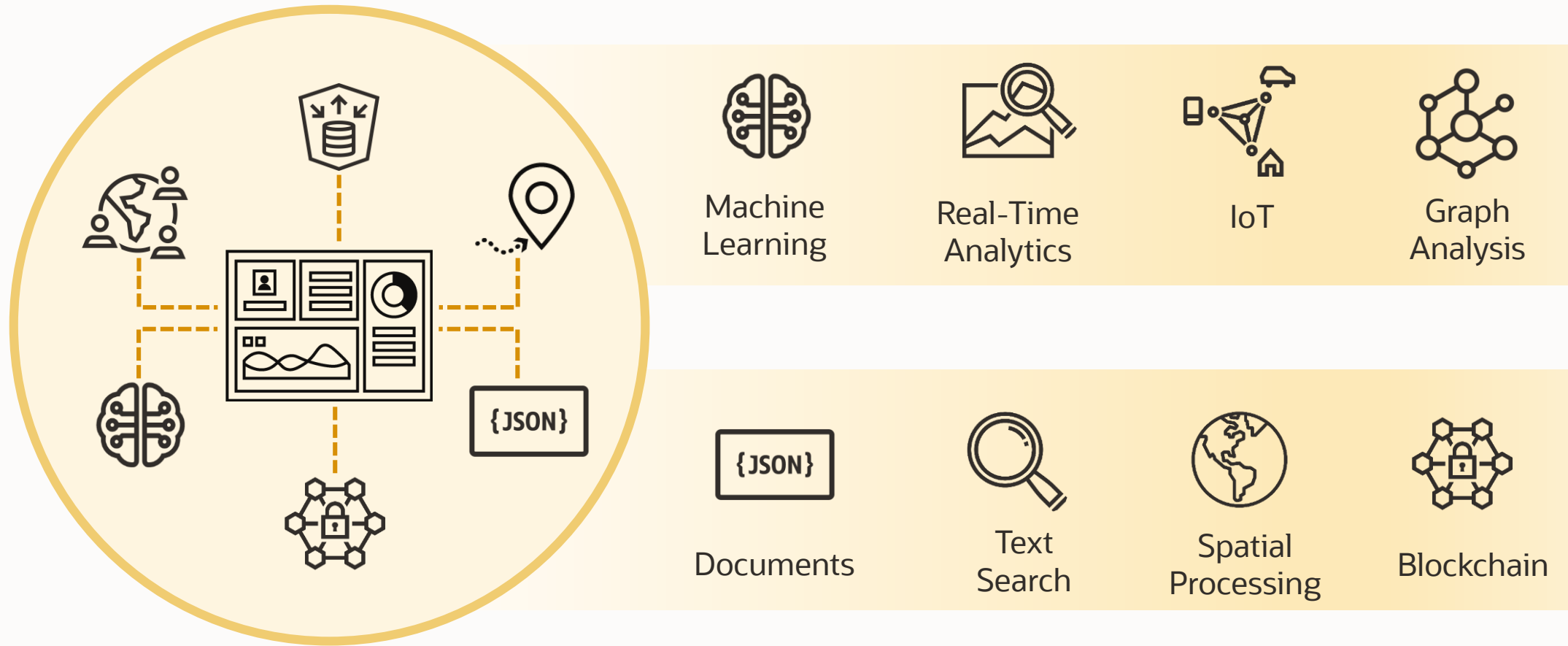


 /alexandre-b-fagundes

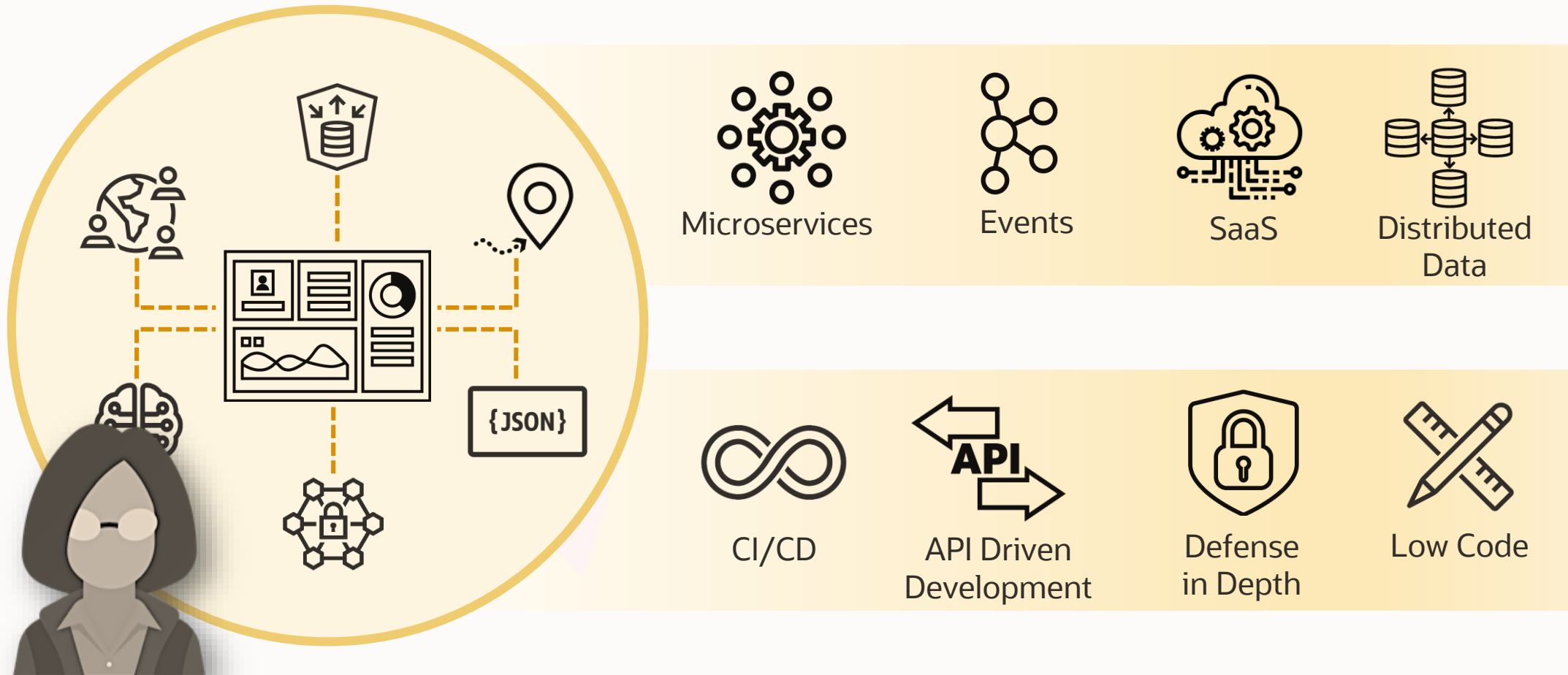


Data Driven Apps **Create Value from Data in New Ways**

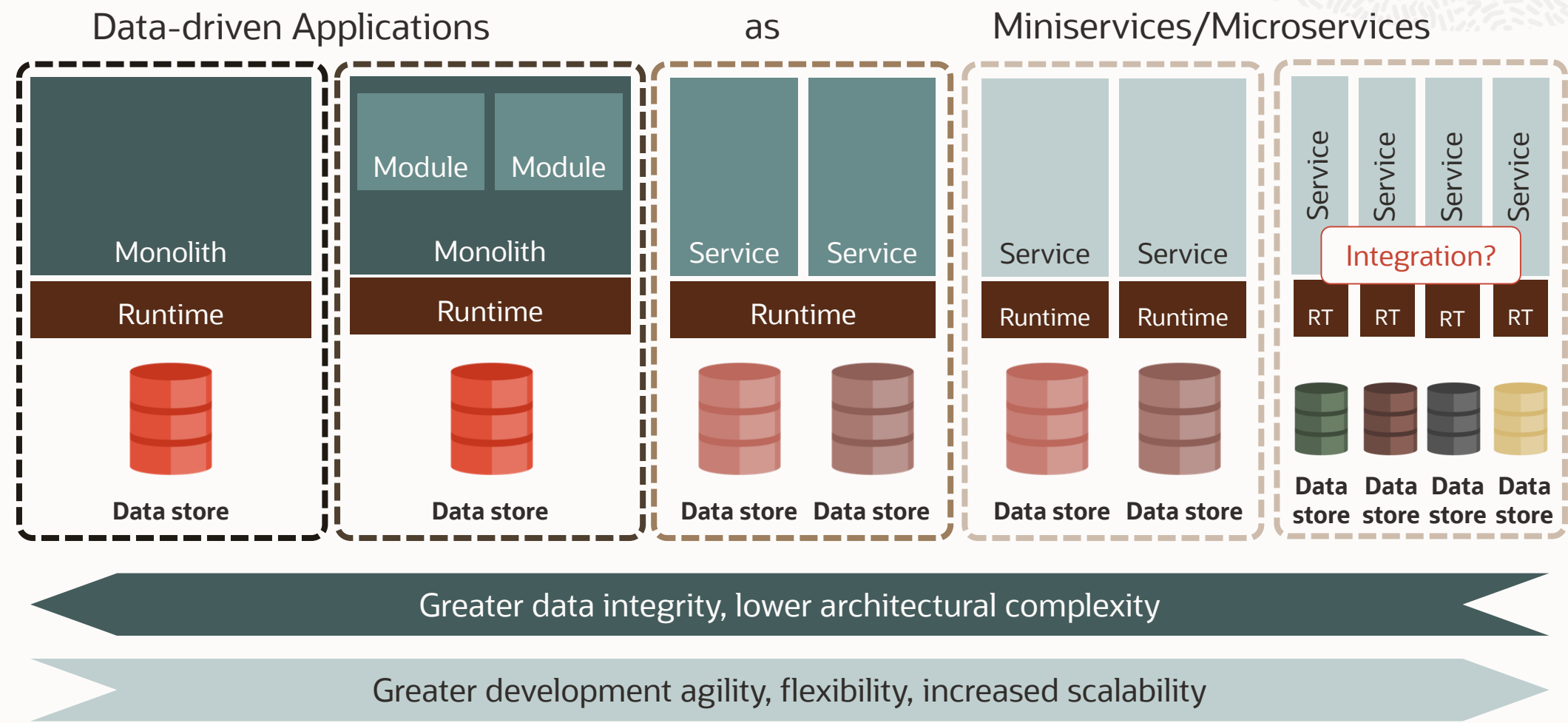
Data Driven Apps **Create Value from Data in New Ways**



Data Driven Apps are **Built Using New Dev Paradigms**



Modernize the Monolith

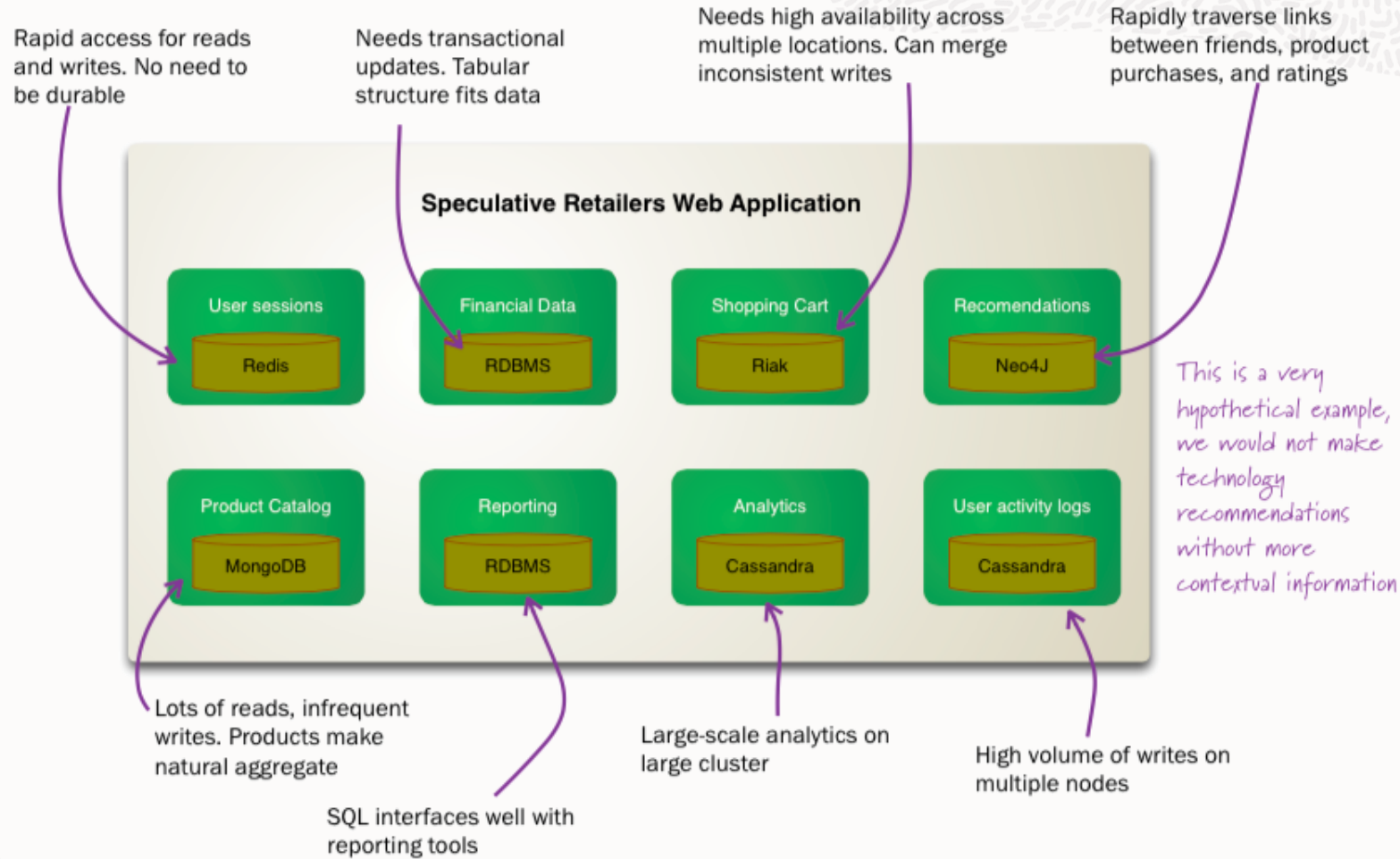


Different Services have Different Data-Store Needs

		User Sessions	Financial Transactions	Insurance Claims	Recommend Engine	Product Catalog	Reporting	Analytics	Activity Logs
Processing	Heavy Writes								✓
	Heavy Reads				✓	✓	✓	✓	
	Fast Read/Write	✓							
	Data Consistency		✓						
	Data Durability		✓						
	Analytic						✓	✓	
	Graph				✓			✓	
	Spatial							✓	
	Geo Distribution			✓		✓			✓
Data store	Relational		✓				✓	✓	
	Key/Value	✓		✓					✓
	Document/JSON					✓			✓
	Graph				✓				



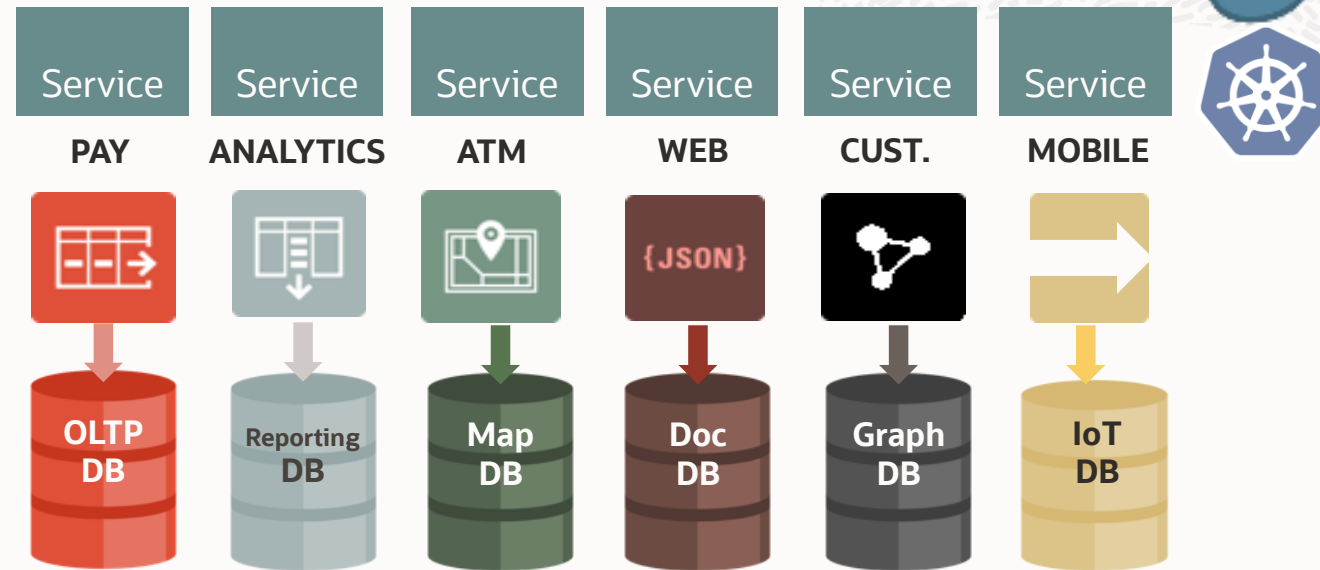
what might Polyglot Persistence look like?



8

Source: The future is: ~~NoSQL~~ Databases Polyglot Persistence
<http://martinfowler.com/articles/nosql-intro-original.pdf>

Data-Stores for Microservices

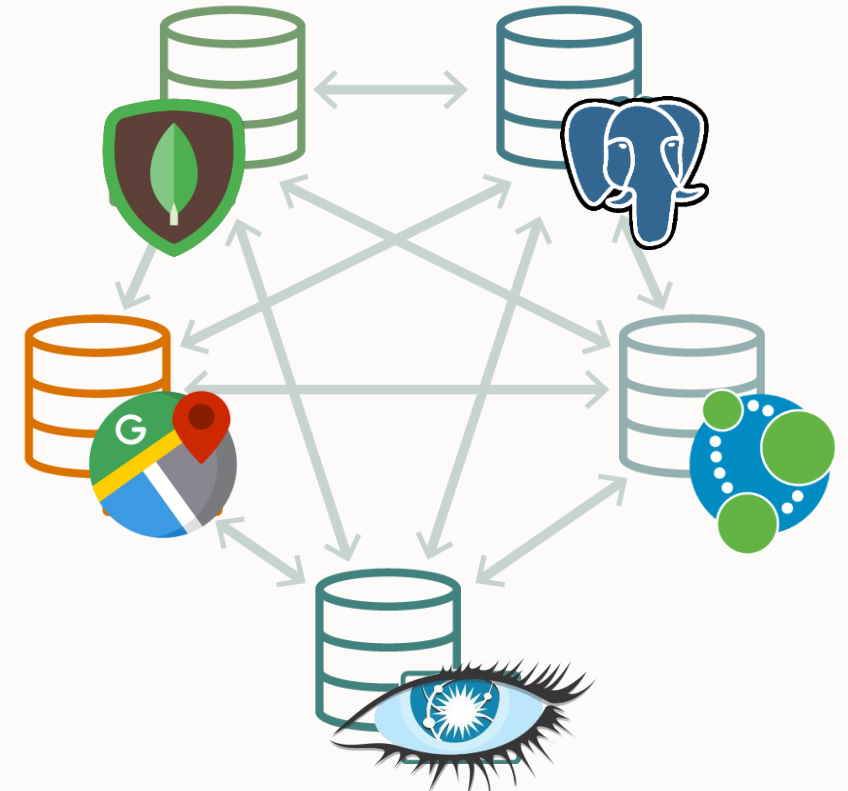


“Polyglot persistence will occur over the enterprise as different applications use different data storage technologies. It will also occur within a single application as different parts of an application’s data store have different access characteristics.”

Martin Fowler & Pramod Sadalage, Feb. 2012
<http://martinfowler.com/articles/nosql-intro-original.pdf>

Data Fragmentation from **Single-Purpose** Databases

- Each single-purpose database stores a **fragment** of your enterprise data, requiring data to be **moved** around and **transformed**
- **Integrating** fragmented data makes application development and analytics extremely **complex**
- **Operating** many unique databases is inherently **complex, expensive, and risky**



In contrast, the converged approach is inherently simpler, lower cost, and lower risk.

Polyglot Persistence Integration Nightmare

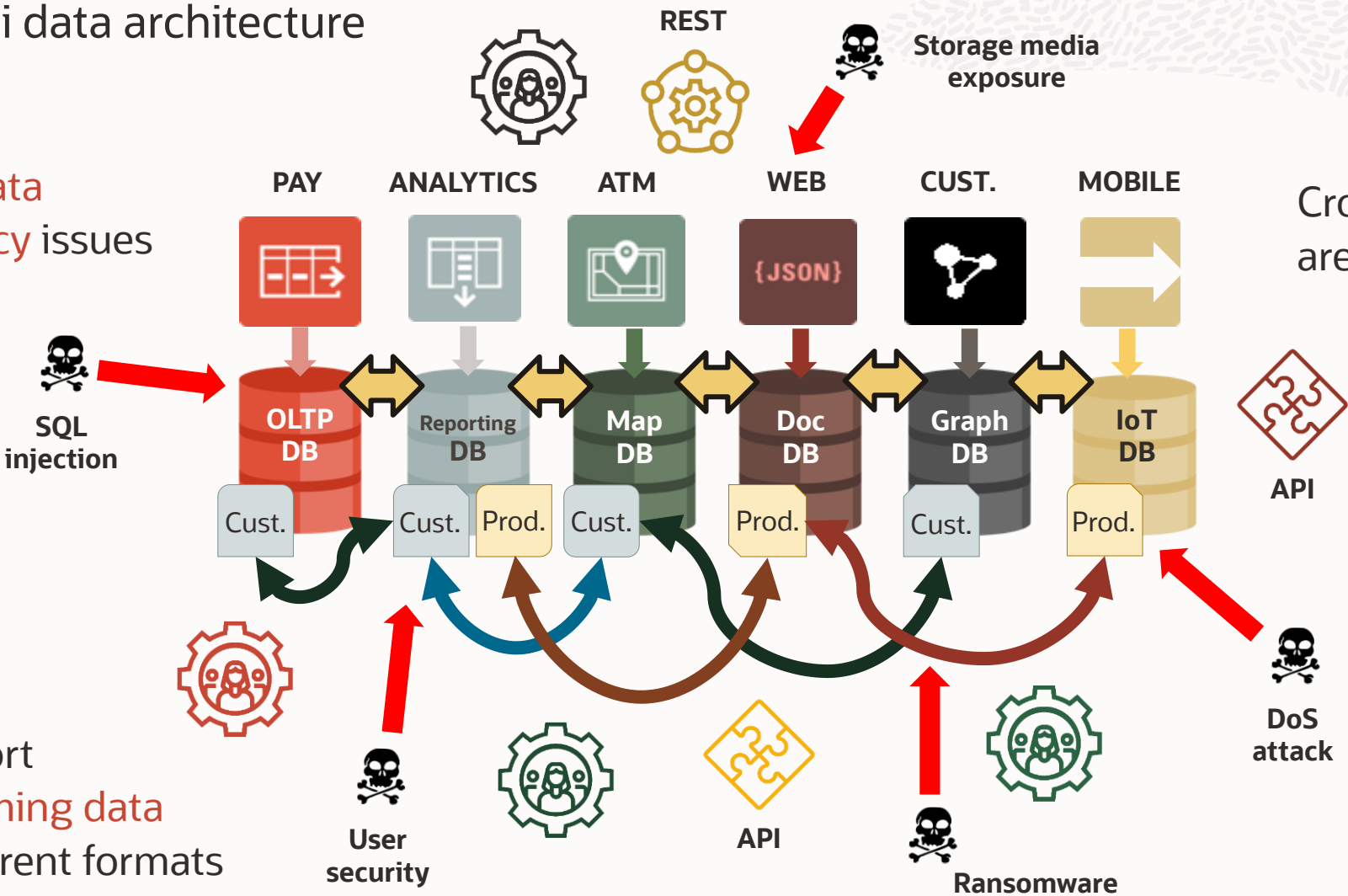
Spaghetti data architecture

Serious **data consistency** issues

Cross-domain reports are **costly** and very **slow**

High effort **transforming data** into different formats

Lacks coherence of **availability**, **durability**, and **security** concepts



Approaches | Single-Purpose vs Converged

Amazon & Niche Database Vendors



Amazon
Aurora



Amazon
DocumentDB



Amazon
Redshift



Amazon
Quantum
Ledger



Azure
Synapse
Analytics



MongoDB
Document Store



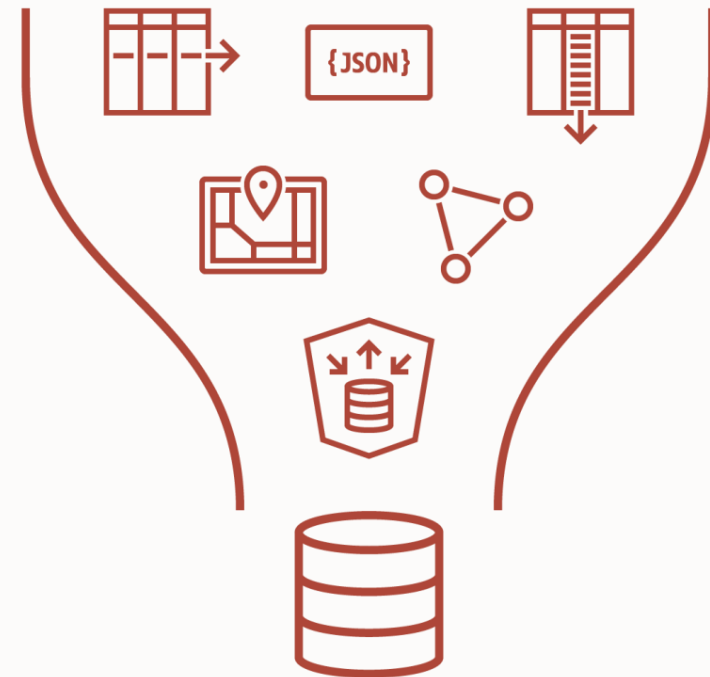
Snowflake
Data Warehouse



Google
Big Query

Run **single-purpose proprietary database** for each data type and workload

Oracle Database



Run **converged, open, SQL Database** for multiple data types and workloads

Oracle Database – Converged Features

- **Multitenant** for Efficient, Agile Database Clouds
- **In-Memory** for Database Acceleration
- **Sharding** for Hyperscale and Geo Distribution
- **Native JSON** for Document Data
- **In-Memory Ingest** for Fastest IoT
- **Cloud SQL** for integrating Object Store Data Lake
- **AutoML** for simple integrated Machine Learning
- **Persistent Memory Store** for Lowest Latency
- **Blockchain Tables** for Preventing Fraud
- **Spatial and Graph** for Mapping and Social Networks
- **APEX** for Low-Code Development
- And many more ...



Multitenant



In-Memory
Analytics



Hyperscale



JSON



In-Memory
IoT



Cloud
Integration



Blockchain



Persistent
Memory



Machine
Learning



Spatial



Graph

Multi-language Support

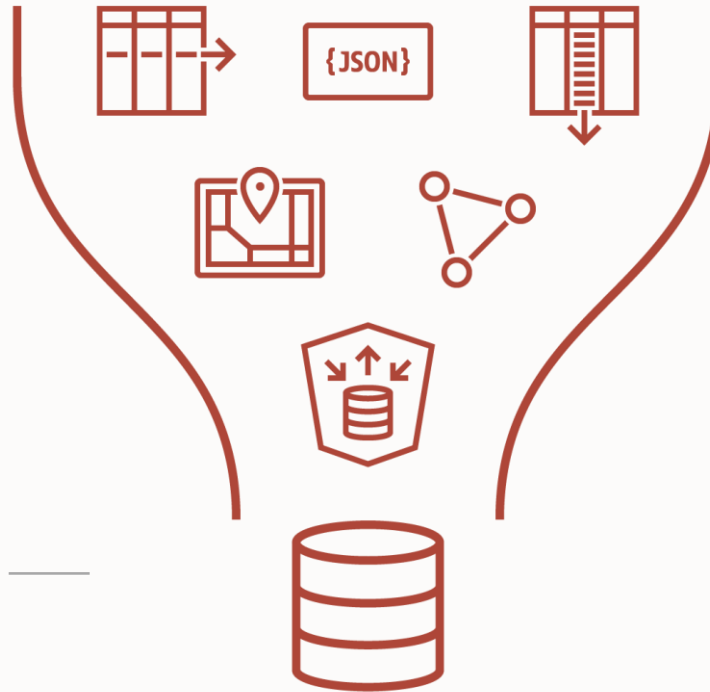
Oracle Database as a Data Platform

Development Services

Node.js, Python, .NET,
Java, PHP, Ruby, PL/SQL,
C, C++, Perl, Go, EBR,
REST Services,
Advanced Queuing,
APEX, SODA, Docker

Platform Services

Cloud to On-Premise, Clustering,
Microservices, Sharding, Security,
High Availability, Isolation,
Zero Data Loss, Administration



Analytical Services

SQL, Columnar In-Memory,
Machine Learning
(Advanced Analytics), R, AI













Data Support




Relational, JSON,
XML, Spatial, Graph,
RDF, Text, Binary.
Object Stores, HDFS,
Kafka, NoSQL Stores

Oracle Database for the Developer

Supporting all major development environments and APIs



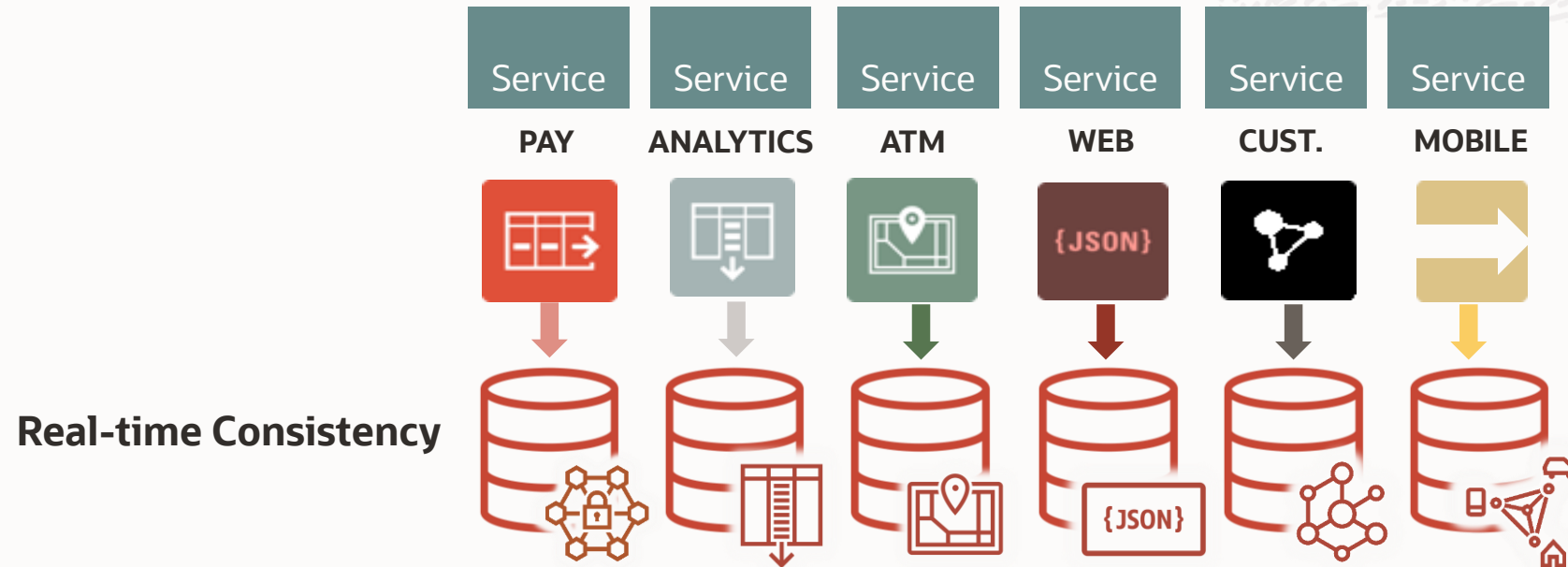
LANGUAGE		DRIVER
C		OCI, ODPI-C
C++		OCCI
Java		JDBC
.NET		ODP.NET
Node.js		node-oracledb
Python		python-oracledb
PHP		OCI8, PDO_OCI
R		ROracle
Go		goracle, rana, mattn
Rust		mimir
Ruby		ruby-oci8
Perl		DBD::Oracle

-  Oracle provided Drivers
-  Open-Source Drivers (Oracle contributions)
-  Open-Source Drivers (Third-party maintainers)



... and ODBC, OLE DB,
Pro*C, Pro*COBOL,
Pro*Fortran, SQLJ

Modernize with Converged Database



- Transforming data > Converged Database
- Availability > Maximum Availability Architecture
- Data consistency > Self-service data catalog
- Costly > Multi-Tenant
- Security > Database Vault
- Slow > Exadata

Strategy | Choose the Best Database Engine

Enterprise-class performance, scalability, reliability & security for all data-driven workloads

Any Data



Converged, open, SQL database for all data types and data models

Any Workload



Transparent scale-up, scale-out, sharding, in-memory, parallel SQL

Which one is Most Productive?

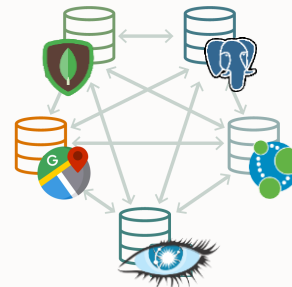


Specialized Data



Overhead from integration complexity

Dedicated Workload

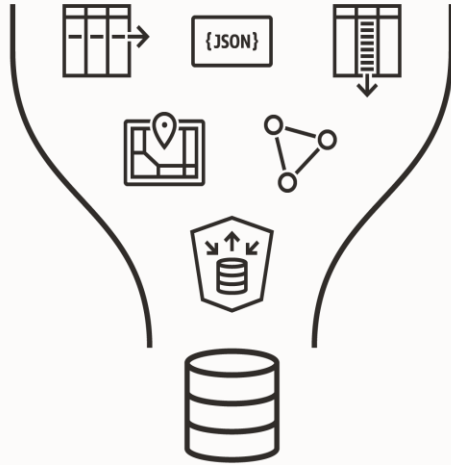


High effort exchanging and transforming different formats

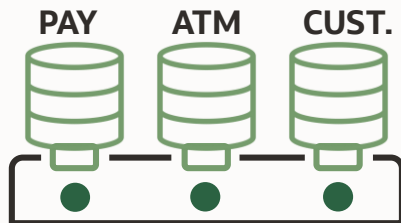
Developers & Architects, Data Analysts/Scientists and DBAs

Oracle Database

A Converged, Open SQL Database Allows You to Focus on Innovation



Converged Database



**Database Container
per Service**

Multi-model

Best-of-Breed Relational, JSON, Spatial, Graph, Cube, Text, Blockchain
Cross-model operations enables you to easily create value across all your data

Multi-workload

High Performance Transactions, DW, Analytics, ML, IoT, Streaming, Multitenant
Deep optimizations deliver exceptional price-performance across all workloads

Most productive for developers and analysts

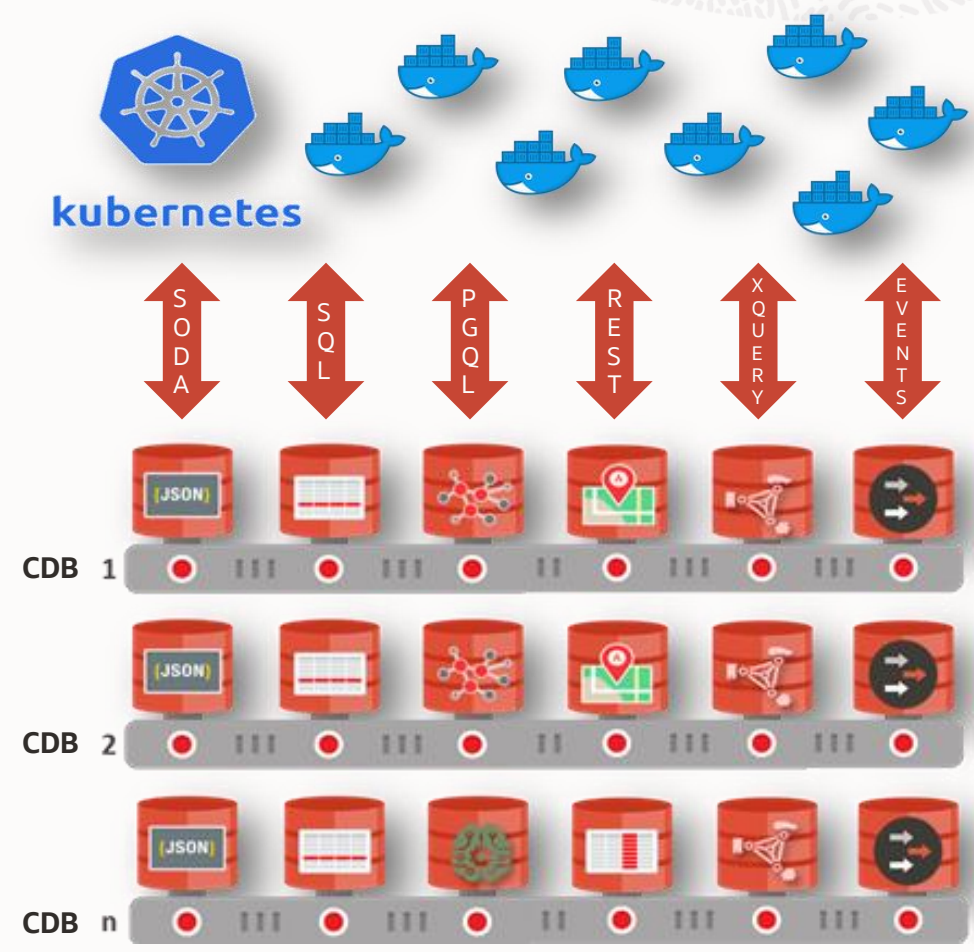
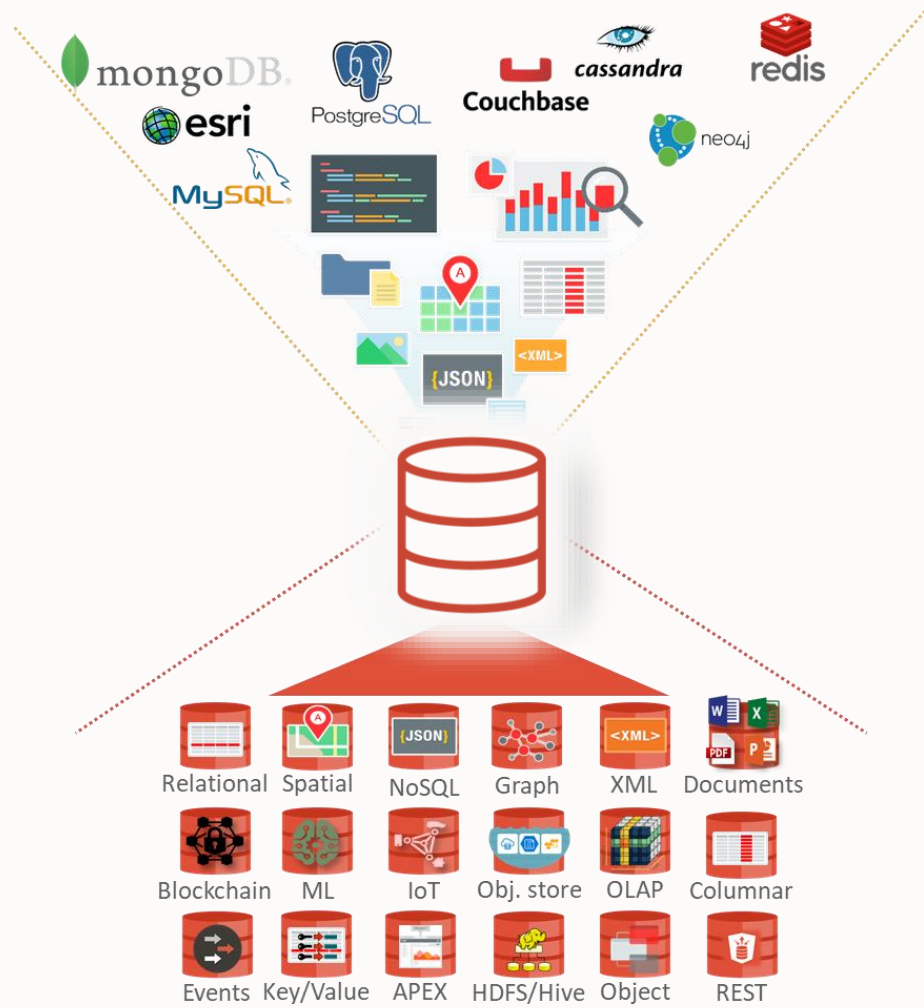
Same SQL and transactions operate on any data and workload
Integrated microservices, events, REST, CI/CD, Low-code

Modularize and containerize by app service

Low-level data types and workloads should not dictate your architecture

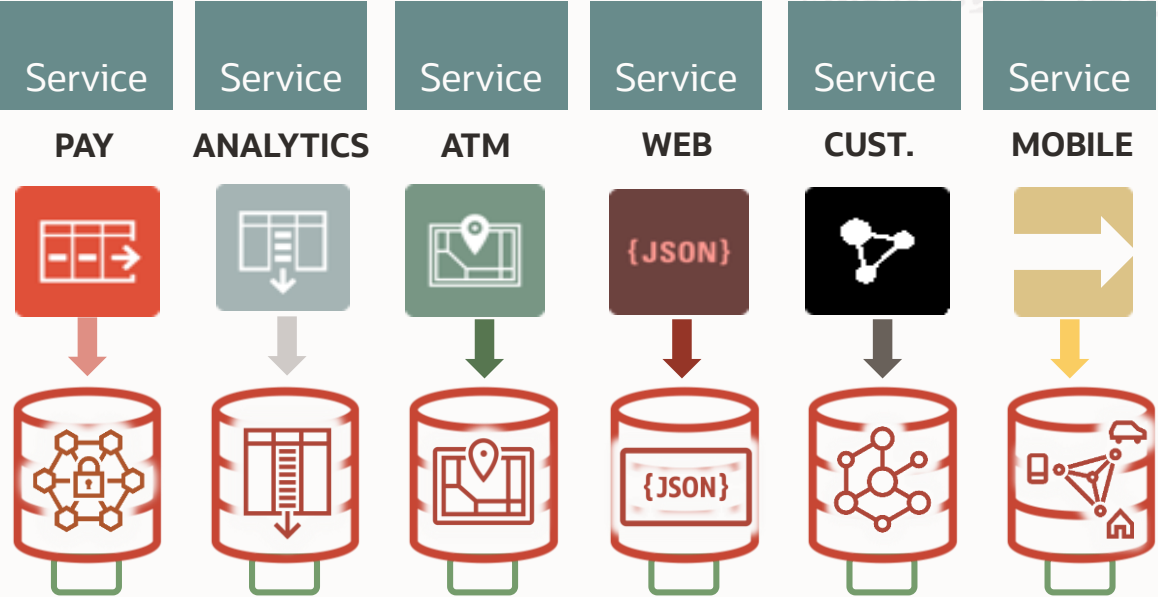
Oracle Database Developer Ecosystem

MODERN deployment model examples: Converged and Multitenant for Microservices

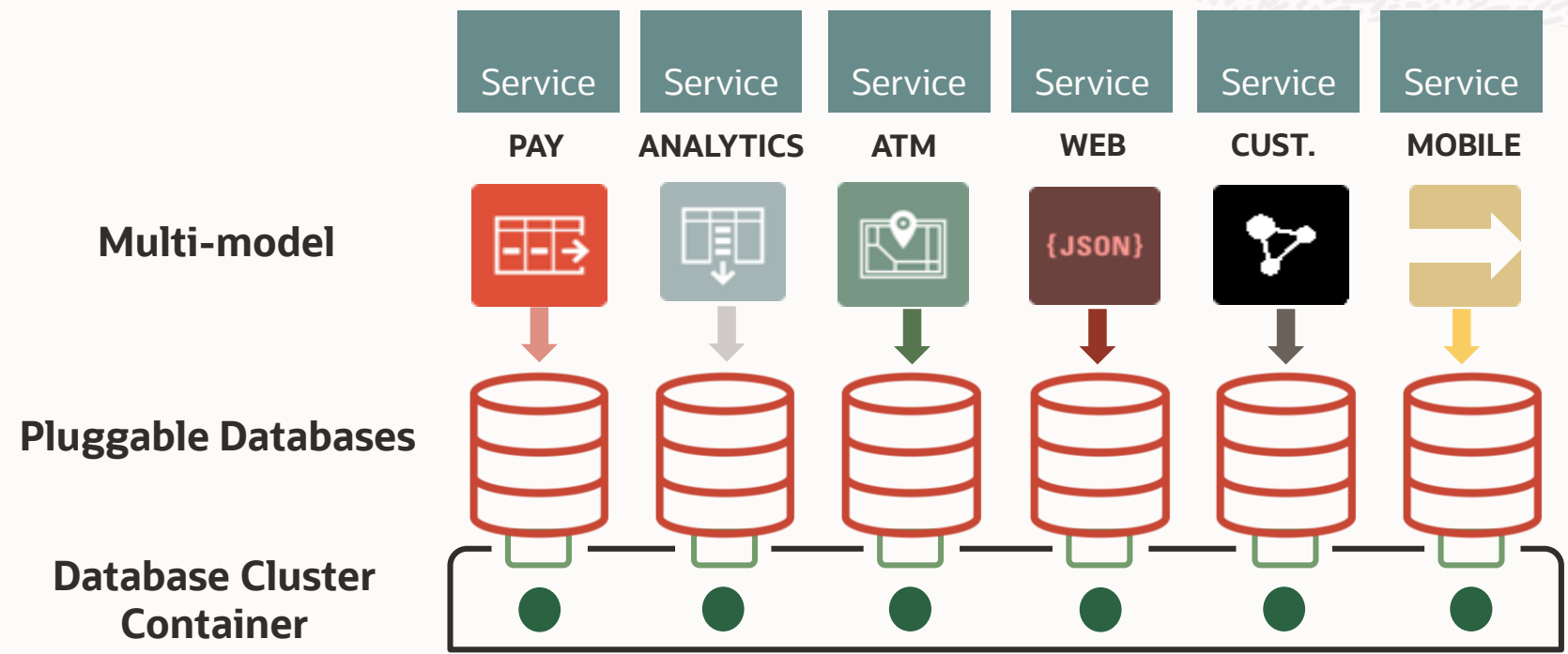


Modularize by Application Service

Data-driven optimization

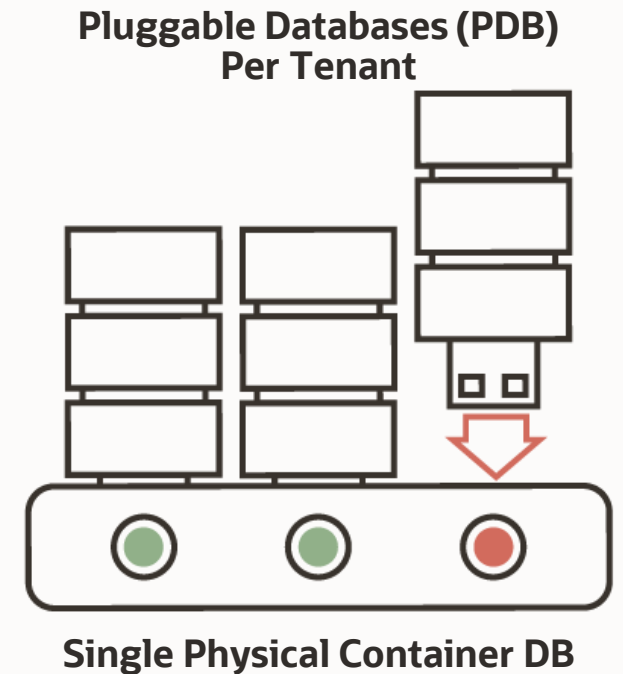


Containerize with Multitenant Architecture



Oracle Multitenant Simplifies SaaS Architectures

- **Oracle makes it simple** to implement secure and modular SaaS Apps
 - Use a separate logical database (PDB) for each SaaS tenant
 - Easier to develop – tenants are transparent to App
 - More agile - tenants are easily cloned, moved, etc.
- Database enforces tenant security
 - No need for risky coding of tenant separation in every App
- Enables tenants to use standard tools (e.g. analytics)
- Used by Autonomous Database, NetSuite, Fusion Apps, Taleo

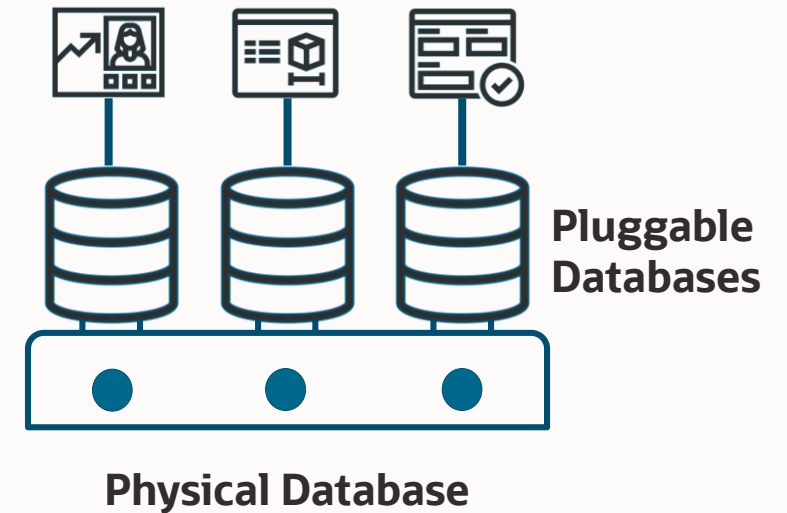


In-Database Containers

Containers simplify deployments by enabling rapid and light-weight deployments of logical systems instead of physical systems

Oracle implements the container concept natively in the database using light-weight **pluggable** container databases

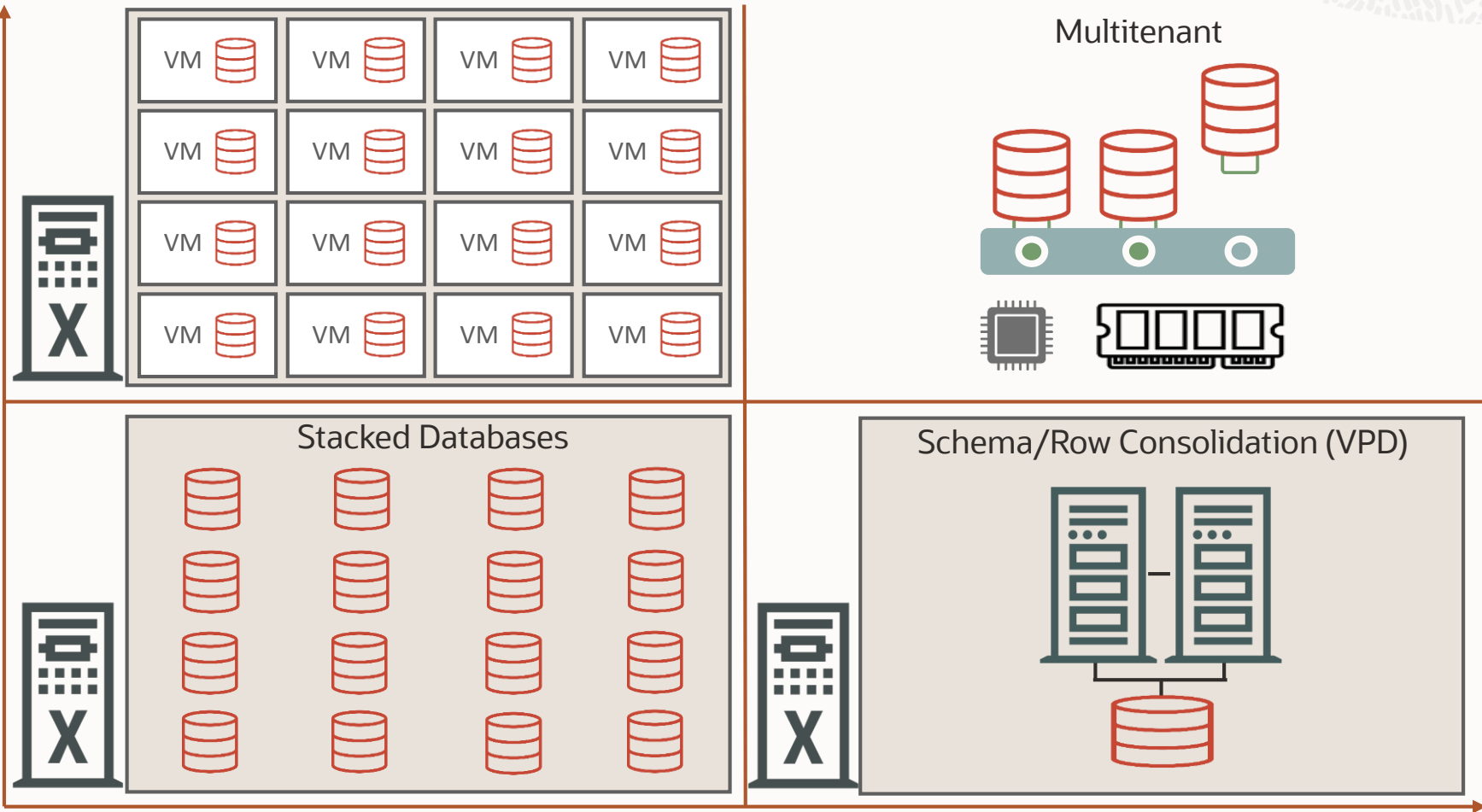
- Avoids the need for deploying a physical database for each Microservice
- **Many Pluggable Databases** can be deployed on top of a **single physical database** to simplify Operations
- Pluggable databases can be easily cloned and moved between physical databases



Up to 255 PDBs included with Database 23c Free

Multitenant vs. other Database Consolidation Architectures

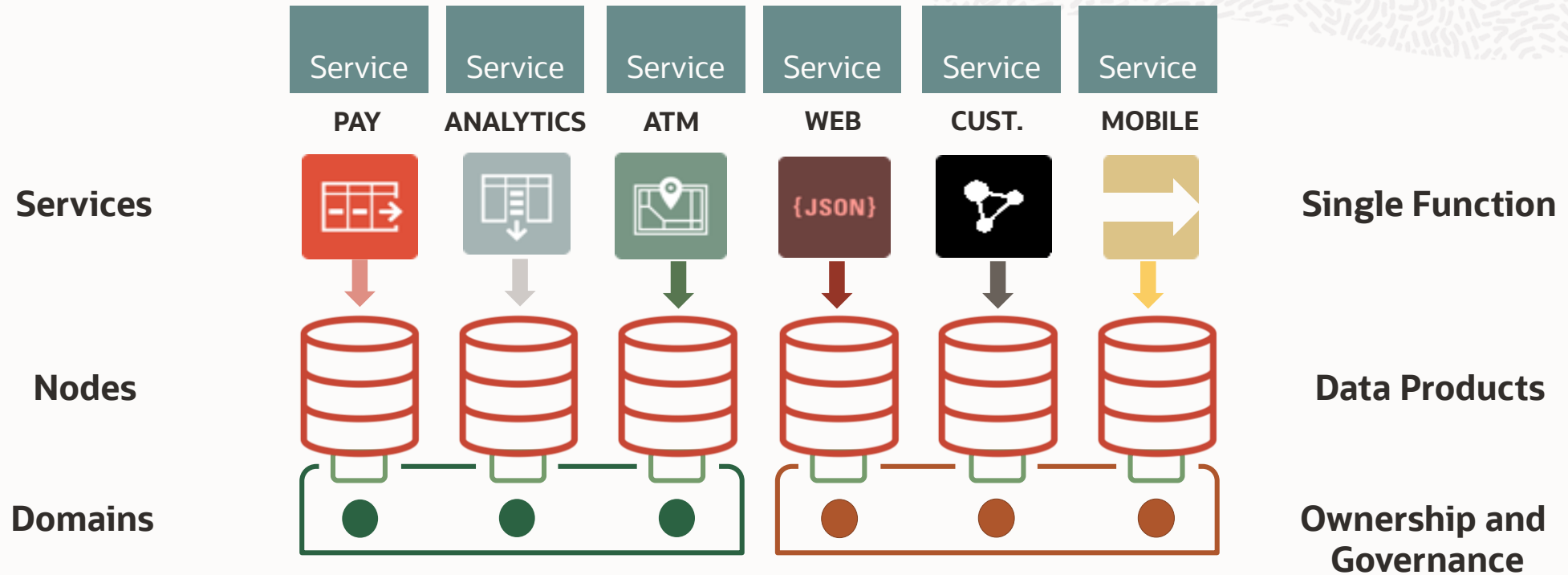
Isolation
& Agility



Efficiency



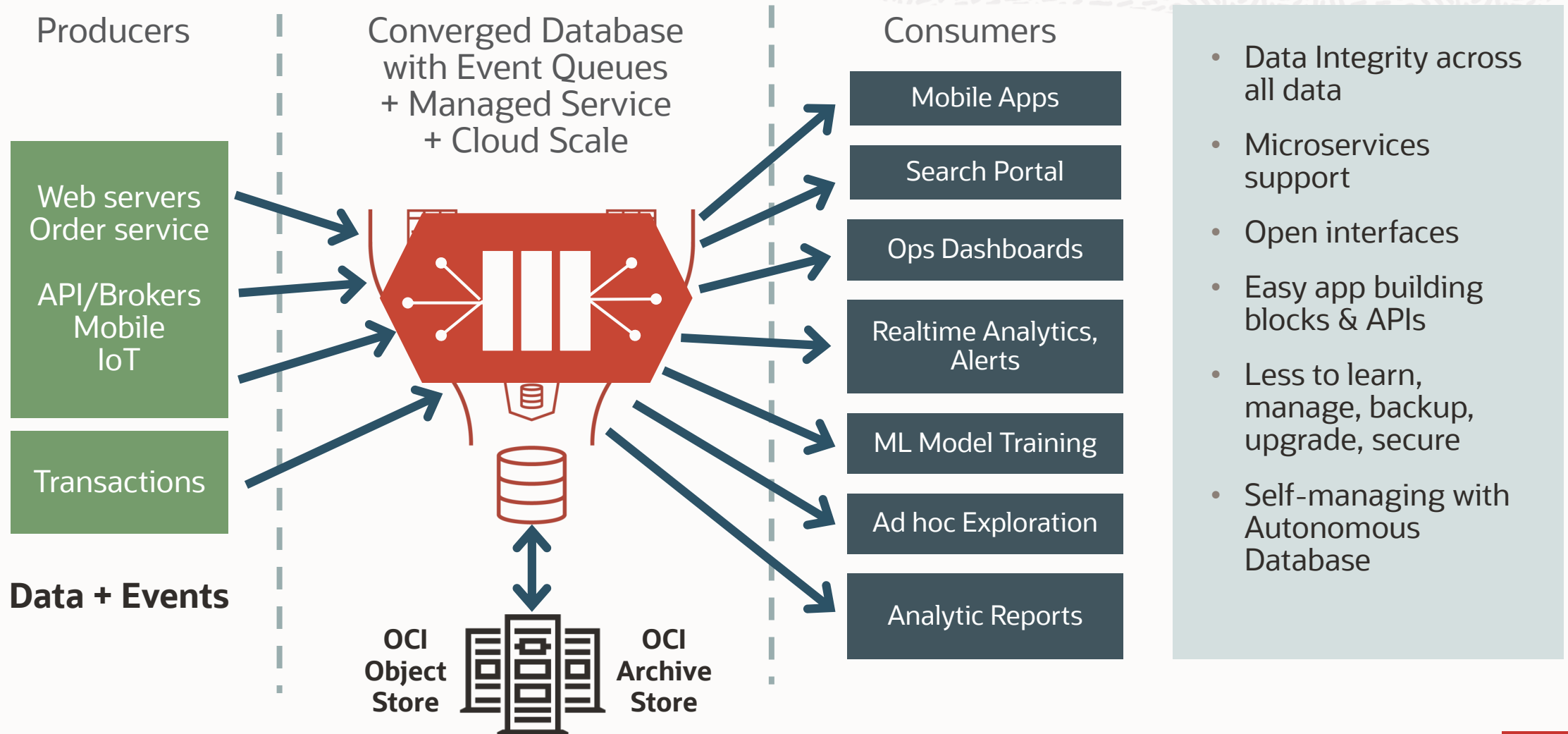
Decentralized Data Architecture



- Data-as-a-Product thinking
- Decentralized data architecture
- Domain-oriented data ownership

- Distributed data-in-motion
- Self-service access
- Strong data governance

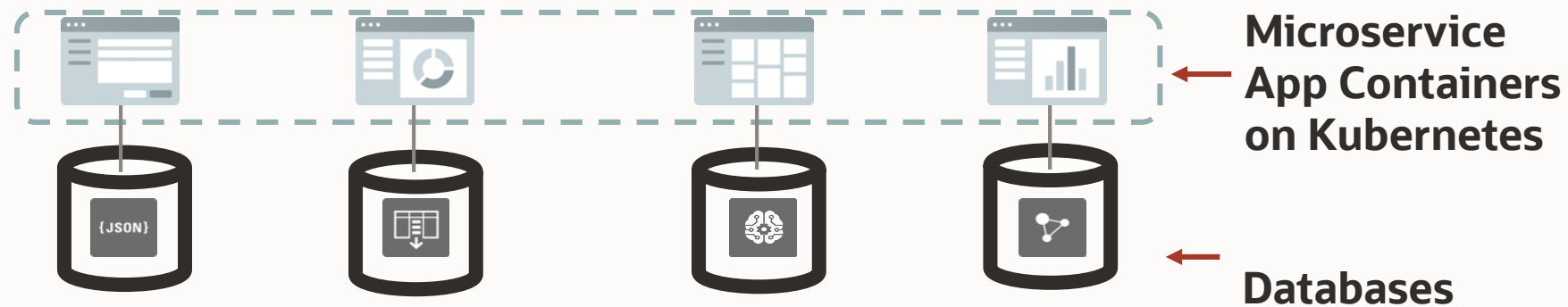
Database Convergence Simplicity



Microservice Architectures

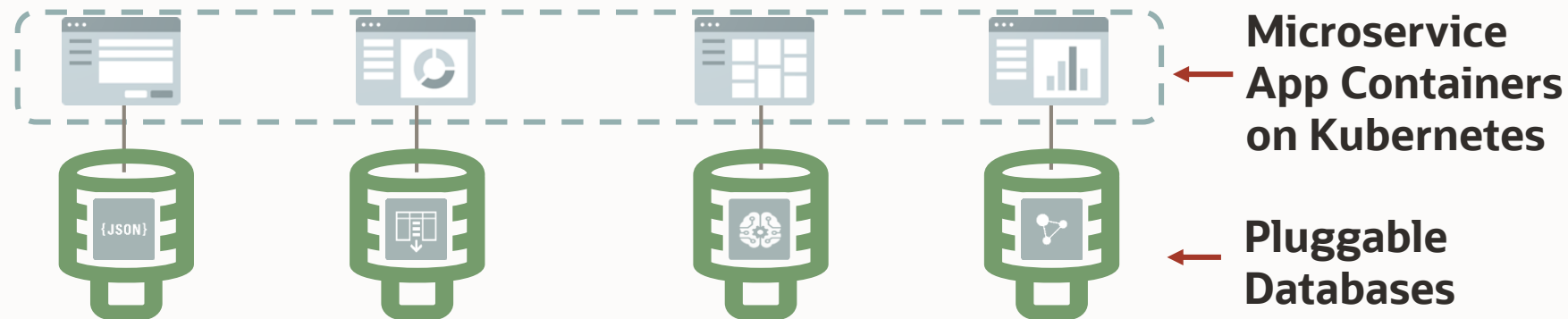
Microservices encapsulate functionality of specific application modules

- Agile: Can be developed and upgraded independently
- Highly available: Microservices can fail independently



Pluggable Databases Simplify Microservice Architectures

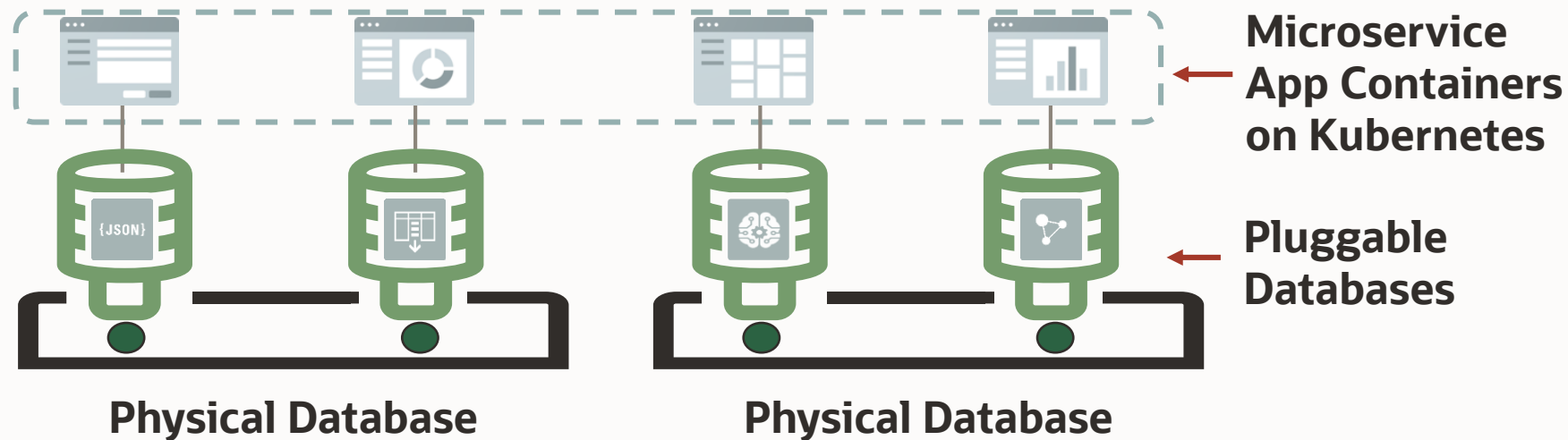
Oracle makes it simple for each Microservice to store its data in a logically separate Data Container called a Pluggable Database



Pluggable Databases Simplify Microservice Architectures

Oracle makes it simple for each Microservice to store its data in a logically separate Data Container called a Pluggable Database

Pluggable Databases can easily be physically **combined** to simplify deployment, or **separated** to improve isolation and scalability



Summary



1. Modernize the monolith basic concepts
2. Polyglot persistence – single-purpose data stores vs. Oracle converged
3. Oracle converged: multi-model, multi-workload, multi-tenant
4. Domain-oriented decentralized data architecture
5. Key features: Security, Blockchain, AI/ML, JSON, Graph, IoT
6. Pluggable databases for Microservice Architecture

Blogs and Workshops



Kubernetes operator

- Livelab: <https://bit.ly/3m6RwuW>
- Blog: <https://bit.ly/41jJK5>
- Web: <https://www.oracle.com/database/kubernetes-for-container-database>

Oracle Backend for Spring Boot and Mobile (Microservices)

- Livelab: <http://bit.ly/CloudBankOnOBaaS>
- Blog: <http://bit.ly/microservicesblogs>
- Web: <http://bit.ly/oraclespringboot>

MicroServices

- LiveLab: <https://apexapps.oracle.com/pls/apex/r/dbpm/livelabs/view-workshop?wid=3375&clear=RR,180&session=114876907409912>
- Event Code: 7731-JXHC-YTFI-DDIS

Thank you



Alexandre Fagundes
Cloud Architect, Oracle Latin America

ORACLE