ORACLE

Oracle GoldenGate: Best is yet to come

Dynamic Data Fabric and Trusted Data Mesh using the Oracle GoldenGate Platform

Vili Tajnić
Principal Solutions Engineer
OD Prime Tech EE
June 2022

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.

Introduction

Executive summary

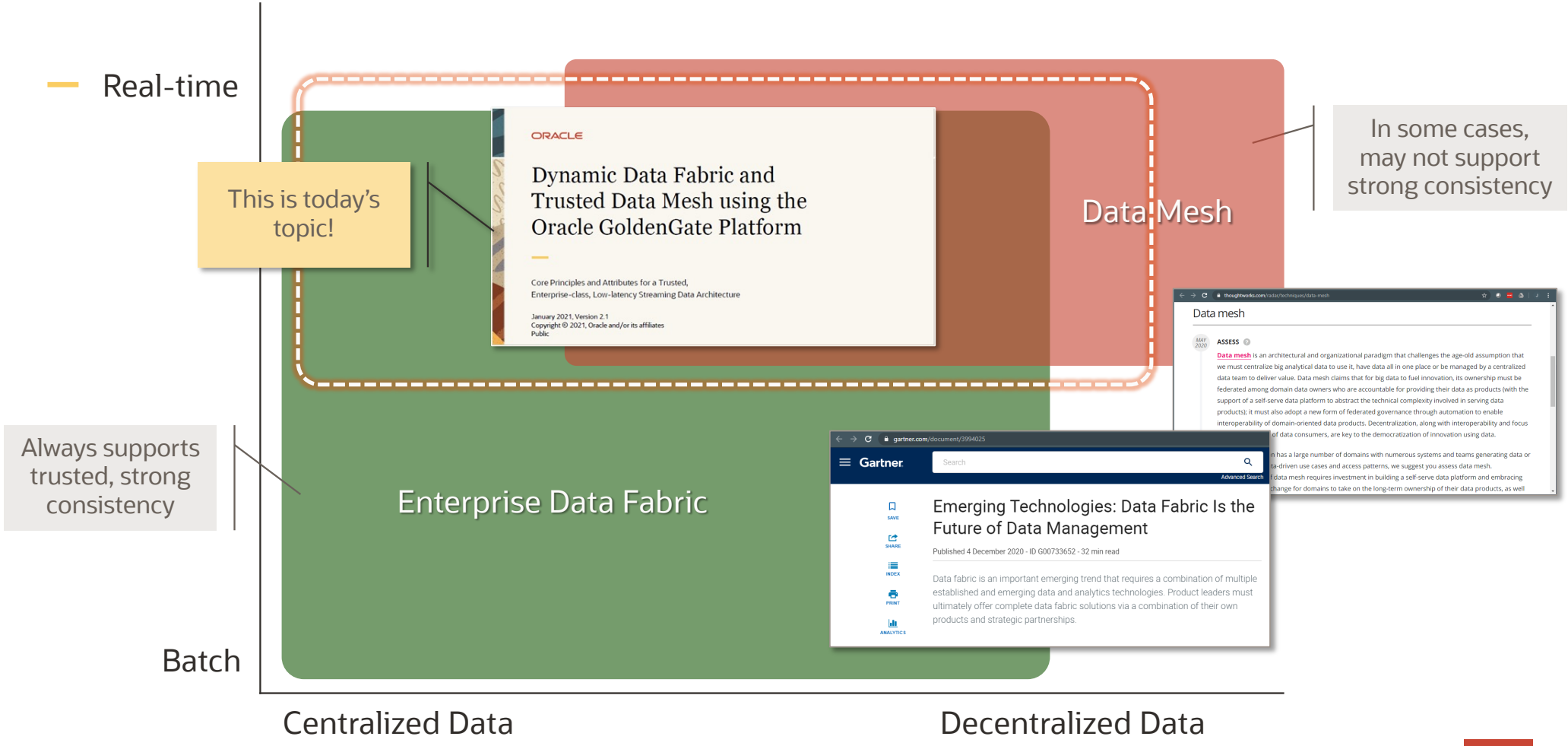
How to move forward?

Market winners will be those enterprises that succeed as driving more value from their data.

Dynamic Data Fabric and **Trusted Data Mesh** is the new kind of data architecture that is prioritizing ”Data Product Thinking”

- Total clarity into data’s value chain
- >99.999% operational data availability
- 10x faster innovation cycles
- ~70% reduction in data engineering

Data Fabric & Data Mesh



A Dynamic Data Fabric for Enterprise Needs

Data Fabric is a design concept that spans many 'styles' of data integration and governance to achieve a harmonized and cohesive solution.

Oracle is recognized as as a leader and the portfolio includes:

CLOUD-NATIVE, COMMON PLATFORM DATA FABRIC	BEST-OF-BREED DATA FABRIC FOR MULTI-CLOUD & ON-PREMISE
• Self-Service ETL for Analytics & Autonomous DB • OCI Data Catalog, OCI Data Integration, OCI Data Flow • OCI GoldenGate and Stream Analytics for OCI • Integration Cloud and Oracle Cloud SQL	• Oracle Data Integrator (w/ETL, Quality, Messaging) • Oracle GoldenGate and Stream Analytics • Oracle Big Data SQL (Data Federation) • Oracle Data Visualization (Data Preparation)

Oracle Concept of a Trusted Data Mesh

Trusted Data Mesh is:

- a data architecture approach focused on outcomes (data products),
- IT agility in a multi-cloud world (mesh),
- trusted data of all kinds (polyglot data streams),
- and faster business innovation cycles (using event-driven data ledgers).

GoldenGate as a Bridge to Fabric / Mesh

Data Replication

Data High Availability
- Oracle/non-Oracle DB
- Low downtime migrations

Transaction Replication
- OLTP/reference data

Data Warehouse Loading
- Non-invasive Capture
- Realtime Staging

Data Pipelines

Data Lake Ingest
- High Fidelity Change Stream
- Event-driven Realtime

Pre-Processing
- Filter, Correlate, Enrich

Data Transformations
- Streaming ETL Ops: Query, Aggregate, Lookup, etc.

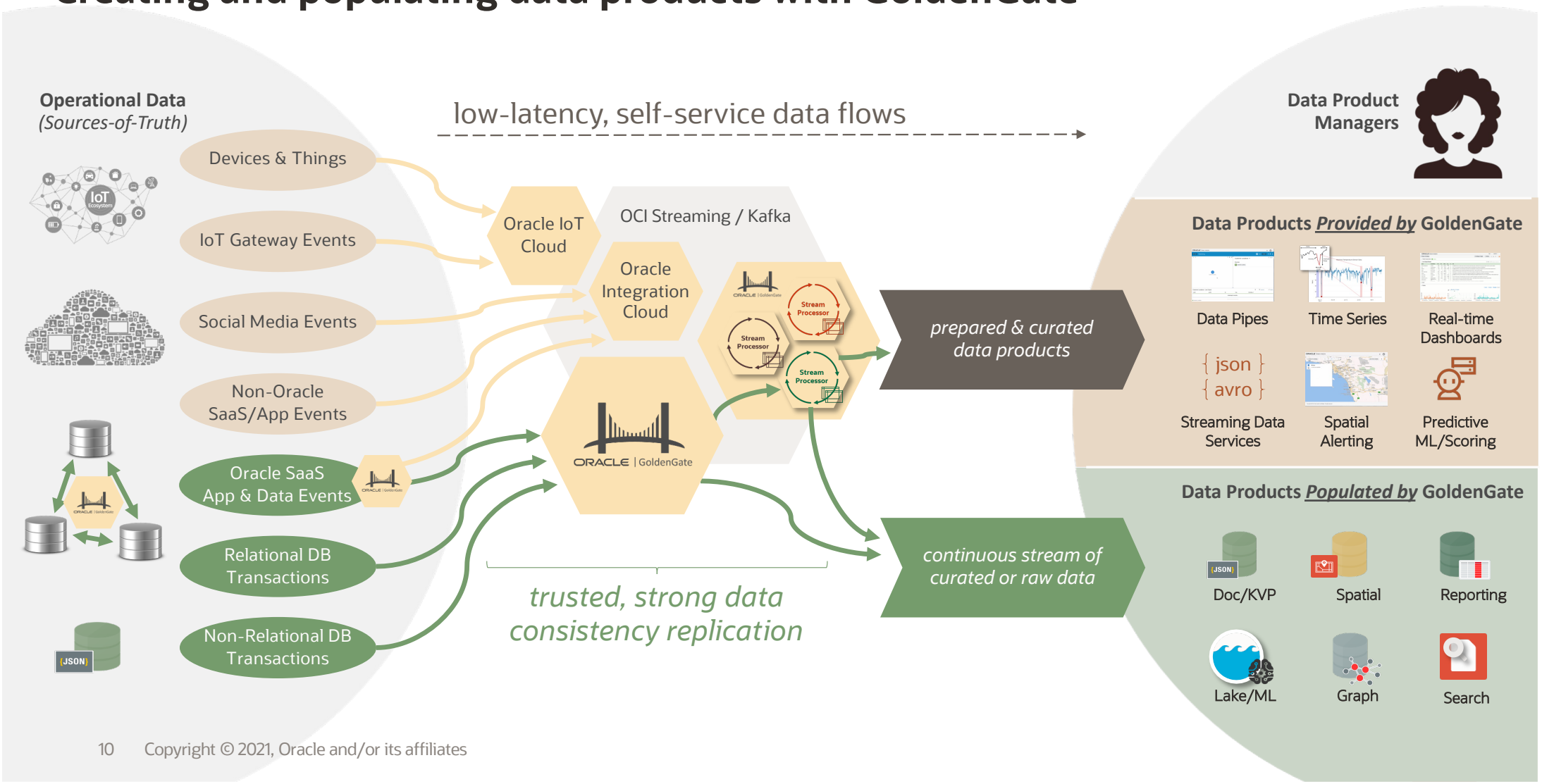
Stream Analytics

Analyze Data Streams
- Time Series Analysis, Machine Learning, Geo-Spatial

Patterns
- Pre-built Accelerators

Delivery & Alerting
- Output Data to Data Stores
- Sophisticated Alerting

Creating and populating data products with GoldenGate



A time for change

Outmoded thinking and cultural changes that must change. Evaluate next-gen tech innovations that are the springboard for this new data architecture approach.

Attributes of classical data management monoliths

TREAT DATA AS AN IT ARTIFACT	*Data is treated as a byproduct of the application functions, requiring the semantics of domain modeling to be done and re-done many times in the IT lifecycle*
MONOLITHIC AND CENTRALIZED	*Hub-and-spoke style architecture dominates the IT ecosystem, but each App, ETL, Mart, Warehouse, Lake etc presumes itself to be the center of the data landscape, thereby requiring IT to constantly fund projects that integrate and align data using incompatible tools*
WATERFALL DATAOPS / DEVOPS	*In the "data landscape" of databases, ETL, Marts, Warehouse, Lakes etc etc there remains a strong bias towards waterfall-style operations, the Agile methodologies of development haven't been able to provide a repeatable CI/CD lifecycle approach across the monoliths*
BATCH PROCESSING CENTRIC	*Movement of data for most Analytic (OLAP) domains remains predominantly batch-oriented… to be driven by the scheduler (clock) rather than the events of the business itself*
OLTP VS. OLAP (DECOUPLED)	*Different operational applications (OLTP) and analytics (OLAP) are often separated organizationally, politically, and technically – causing IT friction (delays), data domain semantic issues (reduced data trust), and least-common-denominator solutions (low innovation)*

Model Crisis: The Data Monolith

Classic monoliths of data management are not optimal for multi-cloud ecosystems where data must be:
(a) cohesively managed across infrastructure separated by 100's/1000's of kilometers,
(b) use different data formats (polyglot) and
(c) stay connected at very low latencies.

Data monoliths examples:

- Classical ETL and Data Preparation Tools
- Classical Data Replication Tools
- Operational Data Stores (ODS)
- Enterprise Data Warehouse (EDW)
- Data Lakes (reservoirs, ponds, swamps, lake houses etc)

Old Ways of Thinking and Assumptions to Abolish

1. Treating Data as a By-product / IT Artifact
2. OLTP -> ETL -> OLAP Fragmentation
3. Waterfall Process for Data Projects
4. Batch-first Design Thinking
5. Hub-and-Spoke Architecture Bias
6. Storage-centric Modelling

Why Now? Enabling Technologies

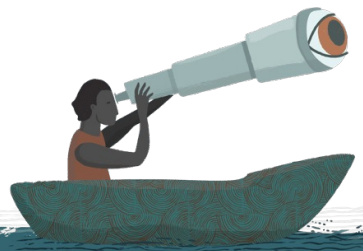
1. Service Mesh, Kubernetes etc
2. Mature Software Defined Networking
3. Microservices for CDC Replication
4. Polyglot and ACID Payloads in DB Replication
5. ETL Can Run in a Stream
6. Free MPP DAGs and Serverless

Key Objectives and Tangible Benefits

Overall Benefits

Reduce Costs for Mission-Critical Data Operations

Faster, Data-Driven Innovation Cycles



Operational Data Outcomes

Multi-Cloud Data Liquidity
- Data capital is fast data

Real-time Data Sharing
- SaaS / Apps / OLTP DBs

Edge, Location-based Data Services
- Correlate IRL device events

Modernize Microservices & Data
- Trusted "event sourcing"
- DataOps and CI/CD for Data

Uninterrupted Continuity
- >99.999% up-time SLAs
- cloud migrations

Analytic Data Outcomes

Eliminate SQL-based Copies
- Event-driven pipelines

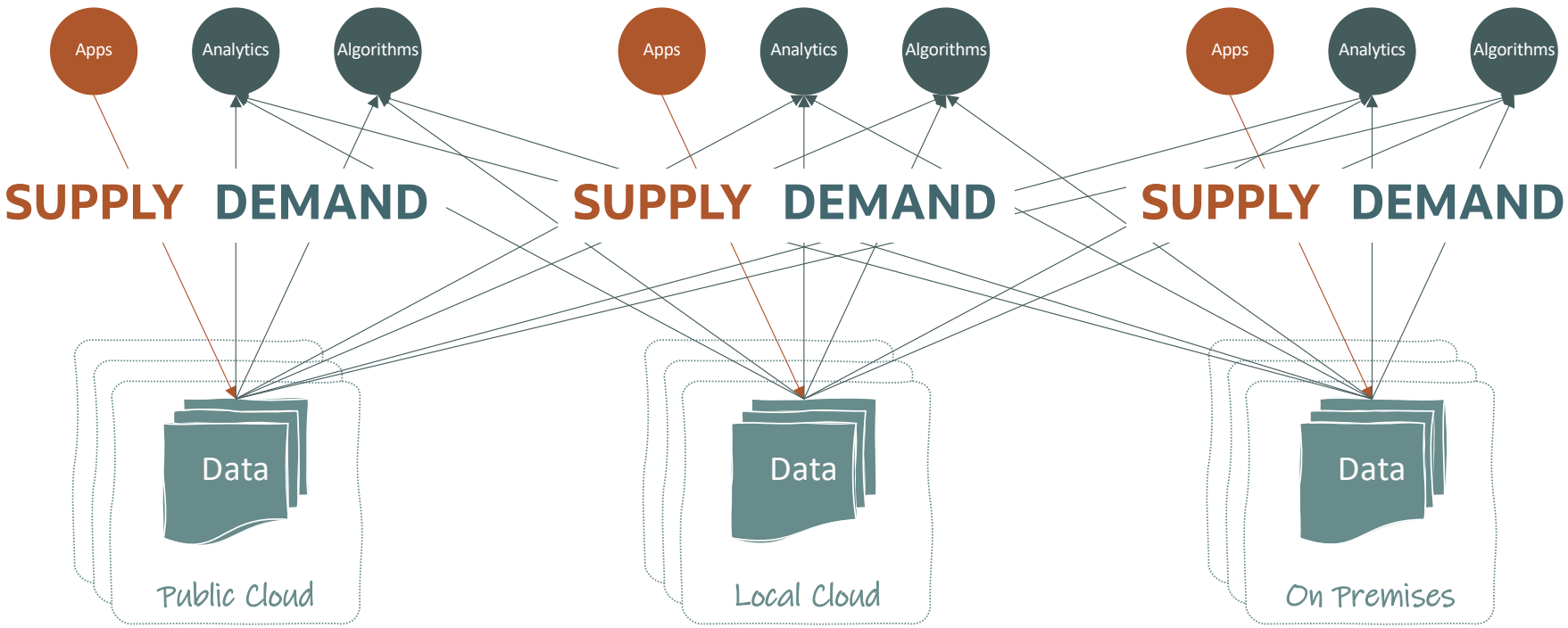
Time Series Data
- Deltas / changed records
- Event-by-event for better data science / ML

Distributed Data Lakes
- Hybrid / multi-cloud

Real-time Stream Analytics
- On events/time-series data

Predictive Analytics
- Data monetization, new 'data services' for sale

Data Liquidity



New Paradigm for Data Architecture

Upon the backdrop of these behavioural and technology changes, a new approach becomes possible: an Enterprise Data Fabric that is dynamic and real-time, and a Data Mesh that works well with trusted enterprise data.

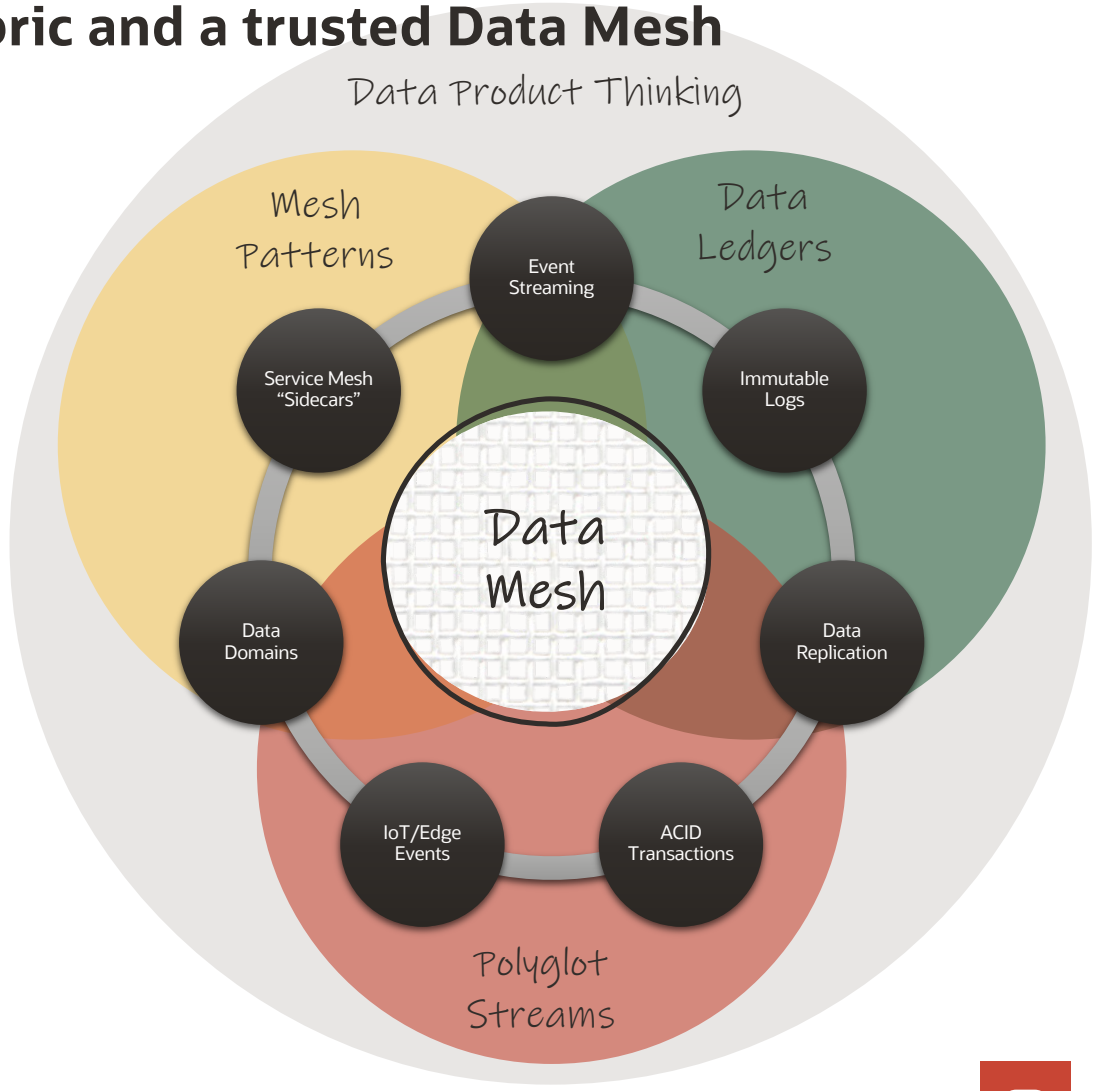
Key principles of a dynamic Data Fabric and a trusted Data Mesh

Principle #1: Decentralized, Modular Mesh

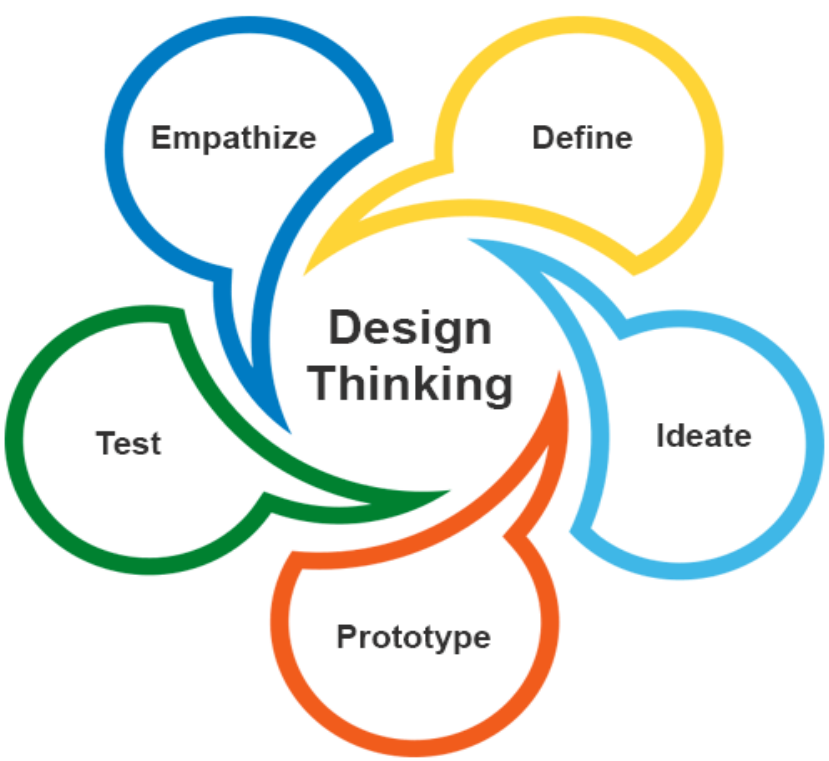
Principle #2: Enterprise Data Ledgers

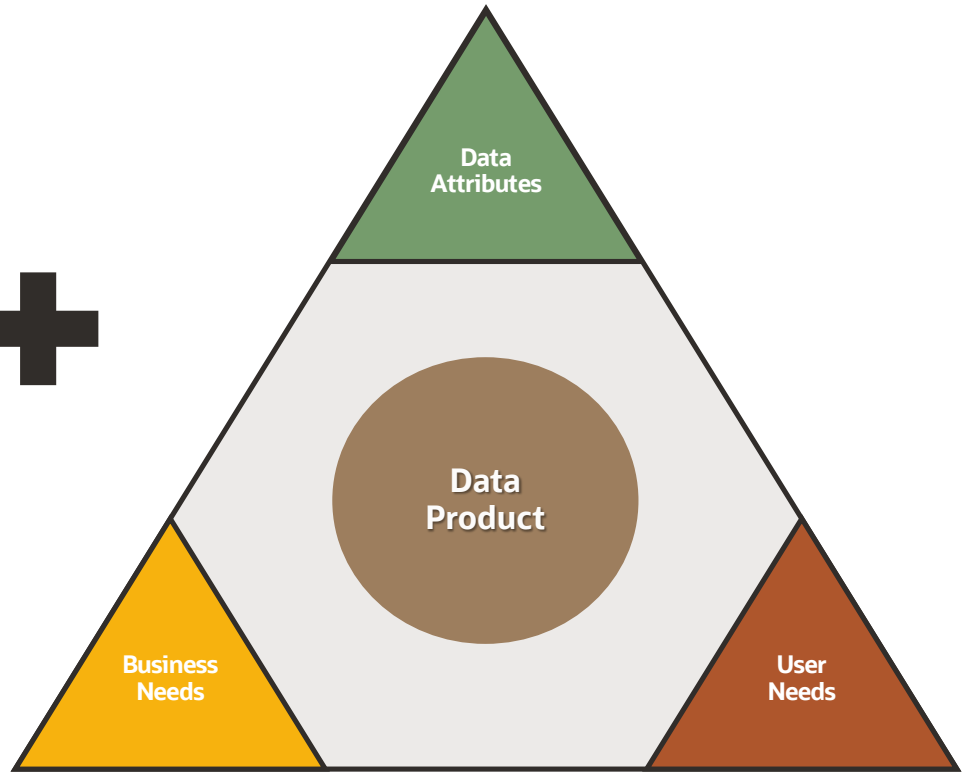
Principle #3: Trusted, Polyglot Data Streams

Foundation: Data Product Thinking



Foundation: Data Product Thinking





Principle #1: Decentralized, Modular Mesh

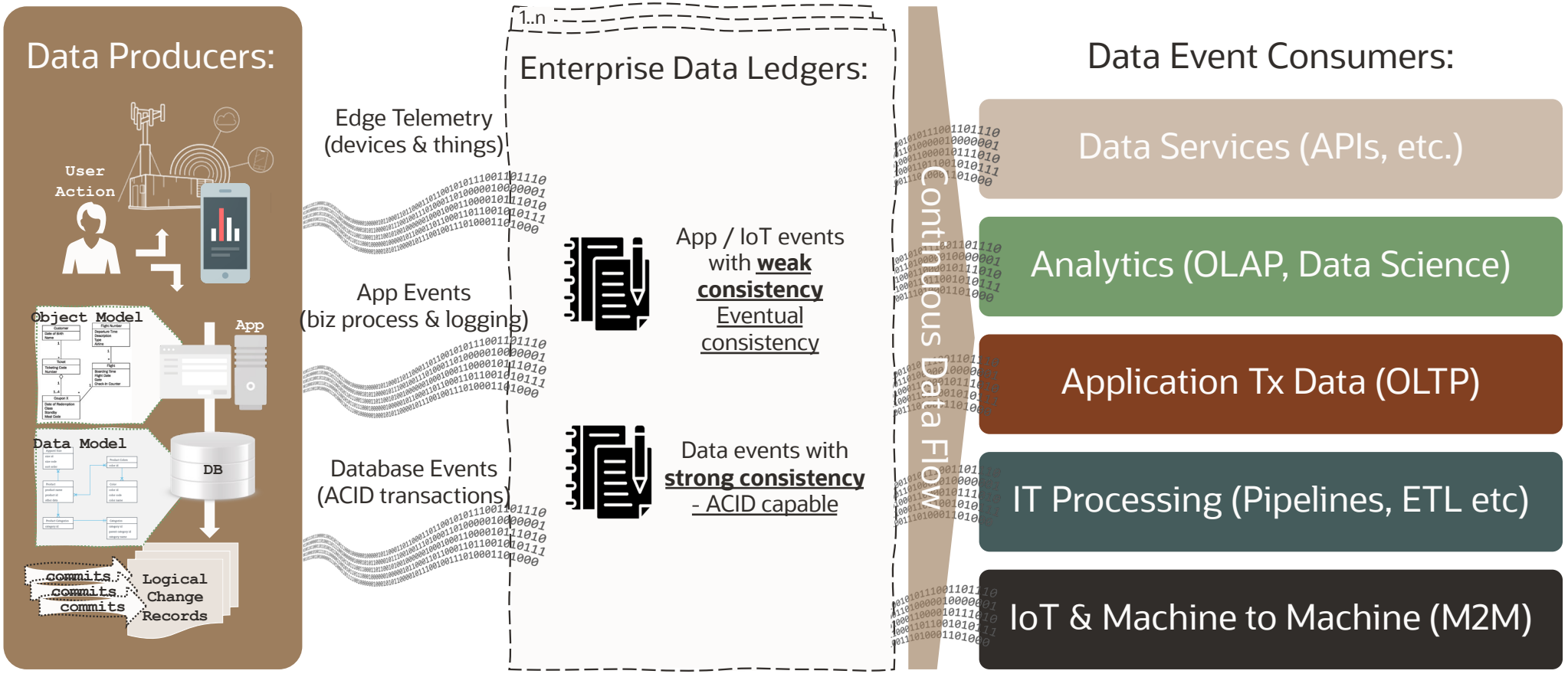
More “Hub and Spoke”?

...or, are you ready to Mesh?

Principle #2: Enterprise Data Ledgers

Event Ledger Type	Common Use Cases
Application Architectures	Rise in popularity for microservices-based ledger solutions utilizing the event sourcing pattern, which depends on an '**event store**' for cross-component communications
Infrastructure Log Ingestion	**Event ledger for collecting telemetry** (operational behaviour) of enterprise infrastructure, tools like Apache Kafka, Pulsar, and AWS Kinesis etc are common examples
Enterprise-wide Data Services	Apart from simple log ingestion, the same ledgers such as Kafka and Pulsar are also being used for **event-based data service**s more broadly in enterprise use cases
Multi-party Blockchain Ledger	Adopted for its transparency and immutability, **blockchain technologies** are being used for cross-organization (eg; B2B) exchanges to track transactions in a highly trusted way
Distributed Database Transactions	globally distributed database transactions (eg; for 'sharded data' or for high availability) will **replicate event**s using Data Replication tools like **Oracle GoldenGate**

Online Event Processing (OLEP)



Principle #3: Trusted, Polyglot Data Streams

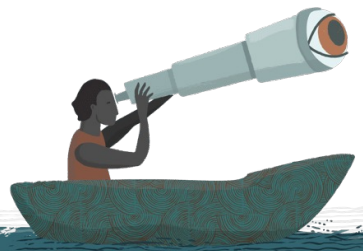
DATA TRUST CHARACTERISTICS	
Exact-once message processing	In a distributed system (messaging or databases), the computers that make up the system can fail independently of each other. Exact-once processing guarantees that if a message producer tries sending a message many times, it leads to the message being delivered exactly once to the end consumer.
Strict ordering of events	Application producers often persist data in 3rd normal form data structures. When data is saved, the precise ordering of the transactions are extremely important to the consistency of the data. A trivial example is ItemHeader and ItemDetail tables, with a Foreign-Key property between them the sequence of an update can result in consistency/commit failure if performed out of sequence.
Transaction atomicity (A)	As with the ordering of events, data transactions are often grouped together for a common "commit point". This allows for the database system to guarantee the atomicity of the transaction as a whole, even though the commits may consist of many different SQL statements. As atomic transactions are converted to other payloads and placed in messaging systems (aka: the enterprise event ledger) the systems must be capable of preserving the holistic atomicity of the transactions as they move through distributed systems.
Transaction consistency (C)	For purposes of this document our focus is on the consistency of the transaction itself when the data requires referential integrity. A trusted Data Mesh must be capable of preserving the consistency guarantees of relational payloads as they flow through the mesh. Row level and system level consistency of the transactions (possibly as large groups of SQL) should be preserved and guaranteed.
Isolation (I) and durability (D)	Concurrent transactions are guaranteed to be processed as if there is sequential isolation, in order to preserve the consistent state of the system even while incomplete transactions are in-flight. For maximum durability a Fabric or Mesh should provide near-zero RPO (recovery point objective). Durability is essentially a measure of acceptable data loss.
Rollbacks and recovery	Should any of the above trust types be violated, or if there is some sort of unplanned disaster within a partition or consumer, then the Fabric/Mesh frameworks should be capable of restoring the data to a consistent state.

Dynamic Data Fabric, Deployed as a Trusted Mesh

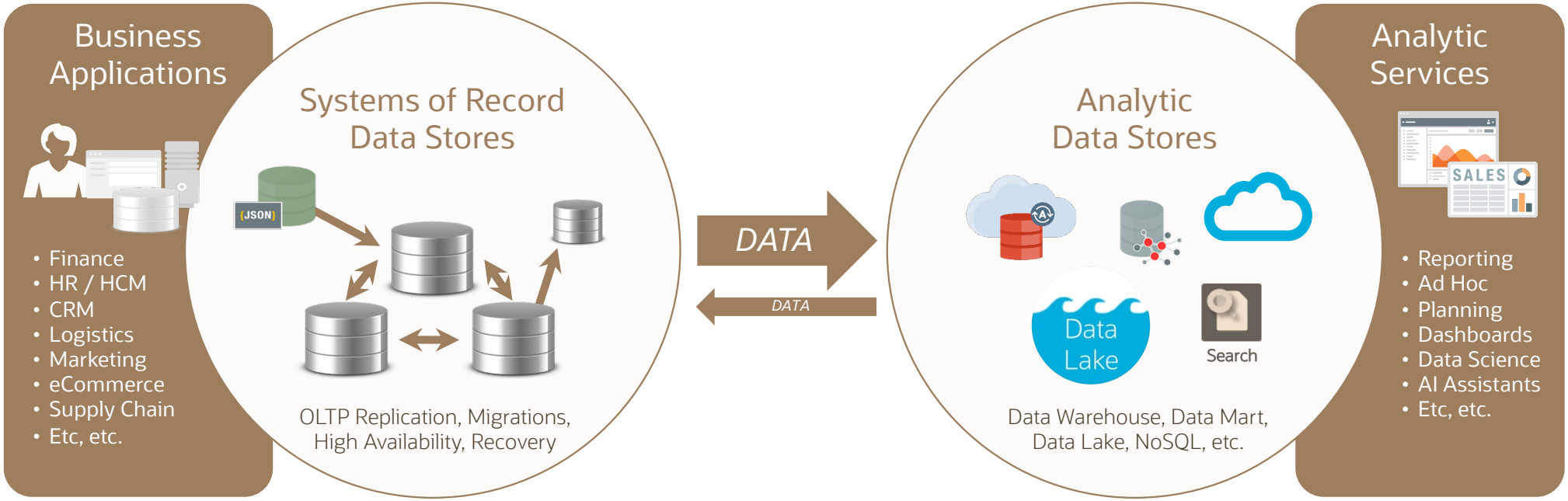
We focus on Data Fabric – a dynamic, real-time and streaming centric Data Fabric

Benefits and opportunities of applying a dynamic Data Fabric and a trusted Data Mesh

Data Monolith to …	Dynamic Fabric / Trusted Mesh
Data as an IT artifact	Data as a Product
Monolithic & Centralized	Distributed & Decentralized
Waterfall Data/DevOps (dominant)	Agile, CI/CD Data/DevOps
Batch Processing Centric	Event-Driven Streaming by Default
OLTP vs. OLAP	OLTP ∩ OLAP

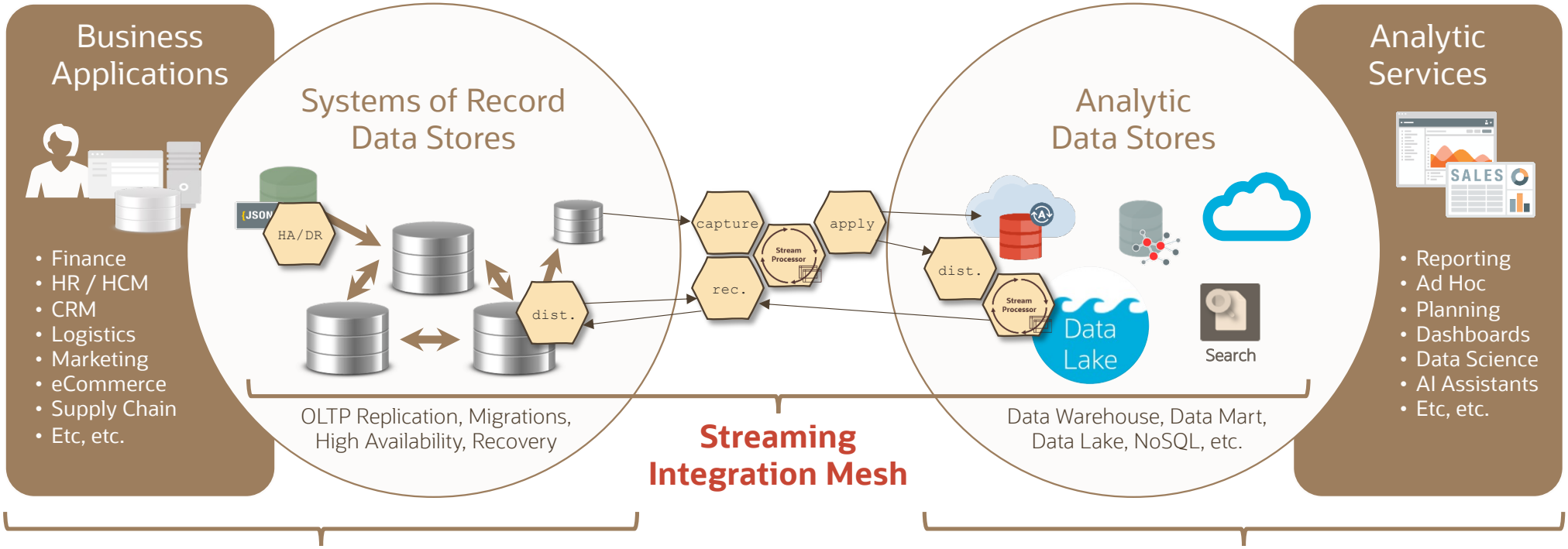


Aligning Operational and Analytic Data Stores



Traditional separation between OLTP and OLAP data domains

Real-time streaming integration



Strong Data Consistency
these data stores "run the business," track the money and most important transactions

Data Consistency Depends…
different analytic data stores will have differing ACID/consistency support

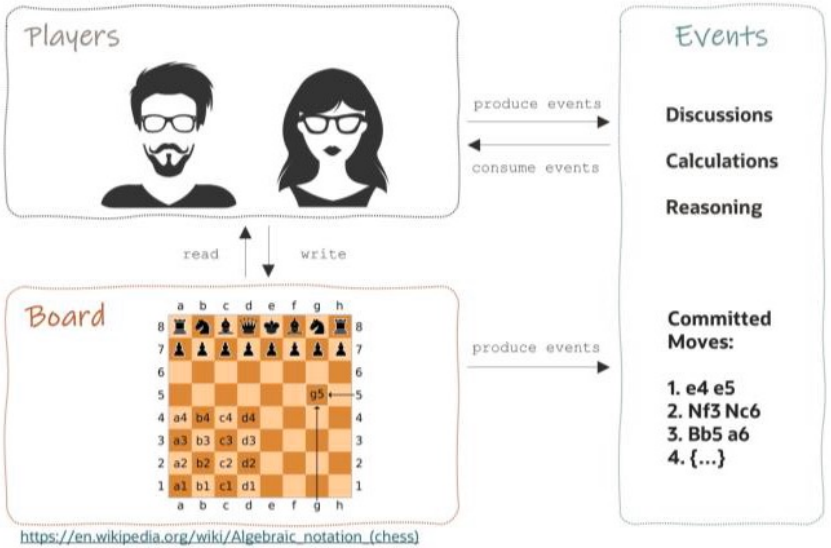
Enterprise event ledger as a narrative of the truth for enterprise systems

“Source of Truth”

- Purpose, intent and strategy
- Game is first played in the mind of the player

“State of the Truth”

- Physical position at the point in time
- Durable re-start point, in case match is paused and restarted later



https://en.wikipedia.org/wiki/Algebraic_notation_(chess)

“Narrative of the Truth”

Sequence of events

(between the snapshots)

- Record of the committed moves, from start of the match to the end
- Strategies may change as new calculations are made (events that occur within the minds of the player)

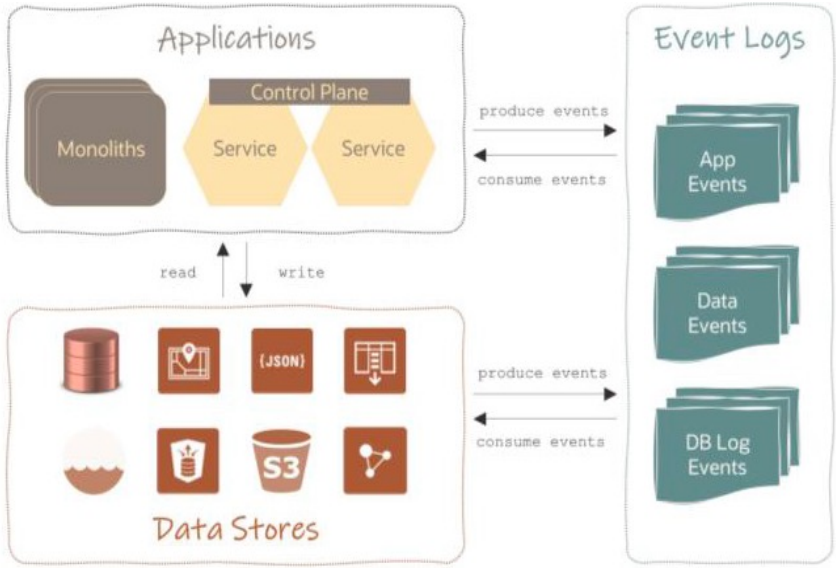
Enterprise Event Ledger - Chess Metaphor

"Source of Truth" at app tier

- APIs, business rules and business object semantics
- Data typically serialized in-memory

"State of the Truth" at the point in time (current or historic)

- Highly durable storage used for Data Recovery / Archives, years of data
- Polyglot persistence, each store may have different formats



"Narrative of the Truth"

Sequence of events

(between the snapshots)

- Days/months+ of event data available as Time Series or Messaging
- Strict ordering of events & Idempotency
- Strong Consistency of DB transaction logs (eg; when using tools like Oracle GoldenGate)

What Ledger Technology?

- **Microservices Event Stores** – that include native features that align to common microservices patterns such as CQRS
- **Time Series Databases** – which are optimized for high-volume writes, typically from IoT devices
- **Message Queues or Service Bus** – proven queues for business process transactions and structured data payloads
- **Event Streaming Platforms** – such as Apache Kafka, Pulsar and various proprietary cloud messaging services
- **Data Replication Tools** – which are natively integrated with database event logs for maximum data consistency
- **Blockchain** – especially useful for multi-party event ledgers, where immutability and transparency are mandatory

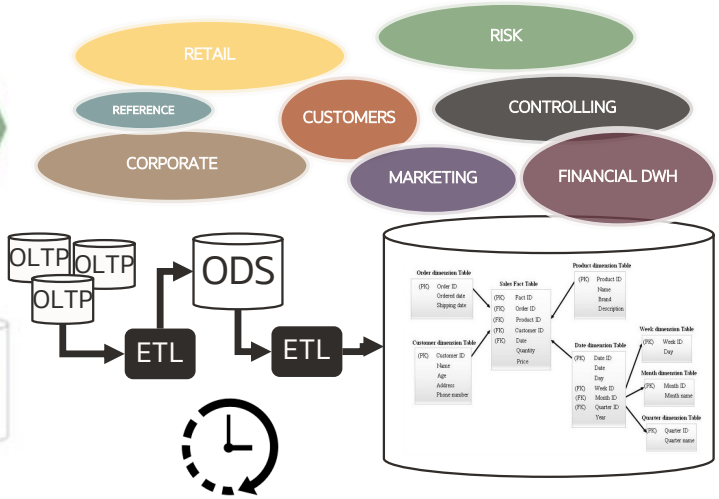
Decomposition of the Data Monolith

Functional Decomposition in Microservices Apps

(grouping by application domains)

Data Decomposition from Classical OLTP to EDW

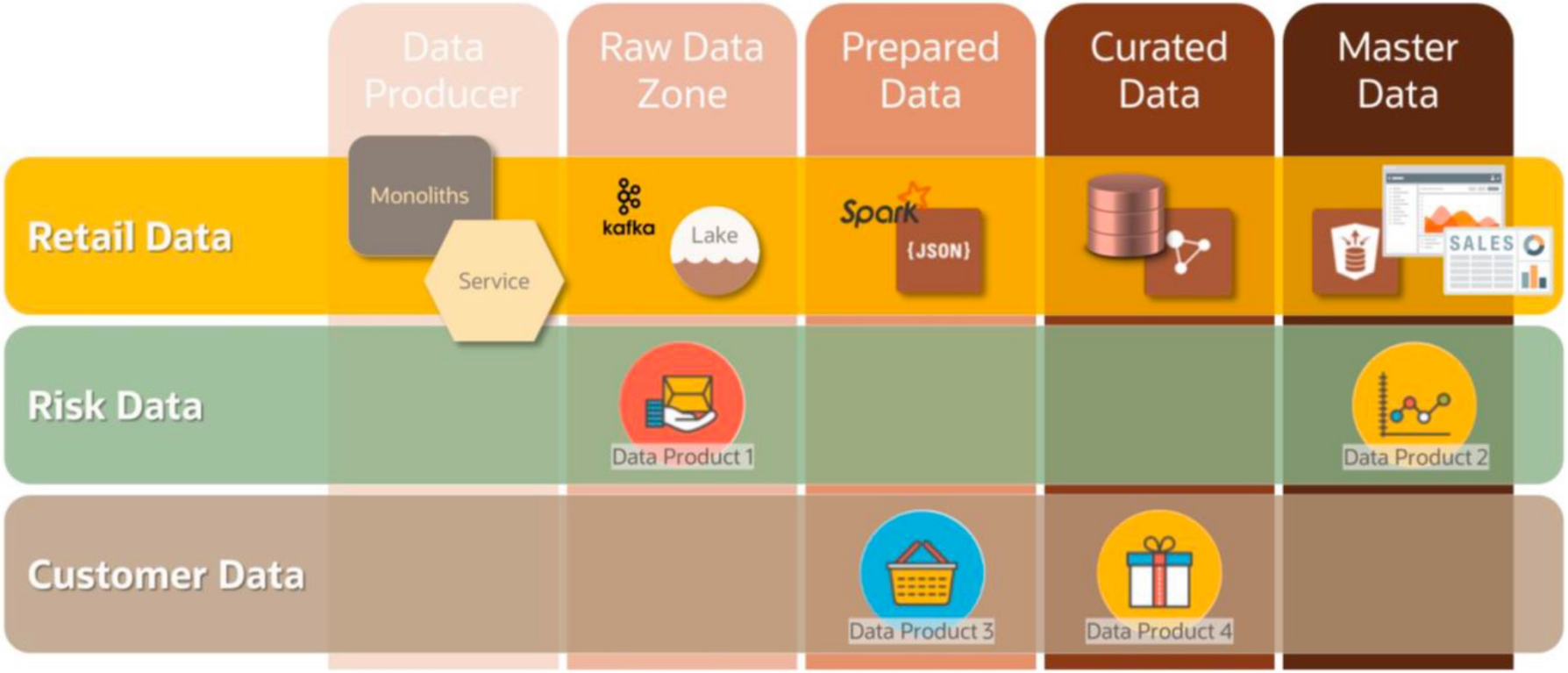
(modeling in 3NF and Dimensions)



Data Fabric / Mesh

(data domain decomposition spans many enterprise systems)

Example data domains, data zones, technologies and data products



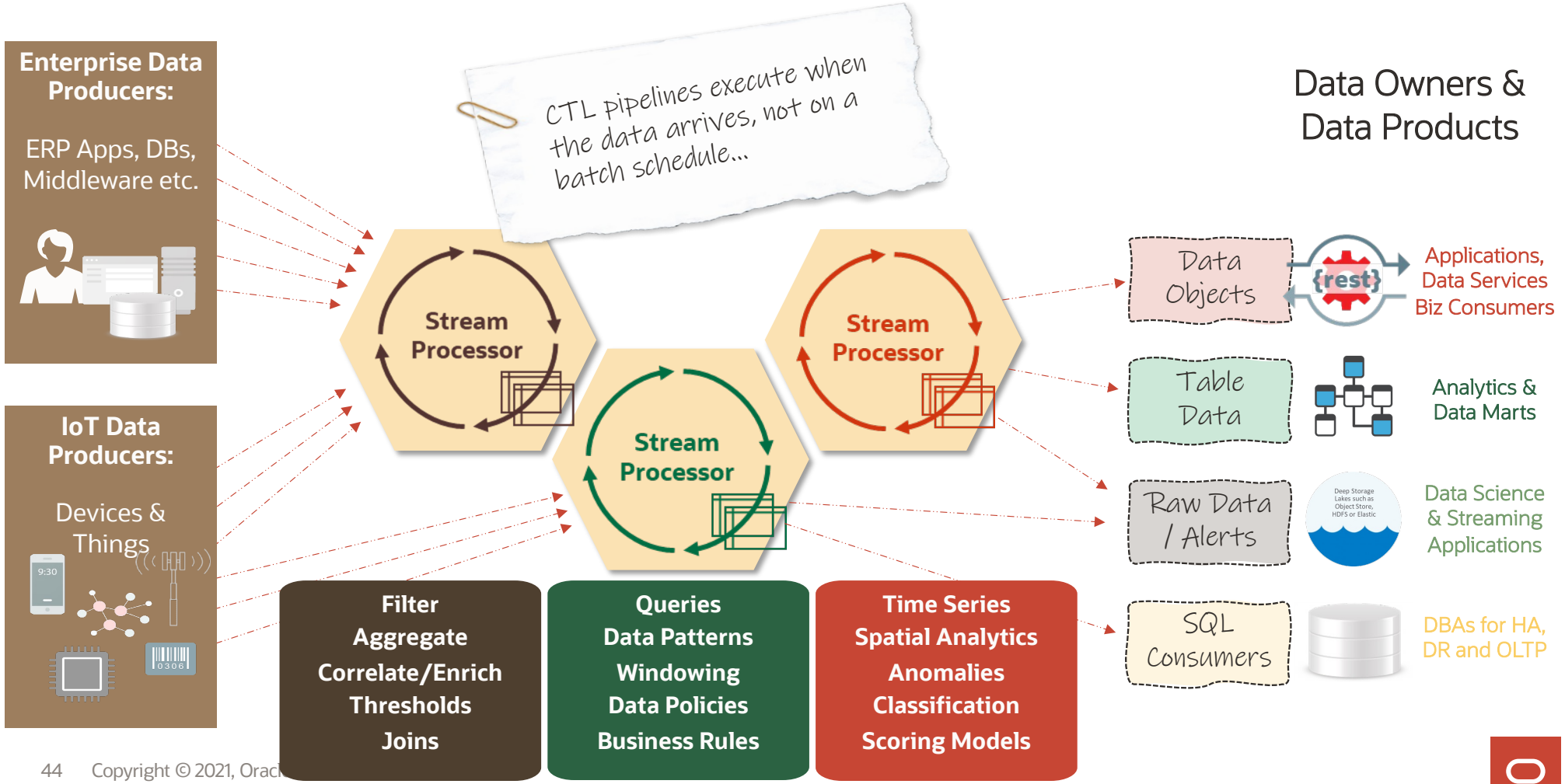
Example Data Zones and Data Products 1/2

DATA ZONE	DESCRIPTION	EXAMPLE DATA PRODUCTS	EXAMPLE TECH
Transient Zone	Data that is staged and prepared for ingestion, may be deleted upon ingest	N/A, ungoverned	File systems, producer provided queues
Raw Data Zone	Collection point for events and data emitted directly from applications and data stores (eg; producers)	Raw data feeds are often quite interesting to data scientists, who may place data product governance controls directly on the raw feeds.	Apache Kafka or JMS for events, Object storage or HDFS for files/bulk data
Prepared Data	In this zone, basic quality filters will have been applied to filter and format data into conforming syntax / structures	Data products for IT data pipelines (eg; ETL or stream integrations), eg; full supplemental change data on source customer tables	Self-service CDC, replication tools, stream integration

Example Data Zones and Data Products 2/2

DATA ZONE	DESCRIPTION	EXAMPLE DATA PRODUCTS	EXAMPLE TECH
Curated Data	Curation often implies models created by human activity, perhaps by a Data Steward or Data Analyst	Conventionally, an operational data store (ODS), or in an event-driven context a set of governed Topics with Avro/JSON payloads	Data catalogs, schema registry and pipelines for transformation
Master Data	As with conventional MDM systems, master data is a 'gold copy' of important data records (eg; customer, product, etc)	Master data records themselves are the data products, fully packaged and managed as discrete entities with their own lifecycle	Relational databases, graph engines, data pipelines, business rules
Secure Zones	There is an entire sub-discipline on this subject. Zones may include DMZ, Trusted or Restricted areas	Data products could include unencrypted financial transactions, account balances, hashed passwords, etc.	Security frameworks of applicable tools like Kafka, Hadoop and DBs

Trusted CTL (Continuous Transformation and Loading) Data Processing



Oracle GoldenGate: Trusted Bridge to a Data Mesh

Oracle GoldenGate technology is unique in the industry because it brings long-standing **trusted data replication** capabilities right alongside modern **decentralized microservices**, **polyglot payload capabilities**, and **stream processing** for CTL or time-series analytics.

Oracle GoldenGate

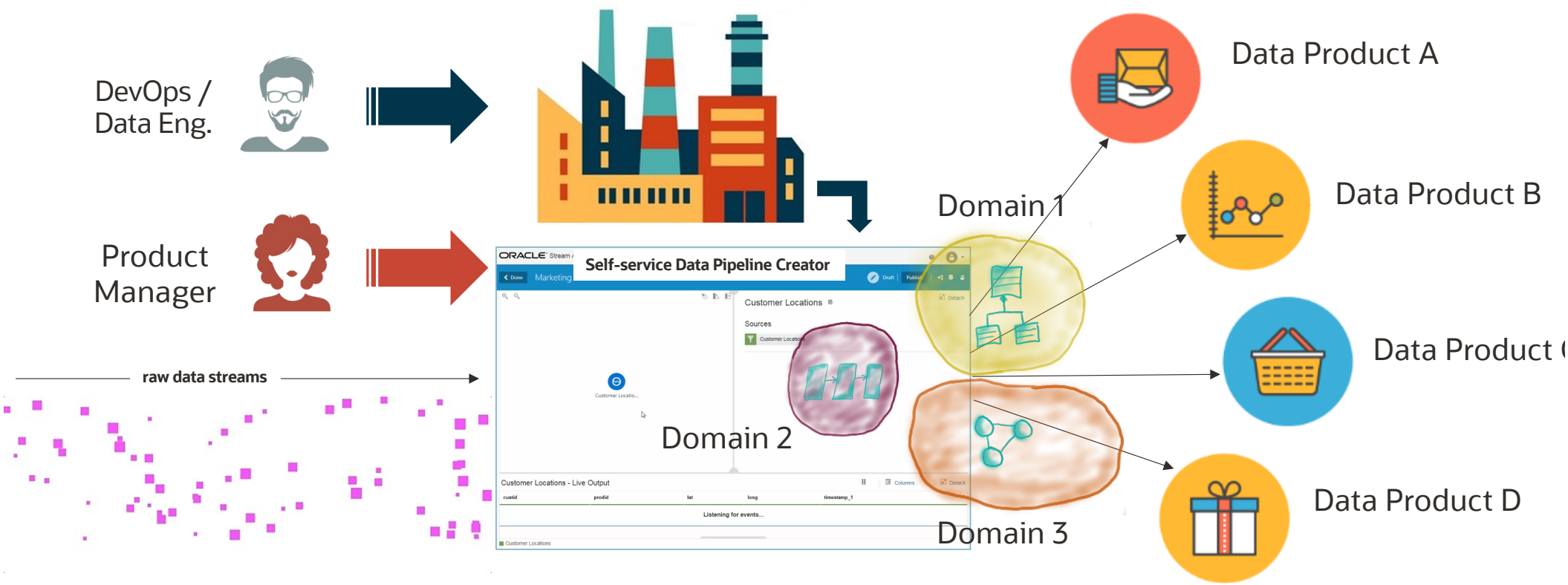
the most trusted solution for data integration

1000's of customers

84% of Fortune 100 use GoldenGate

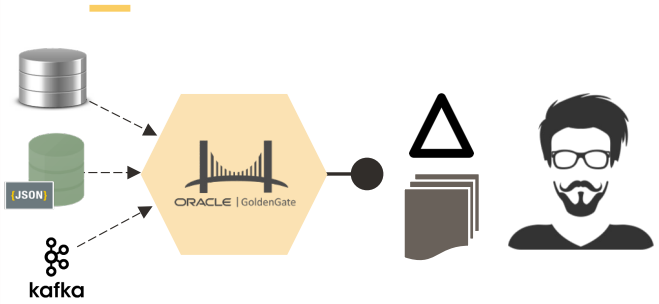
#1 in real-time data & Data Fabric Strategy

Single Pane of Glass for Real-Time Data Mesh

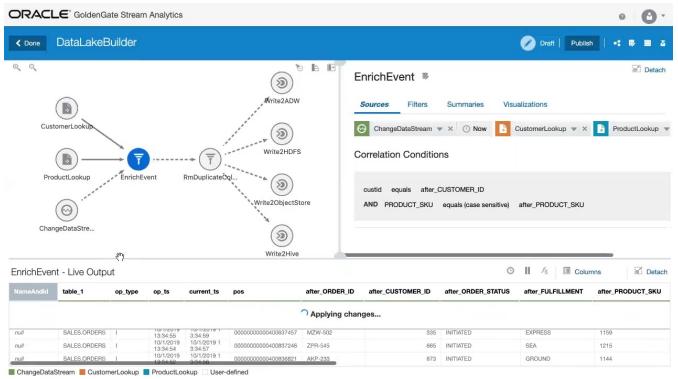


Data Products provided by GoldenGate

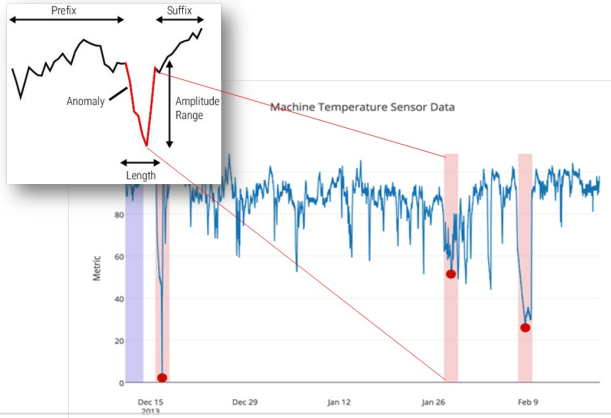
Curated Change Streams



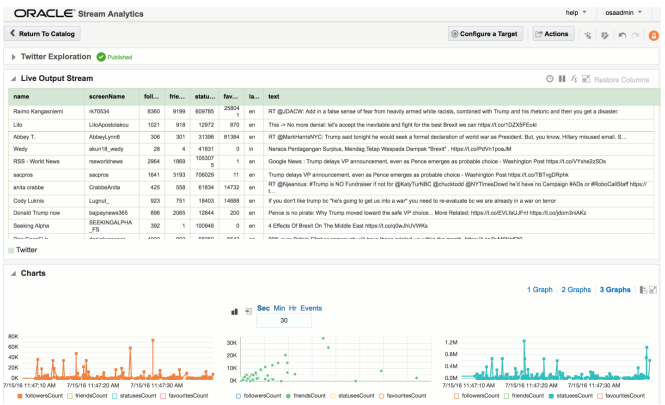
No Code Data Pipelines



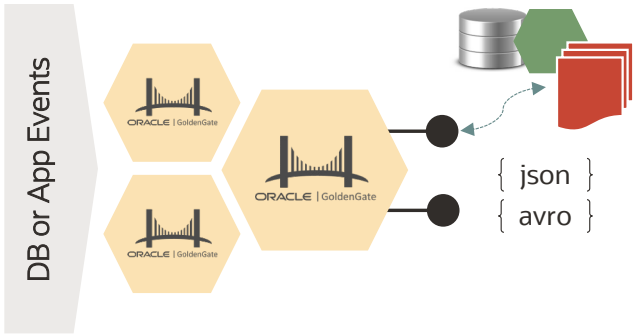
Time Series & Spatial Analytics



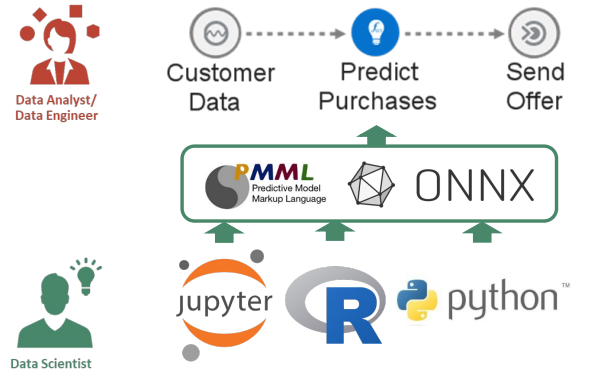
Real-time Dashboards / Alerting



Streaming Data Services

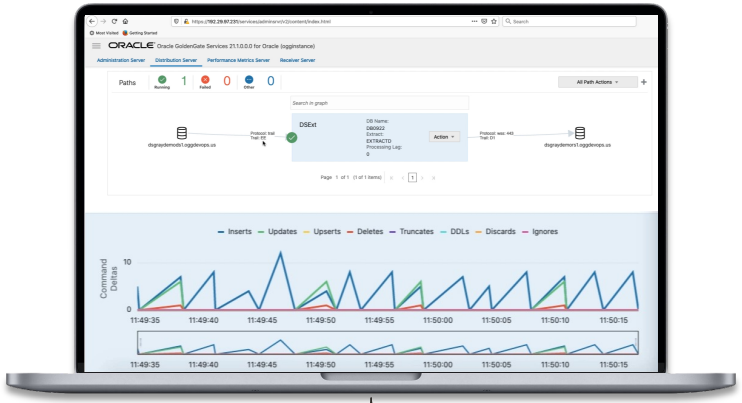


Production ML Scoring / Predictions

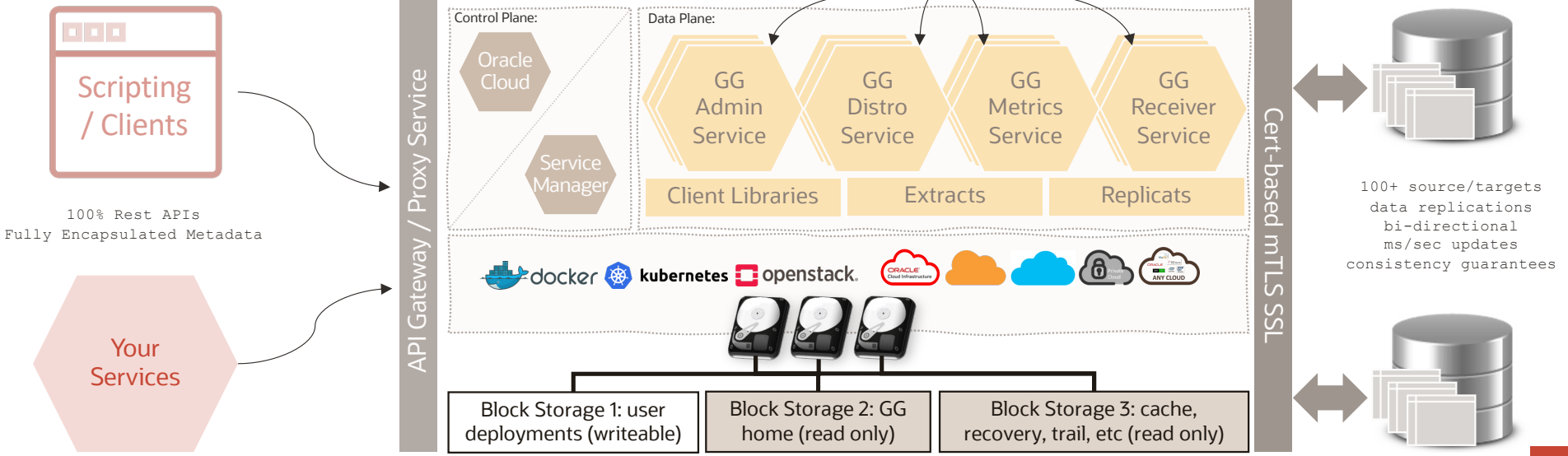


Microservices, Cloud-Native Architecture

GG microservices are fully encapsulated, C-based services, no additional app server or databases required



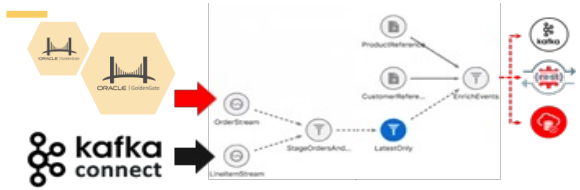
Embedded JavaScript GUI per microservice

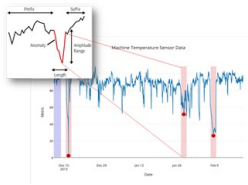


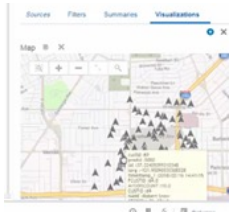
Data Fabric and Data Mesh Patterns Enabled by GoldenGate

Event Processing & Analytics

Data Pipelines | Data Transformation | GoldenGate Integrations | Time Series Analysis | Geo-Fencing | Predictive Analytics

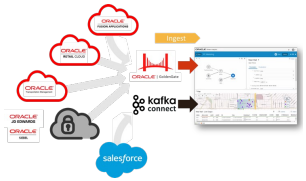






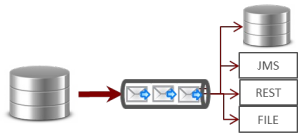
Non-Relational Events

Data Lake Ingest | Streaming Ingest | Cloud Ingest | Messaging Replication | NoSQL Replication | SaaS Replication



Real-time DB Topologies

Unidirectional | Bi-Directional | Peer-to-Peer | Broadcast | Consolidation | Distribution



Next Steps

What Next?

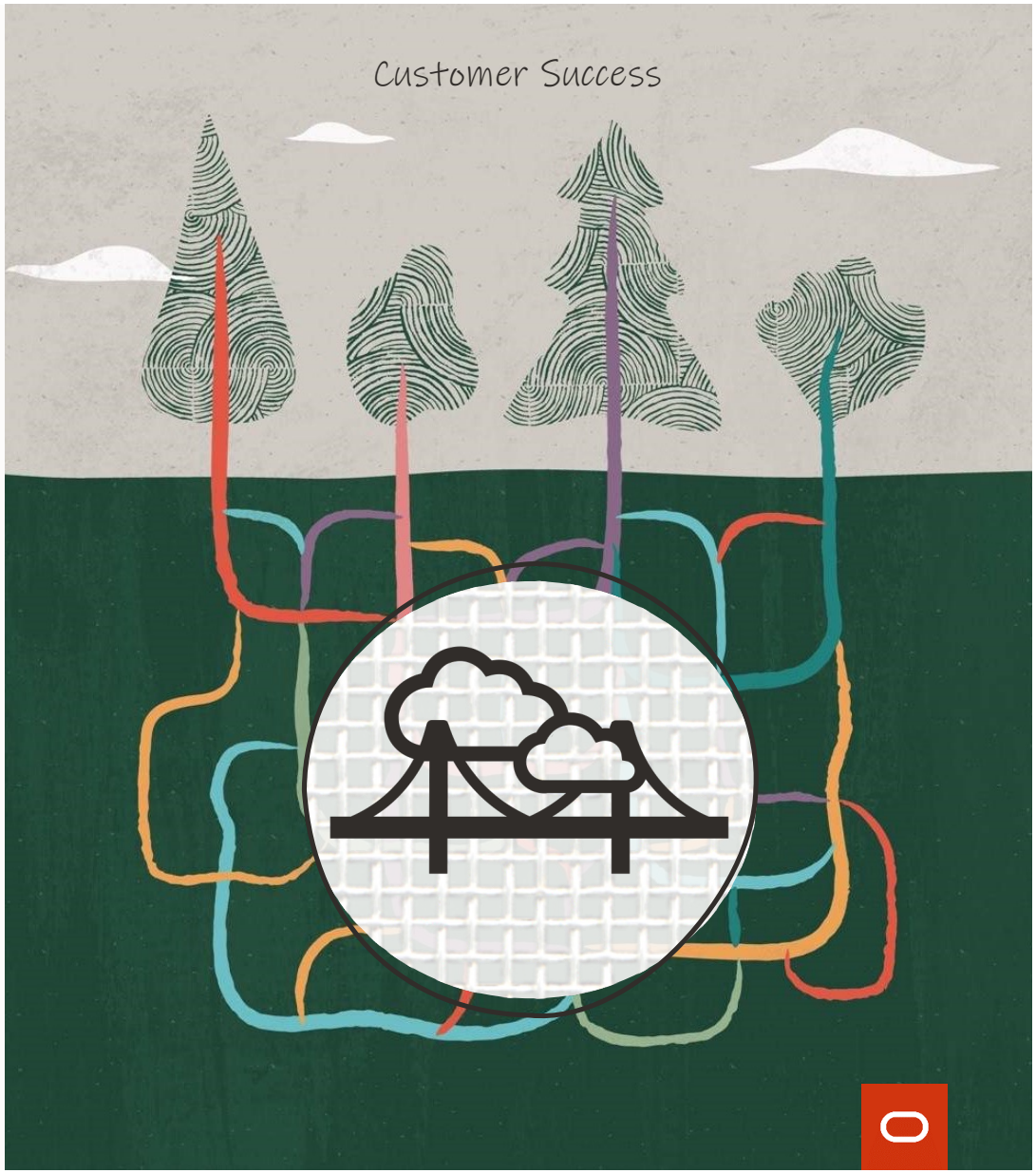
YouTube Channel for GoldenGate
https://www.youtube.com/c/oraclegoldengate

Try OCI GoldenGate yourself!
https://www.oracle.com/cloud/free/

2hr Hands-on Guided Lab:
https://apexapps.oracle.com/pls/apex/dbpm/r/livelabs/view-workshop?wid=797

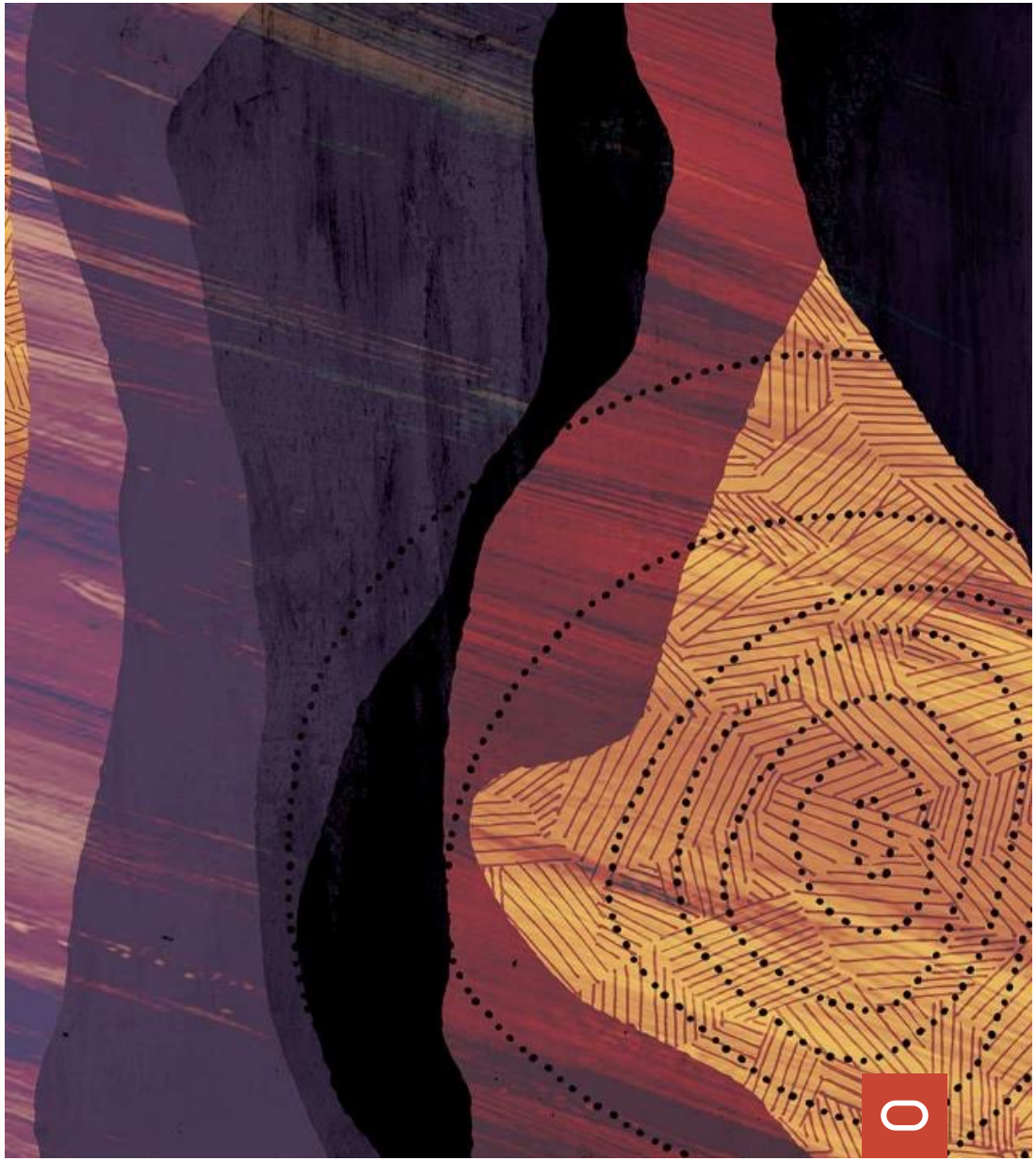
GoldenGate Data Mesh tech paper
https://www.oracle.com/a/ocom/docs/techbrief-enterprisedatameshandgoldengate.pdf

#1 in Data Fabric strategy
https://blogs.oracle.com/dataintegration/oracle_forresterwave_datafabric_2020



Thank You

Vili Tajnić
Principal Solution Engineer



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


ORACLE