

Control Data Mesh CADM



I need to create a slide for controls data mesh concept. There could be multiple datasets that would be consumed in the data mesh like complaints, violations, privacy incidents, data leak protection incidents, risk events, control issues etc. There is a firmwide solution called CADE that supports the concept of data mesh and within that we may implement our control data mesh. The sources of data could be various applications via APIs/Kafka/Email Data Exports or via data lakes and other data meshes using catalogues. Create a one slide representation. Include details of connectivity, implementation using databricks. Don't include too many technical details, represent functionally how this concept would work. Include a section of benefits from this concept, and some commentary of this too. Give me a story and outline.

One-Slide Storyboard — “Control Data Mesh on CADE”

1. Narrative Hook (headline + 1-liner)





“From scattered control records to one connected mesh.”

Our Control Data Mesh on CADE turns every complaints, privacy, or risk event feed into a governed, reusable data product that any controls team can plug into in minutes.

2. Visual Layout (left → centre → right)

Zone	What to Show (icons / call-outs)	Talking Point
Left: Source Domains	6 small tiles: Complaints • Violations • Privacy Incidents • DLP Incidents • Risk Events • Control Issues. Under the tiles, a slim band with connector icons: **API ⚡	Kafka ∞
Centre: CADE Control Data Mesh (Databricks Lakehouse)	One large shaded “mesh” block branded CADE + Databricks with sub-layers: • Bronze – Raw landing • Silver – Standardised Delta tables • Gold – Curated ‘Control Data Products’ Side ribbon: Unity Catalog • Lineage • Row-/column security • Policy engine	“Databricks medallion architecture governs, enriches, and versions every record, so you can trace a single complaint from source to report.”
Right: Consumers	4-icon vertical stack: Dashboards & MI AI/ML risk models Regulatory filings Real-time alerts & workflow	“Teams subscribe once via catalogue; mesh pushes fresh, trusted data to any tool.”

3. Benefits Ribbon (bottom)

 Faster insight	 Better control assurance	 Self-service & reuse	 Lower TCO
Streaming ingestion → near-real-time MI	Single schema + policy layer → audit-ready data	Catalog search + Delta Sharing → 80 % quicker onboarding	One lakehouse → no duplicate ETL & storage

4. Speaker Notes / Commentary

1. **Problem** – Control data lives in silos; analytics teams spend 70 % of time wrangling.
2. **Solution** – CADE’s Data Mesh treats each domain as a “product” with clear owners, SLAs, and discoverable contracts.
3. **How it works (functional)**
 - Publish → Standardise → Share.* Databricks handles quality rules, schema evolution, and fine-grained security; CADE exposes a single catalogue entry per product.
4. **Why it matters** – Risk leadership sees cross-domain patterns (e.g., privacy incidents driving customer complaints) in one view, enabling proactive mitigation.
5. **Next steps** – Prioritise onboarding of high-volume feeds (Complaints, DLP) and open the catalogue for pilot consumers in Q3.

5. Design Tips & Free Asset Packs

- **Template** – Use a three-column SmartArt “Chevron List” in PowerPoint or Google Slides; set centre column wider and shaded in mesh-inspired gradient.
- **Icons** – Font Awesome 6 (free tier) or **Feather Icons** for lightweight SVGs; color-code sources (blue), mesh (teal), consumers (green).
- **Accent shape** – Thin “data flow” arrows threaded through the three zones to convey continuous movement.

With this storyline you can walk an exec through **why** the mesh matters in <60 seconds, zoom briefly into **how** it is built on Databricks without drowning in tech detail, and close with quantifiable benefits.



For the center piece of control data mesh, add more details, both functionally and technically. Add more benefits as well.

Deep-Dive Add-on for the Center Block — “CADE × Databricks Control Data Mesh”

Use this detail to enrich the middle column of your existing slide without overcrowding it. Pick 5-7 call-outs that best serve your audience; park the rest in speaker notes.









1. Functional Layers (show as stacked swim-lanes inside the mesh box)

Layer	Purpose	Typical Activities
Bronze – Raw Ingest	Immutable landing zone	Auto-Loader streams from Kafka, API payloads & SFTP drops; schema inference & CDC capture
Silver – Standardised	Harmonised, row-level lineage	Delta Live Tables apply data contracts, PII tokenisation, field-level quality tests; failures routed to quarantine queue
Gold – Curated Data Products	Business-ready, versioned tables	Join across domains (e.g., Complaint ↔ Control Issue), snapshotted for point-in-time reporting, exposed via Delta Sharing
Catalogue & Governance	Discovery & policy enforcement	Unity Catalog + Immuta tags, automatic lineage graphs, SLA/ownership metadata visible to all producers & consumers

2. Key Technical Capabilities (bubble list on the right edge of the mesh)

- **Structured Streaming** ⇔ **Kafka / Event Hub** – true real-time controls MI
- **Delta Lake ACID & Time-Travel** – audit-perfect replay for regulators
- **Unity Catalog + Column-Level RBAC** – grant “Need-to-Know” views in seconds
- **Expectation-Driven Data Quality** (D LT + Great Expectations) – tests live with pipelines
- **MLflow Feature Store** – share labelled control-risk features with modelling teams
- **Delta Sharing** – publish secure read-only tables to other data meshes or Snowflake/S3 consumers
- **Serverless SQL / Photon** – sub-second queries for dashboarding without tuning clusters

3. Expanded Benefit Gallery (bottom ribbon – pick icons & 2-word descriptors)

 Real-time Visibility	 Proven Lineage	 Self-Serve Analytics	 Embedded Data-Quality	 Fine-Grained Security	 Reg-Ready Audit	 AI/ML-Ready	 Storage & ETL Savings
---	---	---	--	--	--	--	--

Speaker sound-bites

- *“Latency drops from days to minutes, letting Ops act before a violation becomes a headline.”*
- *“Every field carries its birth certificate — source system, timestamp, and transformation lineage, one click away.”*
- *“No more ticket queues: analysts browse the catalogue, click ‘Create notebook’, and run.”*
- *“Automated expectations catch null-spikes or schema drifts at ingest, not in production reports.”*
- *“Column-level masks mean privacy engineers can expose DLP context without revealing PII.”*
- *“When auditors ask ‘show me all records that fed Schedule R on Apr 15’, we time-travel the Gold table in seconds.”*
- *“Curated feature tables flow straight into risk-scoring models — no duplicate munging.”*
- *“One lakehouse tier eliminates three legacy warehouses and €1 M/year in license + infra costs.”*

4. Quick Design Cue

Inside the mesh rectangle:

- **Left half:** vertical medallion stack (Bronze/Silver/Gold)
- **Right half:** a vertical “capabilities” bullet rail with minimalist icons (⚡, 🗄️, 🔄...)
- Overlay a faint grid or ‘mesh’ pattern to subliminally cue federated nodes.

Keep the slide headline and source/consumer columns exactly as before; this richer center drives the discussion when you zoom in during the talk-track.



It has to be presented in a controls committee forum, it should be appealing and attractive, should be executive in nature and language. Should be slightly catchy as well, not too much. Accordingly word the notes and some language above.

Slide Blueprint — “CADE Control Data Mesh: One Fabric, Every Control”

1 Headline & Hook

“Turn scattered control logs into a single, trusted source of truth.”

A unified mesh that lets the Controls Committee see, trace, and act on risk signals in real-time.

2 Visual Story (three-column flow)

Sources _{'publish once'}	CADE Control Data Mesh _{'govern always'}	Consumers _{'act faster'}
Complaints · Violations · Privacy Incidents · DLP Alerts · Risk Events · Control Issues	Bronze Raw, immutable landings ↳ Silver Standardised & lineage-tracked ↳ Gold Curated “Control Data Products” Platform highlights — Unity Catalog with column-level RBAC — Auto-quality tests & quarantine queues — Delta Sharing to any cloud/db — Time-travel & point-in-time replay — MLflow feature store for risk models	Exec Dashboards 📊 Real-time Alerts 🔔 Regulatory Filings 📄 AI/ML Models 🤖

(Use chevron arrows to show left→centre→right flow; apply a subtle grid texture inside the mesh rectangle to cue “fabric”.)

3 Value Ribbon (bottom of slide)

Real-time Insight	End-to-End Lineage	Self-Serve Analytics	Built-in Data Quality	Precision Security	Audit-Ready Snapshots	AI-Ready Features	Lower TCO
--------------------------	---------------------------	-----------------------------	------------------------------	---------------------------	------------------------------	--------------------------	------------------

4 Tight Speaker Notes (≈60 sec talk-track)

1. **Why this matters** — Today our control records live in six silos; analysts spend more time reconciling than preventing risk.
2. **What we're proposing** — A *data-mesh on CADE*, powered by Databricks, that treats each domain as a governed “data product” discoverable in one catalog.
3. **How it works (executive view)**
 - Publish once — any app streams via API, Kafka, or secure file drop.
 - Govern centrally — the mesh enforces quality, lineage, and policy as data lands (Bronze→Silver).
 - Share instantly — curated Gold tables surface in dashboards, alerts, or models with zero additional ETL.
4. **Impact** — We move from weekly hindsight to same-day foresight; every metric is traceable back to source for regulators, and we save ~30 % in duplicate storage & tooling.
5. **Next step** — On-board Complaints and DLP feeds this quarter; pilot consumer dashboards for the Committee by Q4.

5 Tone & Design Tips

- **Template** — Three-column “Process” layout in your corporate theme; mesh column wider and shaded in light teal.
- **Icons** — Feather Icons (free SVG) keep it clean; animate arrows gently to signal live flow.
- **Language** — Use verbs that resonate with senior risk leaders: *trace, govern, prevent, assure*.
- **Catchiness** — One line is enough: “*One Fabric, Every Control.*” Place it under the slide title; let the visuals carry the rest.

Result: An executive-ready, visually engaging slide that tells the Controls Committee **what** the Control Data Mesh is, **how** it functions at a high level, and **why** it delivers immediate control assurance and risk-reduction benefits.



Use more visuals and lesser text specially in the main content of the slide. Do modifications.

“CADE Control Data Mesh — One Fabric, Every Control”

(Executive slide: 75 % visuals / 25 % text)

1 Headline Zone

One Fabric, Every Control

_{Real-time, governed data products for the Controls Committee}

2 Main Visual (take 70 % of the canvas)

left 25 %	centre 50 %	right 25 %
SOURCE ICON CLUSTER 6 circles with icons only: - 🗨️ Complaints - ⚠️ Violations - 🔒 Privacy - 🛑 DLP - 📊 Risk Events - ⚙️ Control Issues	CADE CONTROL DATA MESH (large teal rectangle, light mesh pattern) Bronze _(raw) 📄 Silver _(standardised) 🏆 Gold _(data products) 🏆 Along right edge of rectangle, place four small icon-only badges: 🔗 Lineage 🛡️ Security 📏 Quality ⌚ Time-Travel	CONSUMERS ICON STACK 📊 Dashboards 🔔 Alerts 📄 Reg Filings 🤖 Models

Connect **every source icon** to the mesh with a thin arrow; one big arrow exits the mesh to the consumer stack.

(No sentences—just icons + one-word labels. The speaker narrates the flow.)

3 Value Ribbon (footer, icon + 2-word text each)

🚀 Fast Insight 🔍 Full Lineage 📏 Built-in Quality 🛡️ Fine Security 🤖 AI-Ready 📉 Lower Cost

4 Speaker Notes (≤ 60 sec)

1. **Context** — Six disparate control feeds; siloed, slow, hard to audit.
2. **Mesh Flow** — Publish once → Govern in CADE → Consume anywhere.
3. **Assurance** — Lineage, quality, and security badges travel with every record.
4. **Impact** — Same-day visibility, regulator-ready snapshots, 30 % cost drop.
5. **Next Step** — On-board Complaints & DLP this quarter; pilot dashboard by Q4.

5 Design Cues







- **Icons** — Feather Icons or Font Awesome (free); keep everything monochrome except teal mesh.
- **Animation** — Fade-in arrows on click: sources → mesh → consumers.
- **Minimal text** — One-word labels only inside the visual; everything else is narration.
- **Catchiness** — Tagline under title; avoid more slogans.

Result: A clean, icon-driven slide that lets executives grasp the Control Data Mesh at a glance —while your narration supplies the depth.



The above looks great, give me details of each benefit in the value ribbon. Additionally expand speaker notes more.

Value-Ribbon Benefits — what each icon really delivers

Icon & Tag	Executive Pay-off	Typical Metric / Example
 Fast Insight	Streaming pipelines cut data latency from end-of-week to same-day dashboards, so Controls leaders act while an issue is still small.	MI refreshed every 15 minutes instead of weekly batch; 3× faster root-cause triage.
 Full Lineage	Every field carries its own “birth certificate” — source system, transformation steps, user ID. Regulators or Internal Audit can retrace a metric to the raw record in seconds.	Audit-query time drops from days to minutes ; zero “source unknown” findings.
 Built-in Quality	Quality expectations run in-flight (null checks, schema drift, reference-lookups). Bad rows go to quarantine before they hit a report.	< 1 % bad-data leakage vs. 8 % today; DQ score surfaces in dashboards.
 Fine Security	Column- and row-level RBAC, tokenisation, and dynamic masking mean privacy data is visible only to authorised eyes while still shareable firm-wide.	100 % policy enforcement at query time; aligns to GDPR & internal “Need-to-Know”.
 AI-Ready	Curated Gold tables double as a feature store for risk-scoring models; no extra wrangling. Data scientists spin up notebooks straight from the catalogue.	Model development cycle shrinks by 40 % ; features reused across 4+ use-cases.
 Lower Cost	One lakehouse tier collapses three legacy marts and removes duplicate ETL jobs. Pay-for-what-you-use compute scales to demand peaks only.	≈ 30 % annual run-rate savings (storage + licenses + support).

Expanded Speaker Notes (3-minute narrative)

Slide Purpose

“We’re here to unify six fragmented control datasets into a single, governed fabric, giving this Committee real-time oversight it has never had.”

1. Problem Statement

“Today, complaints, violations, privacy incidents, DLP alerts, risk events, and control issues live in separate silos. Analysts spend 70 % of their time reconciling numbers, and we still can’t answer simple lineage questions fast enough for regulators.”

2. Vision — “One Fabric, Every Control”

“We will publish each domain once into CADE, govern it centrally on Databricks, and make it instantly discoverable to any authorised consumer.”

3. High-Level Flow (point to the visual as you speak)

- **Left** — “Source systems keep doing what they do: push via API, Kafka, or secure file drop.”
- **Centre** — “CADE’s Data Mesh lands raw data (Bronze), standardises & unifies it (Silver), and creates curated, policy-governed products (Gold). The badges around the mesh — lineage, security, quality, and time-travel — travel with every record.”
- **Right** — “Dashboards, alert engines, regulatory filings, even ML models subscribe once and always have fresh, trusted data.”

4. Benefit Deep-Dive (tap the ribbon icons)

“Real-time Insight lets us spot a privacy spike on the same day, not next Tuesday. Full Lineage means an auditor can click from a dashboard metric to the exact source row and transformation history in seconds. Built-in Quality quarantines bad records at the gate. Fine-grained Security exposes DLP context

while masking PII. AI-Ready data accelerates predictive risk scoring, and we retire three legacy marts for a 30 % cost drop.”

5. Governance & Risk Assurance

“Unity Catalog logs every permission change; Delta Lake’s ACID controls prevent ‘ghost edits’; time-travel lets us replay any day in history exactly as it was delivered.”

6. Implementation Snapshot

“Build leverages existing CADE investment; Databricks clusters are already whitelisted. We target zero-footprint agents in source apps — just push to Kafka or call an API.”

7. Next 90 Days

“We will on-board Complaints and DLP feeds by end-Q2, publish the first cross-domain dashboard, and invite Committee members to test-drive lineage and time-travel views.”

8. Call to Action

“We’re looking for the Committee’s endorsement to prioritise these feeds and allocate a small SME squad for data-contract reviews. With that green light, you’ll see live dashboards before the next quarterly meeting.”

Delivery Tip

Keep the slide visually simple; let your voice provide the colour. Pause briefly on each ribbon icon so executives can connect the picture to the payoff.



Now create a second slide based on this concept. That should be more for technical audience. Add few functional details and more technical details in the main portion of the slide. Modify the values and the headers accordingly.

Slide 2 — “CADE Control Data Mesh: Technical Blueprint”

(For engineering, data-ops, and architecture teams)

1 Header Band

From Raw Feeds to Governed Products — Under the Hood

2 Main Canvas (three-panel schematic, ~70 % of slide)

Ingestion & Contracts	Lakehouse Processing on CADE	Serving & Consumption
Connect <ul style="list-style-type: none"> • REST API (JSON/Avro) • Kafka / Event Hub (ProtoBuf) • SFTP + Auto-Loader (CSV/Parquet) • DB CDC → Debezium Data Contracts <ul style="list-style-type: none"> ✓ Versioned schemas (Confluent Registry) 	Pipeline Orchestration <ul style="list-style-type: none"> • Delta Live Tables (DLT) — declarative • Structured Streaming for Kafka/CDC • Expectation Suite (Great Expectations) • Merge-on-read CDC (Change Data Feed) Transform Layers	Query & Share <ul style="list-style-type: none"> • Serverless SQL endpoints (Databricks SQL) • Delta Sharing — secure open-format feed to Snowflake/S3/BigQuery • REST Data API (CADE catalog wrapper) Consumers <ul style="list-style-type: none"> 📊 Power BI / Tableau dashboards 📄 Regulatory & SOX filings (Informatica ETL picks Gold)

Ingestion & Contracts	Lakehouse Processing on CADE	Serving & Consumption
SLAs & ownership tags (YAML) Landing Bronze tables (raw) stored as Delta `Landing_*`	<ul style="list-style-type: none"> ♦ Bronze → Silver – schema align, PII tokenise, soft-delete flag ♦ Silver → Gold – joins, SCD-2 snapshots, KPI materialisation Governance & Security <ul style="list-style-type: none"> • Unity Catalog — lineage graph, tag-based access • Row / column RBAC, dynamic masking (Immuta) • Time-Travel / Audit Mode (<code>`SELECT ... VERSION AS OF`</code>) Dev & Ops <ul style="list-style-type: none"> • Git-backed notebooks + CI/CD via Databricks Repos • Infra-as-code (Terraform modules for clusters, ACLs) • Pipeline health in Lakehouse Monitoring dashboard 	Feature Store → MLflow models (risk scoring, anomaly detectors) Event-driven alerts (DB notebook → ServiceNow API)

(Use colour coding: ingestion = purple, processing = teal, serving = green. Arrows with latency labels: “< 5 s stream” or “hourly batch”).

3 Tech Value Strip (icon + metric)

Declarative Pipelines _{DLT reduces ETL code by ~50 %}	Built-in QoS _{1 % bad rows reach Silver}	Zero-Trust Access _{Row & column masks at query-time}	Reproducible History _{Time-travel to any version in <1 s}	Elastic Compute _{Serverless SQL auto-scales to 100 conns}
--	---	---	--	--

4 Speaker Notes (technical walk-through — ~3 min)

1. Ingestion Discipline

“Every upstream team publishes through a signed data contract: schema in Avro, SLA targets embedded. Auto-Loader handles parquet drops; Kafka topics are schema-validated at the gateway.”

2. Bronze Layer

“Raw files land unchanged, partitioned by EventDate and Source. This preserves immutability for audits and replay.”

3. Silver Layer

“DLT applies harmonised field names, ISO-8601 timestamps, and runs checkpointed Expectations — null checks, referential tests, privacy tokenisation. Failures route to a managed quarantine table.”

4. Gold Products

“We snapshot cross-domain joins into Slowly Changing Dim 2 tables; KPIs like ‘Open Control Issues by Region’ materialise every 30 min. Gold tables are registered as governed assets in Unity Catalog.”

5. Governance & Security

“Column tags drive RBAC — for example, any column tagged ‘PII’ auto-masks unless role = PrivacyEngineer. Unity Catalog’s lineage graph satisfies EU DORA traceability in out-of-the-box format.”

6. Ops & Observability

"DLT event logs stream to Lakehouse Monitoring; red/yellow/green status surfaces in Grafana. Pipelines version through Git; Terraform replays infra for higher-env promotion."

7. Serving Patterns

"Business reports hit Serverless SQL; ML engineers pull features via Feature Store or direct Delta Sharing to their personal S3."

8. Performance Snapshot

"Prod cluster: 14 TB Delta; 15 min watermark for Kafka feeds; 2.3 s p95 dashboard query; storage cost cut by 28 % after decommissioning legacy Oracle marts."

9. Next Technical Milestones

"Enable automatic schema evolution with DLT 2.0; migrate remaining batch SFTP jobs to Kafka Connect; roll out fine-grained lineage push to Collibra."

10. Ask

"Approve sandbox credentials for three additional source systems and endorse the data-contract checklist going forward."

Design Reminder

Let the schema diagram dominate; keep bullet text off the canvas (use call-out bubbles). Colour badges on value strip draw the eye without crowding the architecture view.



Explain these technical concepts to me, also give functional and technical examples in between wherever needed.

Below is a "cheat-sheet" that unpacks every building-block from the **Technical Blueprint** slide, pairing each term with a plain-English explanation **and** a short functional + technical example so you can see how it plays out day-to-day.

1. Ingestion & Data-Contracts

Concept	What it means	Example in practice
Data Contract	A version-controlled agreement that locks down the <i>shape, semantics and SLA</i> of a feed before it ever lands.	<i>Functional:</i> The Privacy team promises to deliver a field called <code>`incident_type`</code> with values from a controlled list and an hourly freshness SLA. <i>Technical:</i> Contract stored as an Avro schema in Confluent Registry plus a YAML file that adds <code>`owner=jane.doe`</code> and <code>`max_lag_mins=60`</code> .
REST API / Kafka / SFTP	Multiple on-ramps so source systems can publish in the format they already speak.	<i>Functional:</i> A complaints portal posts JSON to <code>`POST /cade/v1/complaints`</code> . <i>Technical:</i> An on-prem mainframe pumps change events to the <code>`violations_cdc`</code> Kafka topic with Protobuf payloads.
Debezium CDC	Open-source connector that watches a relational DB's redo logs and	<i>Functional:</i> Instead of nightly extracts, control-issue inserts from Oracle appear within seconds. <i>Technical:</i> Debezium emits <code>`{op: "c", after:</code>

Concept	What it means	Example in practice
	streams Change Data Capture as Kafka events.	<code>{...}</code> messages; Structured Streaming picks them up downstream.

2. Lakehouse Processing on CADE (Databricks)

Concept	What it means	Example in practice
Bronze / Silver / Gold	The “medallion” pattern. • Bronze – raw, immutable • Silver – standardised & quality-checked • Gold – ready-to-consume “data products”	<i>Functional:</i> An auditor asks for the untouched payload that drove last week report – you time-travel the <i>Bronze</i> table. <i>Technical:</i> <code>`bronze.privacy_incident`</code> → <code>`silver.privacy_incident_curated`</code> → <code>`gold.control_privacy_r`</code> (SCD-2 snapshot).
Delta Live Tables (DLT)	Declarative pipeline engine: you write SQL/Python that says <i>what</i> the table should look like; DLT handles cluster spin-up, retries, testing and lineage.	<i>Example code:</i> <code>`CREATE STREAMING LIVE TABLE silver_privacy_incident AS SE hash(PAN) AS pan_token FROM LIVE.bronze_privacy_incident W _expect(!isNull(id))`</code>
Structured Streaming	Micro-batch engine in Spark/Databricks that treats an unbounded Kafka topic “like a table that keeps growing”.	<i>Functional:</i> Controls MI dashboard refreshes every two minutes without re-a full dataset. <i>Technical:</i> <code>`readStream.format("kafka").option("subscribe","violations</code>
Expectation Suite / Great Expectations	Declarative data-quality checks that fail fast.	If <code>`severity`</code> must be one of {LOW, MEDIUM, HIGH}, an invalid <code>`CRITICAL`</code> value is quarantined and paged to the data steward.
Merge-on-Read CDC / Change Data Feed	Delta Lake can emit only the rows that changed (<code>`_change_type`</code> , <code>`_commit_version`</code>).	A Silver table auto-maintains an SCD-2 view without bulk updates.
SCD Type 2 Snapshot	Technique to keep historical versions of a record with <code>`effective_start`</code> / <code>`effective_end`</code> dates.	Gold table <code>`control_issue_dim`</code> lets you recreate how a control looked <i>point in time</i> for SOX evidence.

3. Governance & Security

Concept	What it means	Example in practice
Unity Catalog	Databricks’ central catalogue + policy engine; stores tables, view lineage, tags and permissions.	Business user searches “ <i>privacy</i> ” and instantly finds the Gold data product, owner, SLA, and full lineage graph.

Concept	What it means	Example in practice
Row / Column RBAC & Dynamic Masking	Fine-grained policies that show or hide data at query time.	<i>Functional:</i> Analysts see <code>`customer_name`</code> ; interns see <code>`***`</code> . <i>Technical:</i> <code>`GRANT SELECT(mask(customer_name)) ON TABLE ... TO ROLE intern;`</code>
Time-Travel / Audit Mode	Query any Delta Lake table <i>as-of</i> a commit version or timestamp.	Auditor: "Show me what the Gold KPIs looked like on 15 Apr 2025, 09:00 UTC." → <code>`SELECT * FROM gold.control_kpi VERSION AS OF 1271`</code> .

4. Serving & Consumption

Concept	What it means	Example in practice
Serverless SQL	Elastic, no-ops query endpoint that auto-scales & suspends; pay per second.	A Tableau workbook hits <code>`SELECT region, open_issues FROM gold.control_issue_rollup`</code> and spins up capacity only while the analyst is online.
Delta Sharing	Open protocol that streams Delta tables to <i>any</i> platform (Snowflake, pandas, BigQuery) over HTTPS.	Risk partner in another LoB connects via Python: <code>`df = delta_sharing.load_table("s3://share/gold/control_privacy_rollup")`</code> – no VPN, no replica.
Feature Store / MLflow	Managed registry linking features to training & production models.	Data scientist registers <code>`avg_time_to_close_issue`</code> as a feature; MLflow auto-tracks which model version used which feature snapshot for provenance.
Event-Driven Alerts	Notebooks or Delta triggers call downstream systems (ServiceNow, Slack) when rules fire.	If <code>`open_privacy_incidents > 50`</code> , notebook posts a ServiceNow ticket with offending record list.

5. DevOps & Observability

Concept	What it means	Example in practice
Databricks Repos + CI/CD	Git-backed notebooks; pull requests trigger unit tests and DLT pipeline deployment.	PR adds a new <code>`privacy_score`</code> column; automated tests confirm schema contract; Terraform redeploys the pipeline.
Terraform Modules	Infrastructure-as-code for clusters, RBAC, secrets.	Developers provision a test Lakehouse with one command: <code>`terraform apply -var env=dev`</code> .
Lakehouse Monitoring	Native dashboard for pipeline health, throughput, errors.	SRE sees latency spike on <code>`violations_stream`</code> ; clicks into lineage to pinpoint slow source topic.

Putting it all together — Mini Walk-Through

1. A new **control-issue record** is inserted in the source Oracle DB.
2. **Debezium** captures that insert and publishes a Kafka message within ~2 s.
3. **Structured Streaming** ingests the topic into *Bronze*.
4. **DLT** transforms to *Silver* with Great Expectations enforcing the contract. A bad ``status_code`` would be quarantined automatically.
5. A **Gold SCD-2 snapshot** rolls up the issue with prior history; Unity Catalog tags it ``domain=controls``, ``pii=false``.
6. **Serverless SQL** refreshes the Exec dashboard; simultaneously, the **Feature Store** updates a risk-scoring model.
7. A **dynamic mask** ensures that only users in the ``Control_Manager`` role can see the employee name linked to the issue.
8. **Time-travel** lets an auditor rewind the Gold table to the instant the Q1 report was generated, proving data immutability.

Why these mechanics matter

- **Controls Assurance** – lineage + time-travel satisfy EU DORA, SOX and OCC “golden-source” evidence.
- **Operational Agility** – declarative pipelines and contracts cut onboarding time from *months* to *weeks*.
- **Risk Insight** – streaming ingestion and Delta Sharing mean AI models detect patterns (e.g., DLP spikes leading to complaints) before they become headlines.
- **Cost & Compliance** – one encrypted lakehouse replaces multiple licensed silos while enforcing least-privilege at the column level.

Use this as your deeper narrative when the technical audience asks “*how does it really work?*”.



Now create a third slide that should be a combination of technical and functional for mixed audience. That helps the functional audience understand the technology used in simple language, not extremely technical.





Slide 3 — “Control Data Mesh: What Happens Under the Hood (Plain-English View)”

(Bridges business & tech in one picture)

1 Headline

From Control Events to Actionable Insight — in Four Simple Steps

2 Main Visual (4-step horizontal flow, icons + one-line captions)

1. Publish	2. Organise	3. Protect & Govern	4. Use & Act
 Apps send data _{(complaints, privacy hits, risk events) via API or secure feed}	 Mesh sorts & cleans _{raw → standard → ready-to-use tables}	 Built-in guard-rails _{quality tests, audit trail, row/column security}	 Everyone consumes _{dashboards, alerts, ML models, filings}






(Use a gentle left-to-right arrow beneath the icons.)

3 “Business Need → Tech That Delivers” Cheat-Sheet (centre of slide)

Business Need	How the Mesh Satisfies It (plain language)	The Tech Piece Doing the Work
See issues same day, not next week	Data streams in continuously; dashboards refresh every few minutes.	Kafka + Databricks “Structured Streaming”
Trust every metric	Each record carries its source, timestamp, and any transformations.	Delta Lake “lineage” & time-travel
Block bad or missing data	Automated checks stop dirty rows before they hit reports.	Built-in data-quality rules (Great Expectations)
Show sensitive fields only to the right people	Names or IDs are hidden unless your role requires them.	Column-level security in Unity Catalog
Reuse data for AI and compliance filings	One curated table feeds both models and regulatory reports—no extra ETL.	Gold “data products” shared by Delta Sharing

(Keep wording conversational; avoid acronyms unless shown at right in the tech column.)

4 Value Ribbon (bottom)

 Faster Insight  Proven Lineage  Quality by Default  Right-Sized Access  AI-Ready Data

5 Speaker Notes (≤ 2 min)

1. **Story in 10 seconds** — “We collect control data once, groom it automatically, guard it with strong policies, and serve it everywhere.”

2. **Why mixed audience should care**

- Business owners: quicker, cleaner MI with trace-back.
- Tech teams: fewer hand-offs, declarative pipelines instead of ad-hoc scripts.

3. **Key takeaway** — *Same data, trusted everywhere*—whether you are closing a control issue, filing a report, or training a risk model.



I don't like this slide, give me more ideas. Be executive but semi formal, not too formal, not too informal.

Slide 3 — “Controls Data Mesh: How It *Really* Works (At a Glance)”

Tagline: “One pipeline. Two guarantees: trust & speed.”

Layout Concept

A “metro-map” across the slide:

- **Four stops** on a single coloured line (left → right).
- Each stop has a **station name + icon** on top (business language) and a **platform sign** underneath (1–2 tech keywords).
- A thin legend at the bottom shows benefits as “commuter perks.”

less

```
[ Feeds In ] — [ Clean & Tag ] — [ Guard & Trace ] — [ Share & Act ]
  🚉           🚉           🚉           🚉
API / Kafka | Golden Names | Locks & Logs | Dashboards / AI
```

(Use your house colours; keep icons monochrome except the line itself so the eye follows the journey.)

Station Details

#	Station (Business wording)	Simple Tech Signpost	10-second Script
1	Feeds In _{Apps drop control events}	API • Kafka	“Everything—from complaints to DLP hits—rides the same rail within 5 s.”
2	Clean & Tag _{Names aligned, junk filtered}	DLT • Quality Rules	“Automated rules fix dates, flag gaps, and tag data owners—no spreadsheets.”
3	Guard & Trace _{Security + lineage baked-in}	Unity • Time-Travel	“Every row knows who touched it and masks PII on the fly.”
4	Share & Act _{Dashboards, alerts, ML models}	Serverless SQL • Delta Sharing	“Stakeholders plug-in once and always see the freshest, audited truth.”

Commuter Perks (Value Strip)

 Same-Day Insight	 Proven Lineage	 Always Clean	 PII Safe	 Model Ready
--	--	--	--	---

(Appear as coloured tickets clipped to the bottom of the metro map.)

Speaker Notes (90 sec)

1. **Set the scene** – “Think of our control data as travellers: they hop on at source systems and ride a single metro line to their destination.”
2. **Stop 1 – Feeds In** – “Whether an API push or a Kafka stream, everything boards the same track within seconds.”
3. **Stop 2 – Clean & Tag** – “Databricks Live Tables scrub and standardise the data automatically; any rule break kicks off an alert before it reaches reporting.”
4. **Stop 3 – Guard & Trace** – “Unity Catalog stamps every record with lineage and applies masks—so privacy and audit teams are covered from day one.”
5. **Stop 4 – Share & Act** – “Dashboards, ServiceNow alerts, even AI models read from the *exact same* Gold tables—no duplicate ETL, no ‘version-of-the-truth’ debates.”
6. **Land the value** – “Net result: same-day insight, regulator-ready lineage, and data you can trust for both Board packs and machine-learning.”

Design Tips

- **Icons:** Feather or Phosphor (free); keep them line-style to match the metro theme.
- **Colour:** Use one bold line colour (e.g., teal) with soft greys for everything else—focus stays on the journey.
- **Animation (optional):** Motion-path the teal dot from left to right during your talk; each stop pops a short caption.

This concept speaks to **business leaders** in plain language—“Feeds In, Clean, Guard, Share”—while still nodding to the **key tech enablers** they’ve heard of. The metro metaphor is familiar, visually tidy, and makes the journey—and its guarantees—instantly clear.