From Source to a Working Agent in 3 Hours:

Building an AI Ready Lakehouse in Microsoft Fabric

DATA SATURDAYS HOLLAND



Thank you sponsors!



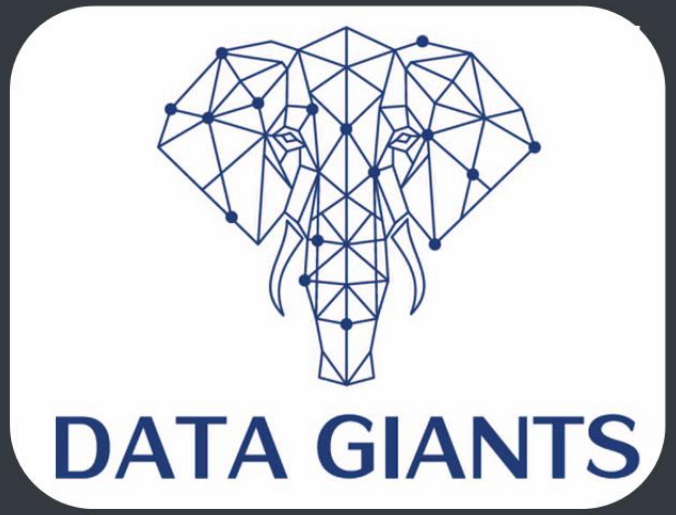




















Data Saturdays

Marc Lelijveld

Architect & Technical Advocate
Macaw Netherlands

Microsoft® Most Valuable Professional

linkedin.com/in/MarcLelijveld

Data-Marc.com

DutchFabricUsergroup.com

FAVORITE STUFF:





What are your key objectives to Learn today?

What is your Fabric Experience?

Who are you?

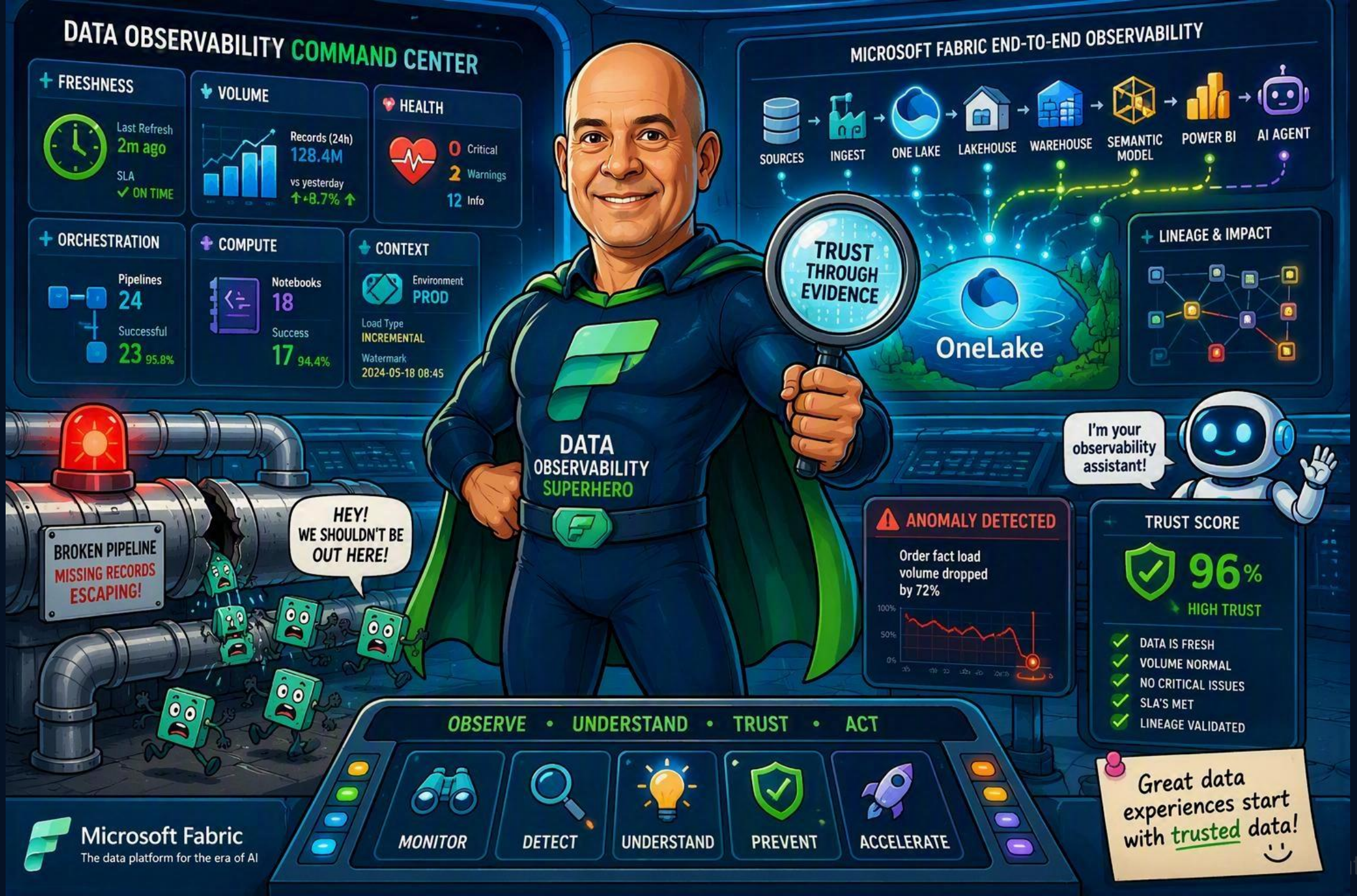
Session outline

3 hours in total

- Building the foundation – theory only
- Building the semantic layer and agents on top – demo packed!

Before building an AI agent, we need an AI-ready data foundation.

Let's build one


DATA OBSERVABILITY COMMAND CENTER
FRESHNESS
Last Refresh
2m ago
SLA
ON TIME
VOLUME
Records (24h)
128.4M
vs yesterday
+8.7%
HEALTH
0 Critical
2 Warnings
12 Info
ORCHESTRATION
Pipelines
24
Successful
23 95.8%
COMPUTE
Notebooks
18
Success
17 94.4%
CONTEXT
Environment
PROD
Load Type
INCREMENTAL
Watermark
2024-05-18 08:45
MICROSOFT FABRIC END-TO-END OBSERVABILITY
SOURCES
INGEST
ONE LAKE
LAKEHOUSE
WAREHOUSE
SEMANTIC MODEL
POWER BI
AI AGENT
LINEAGE & IMPACT
TRUST THROUGH EVIDENCE
OneLake
DATA OBSERVABILITY SUPERHERO
BROKEN PIPELINE
MISSING RECORDS ESCAPING!
HEY! WE SHOULDN'T BE OUT HERE!
I'm your observability assistant!
ANOMALY DETECTED
Order fact load volume dropped by 72%
100%
50%
0%
TRUST SCORE
96%
HIGH TRUST
DATA IS FRESH
VOLUME NORMAL
NO CRITICAL ISSUES
SLA'S MET
LINEAGE VALIDATED
OBSERVE • UNDERSTAND • TRUST • ACT
MONITOR
DETECT
UNDERSTAND
PREVENT
ACCELERATE
Microsoft Fabric
The data platform for the era of AI
Great data experiences start with trusted data!



Data-Marc.com



Data-Marc.com

The promise: one reliable path from source to intelligence



The platform is successful when every step is repeatable, observable and governed.

Data-Marc.com

Why AI projects fail before the model

Change everywhere	Code everywhere	Trust Nowhere
❖ Sources evolve	❖ Copy-paste pipelines	❖ Unknown freshness
❖ Schemas drift	❖ One-off notebooks	❖ Unknown completeness
❖ Rules change	❖ Hidden parameters	❖ Unclear lineage
❖ Consumers multiply	❖ Inconsistent logging	❖ Silent downstream impact



Platform engineering turns variability into controlled configuration.

Data-Marc.com

Platform engineering by design



Developer experience

- ❖Reusable templates
- ❖Predictable contracts
- ❖Environment-aware configuration

Operational experience

- ❖Shared run identity
- ❖Consistent telemetry
- ❖Actionable failure context

Governance experience

- ❖ Shared run identity
- ❖ Consistent telemetry
- ❖ Actionable failure context

Build the paved road before asking every team to drive faster

Data-Marc.com

STANDARDIZE

Data-Marc.com

The architecture: data plane + control plane

DATA PLANE

Pipelines
Notebooks
Lakehouse

Execute ingestion, transformation, validation and publishing using the configuration supplied by the control plane.

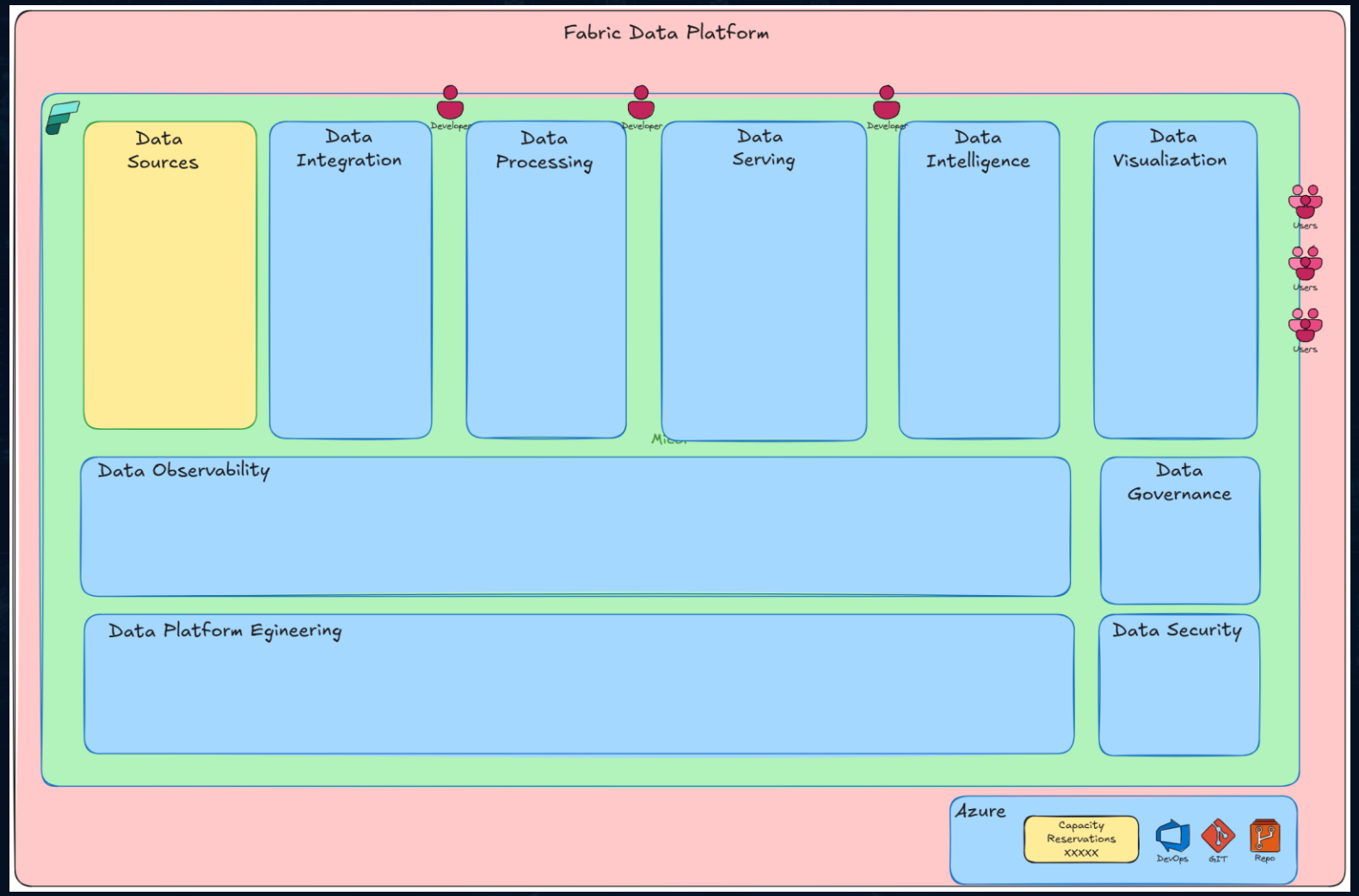
CONTROL PLANE

Fabric SQL Database

Metadata
Parameters
Dependencies
Rules
Run state
Logging

Data-Marc.com

Platform engineering by design



Data-Marc.com

Platform engineering by design

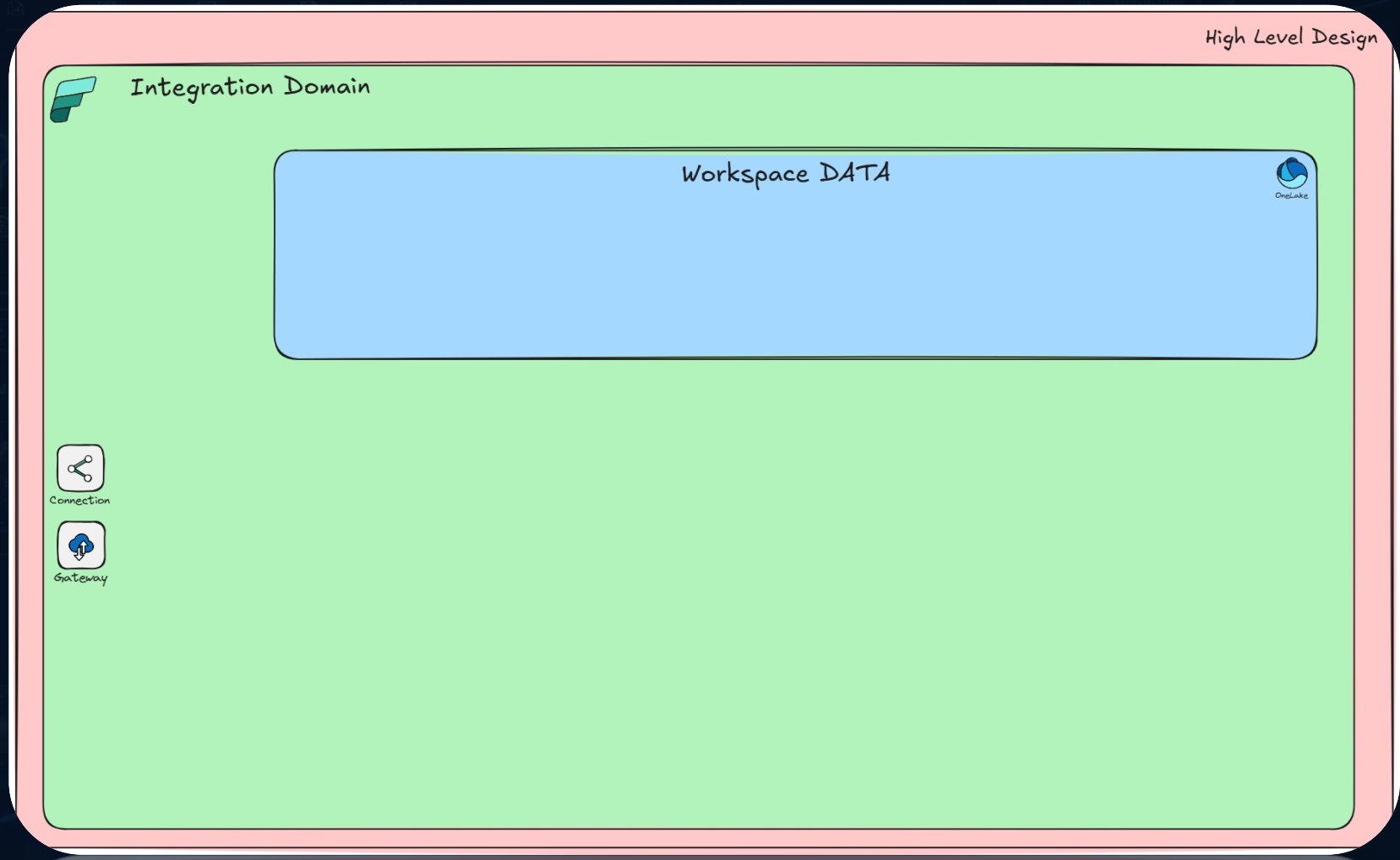
- Azure subscription
- Fabric Capacity (Trial)
- Fabric Administrator role



Data-Marc.com

Platform Engineering (Data Workspace)

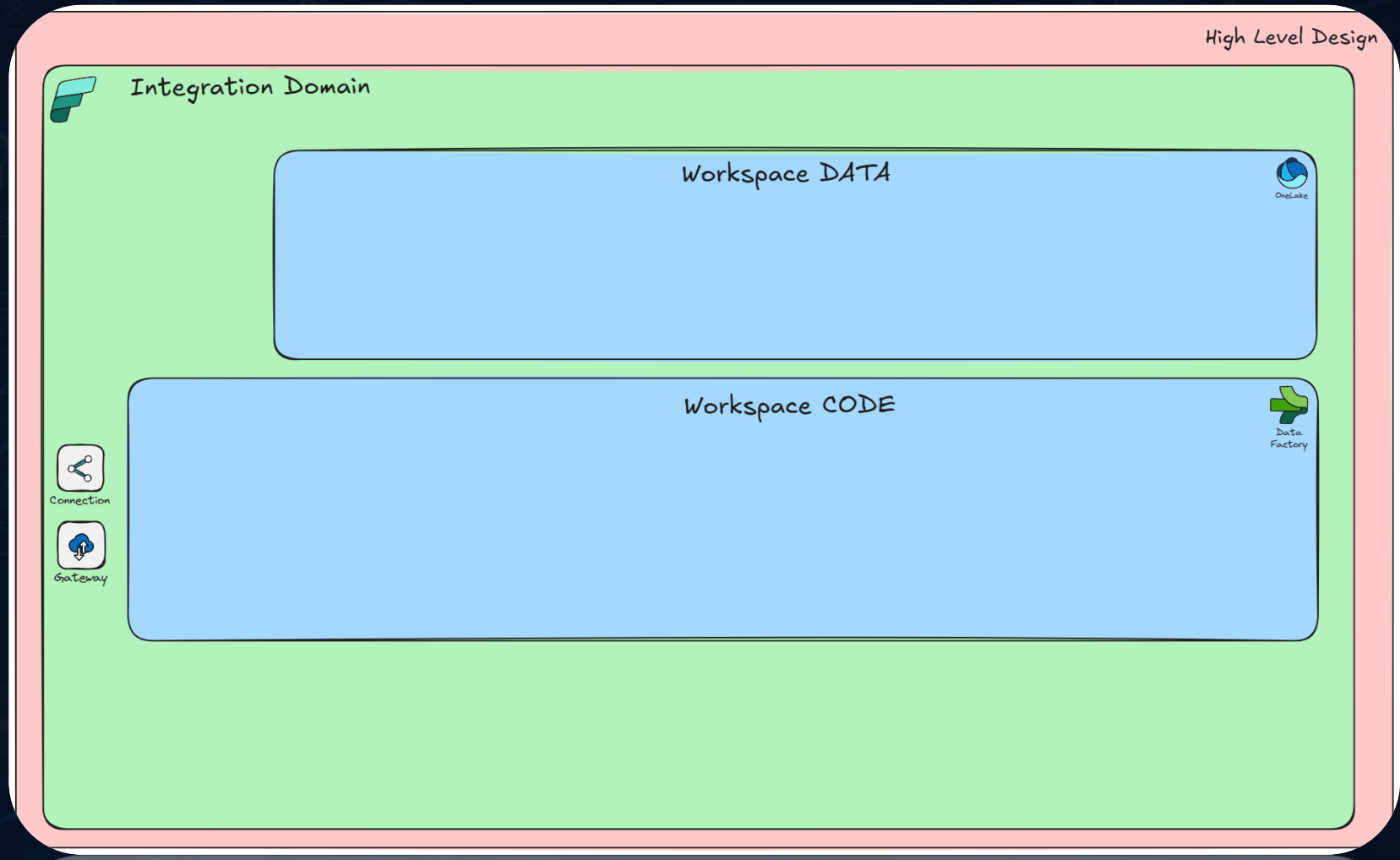
- Lakehouses



Data-Marc.com

Platform Engineering (Code Workspace)

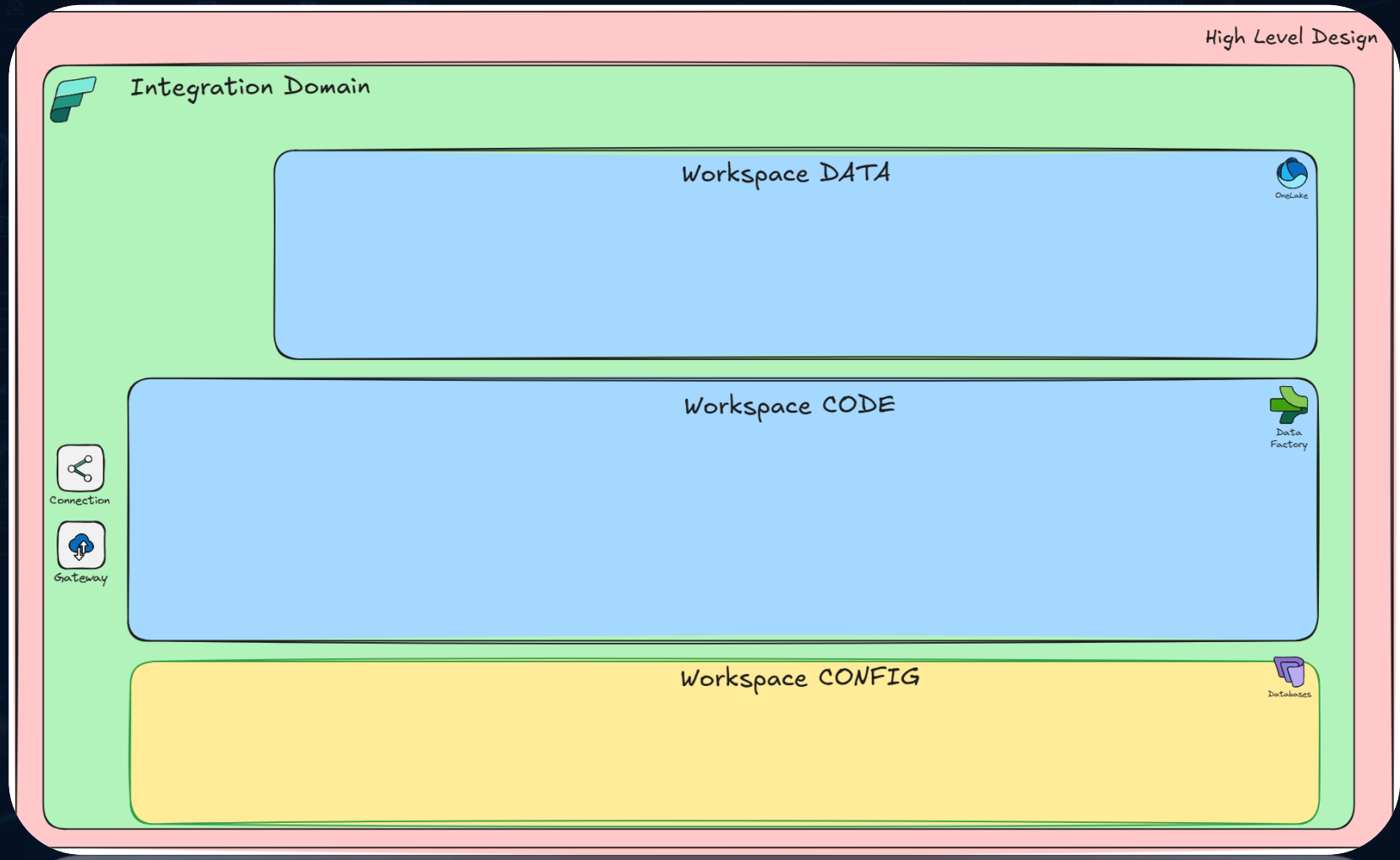
- Data Pipelines
- Notebooks
- Variable Libraries
- Environments



Data-Marc.com

Platform Engineering (Control Plane)

- Fabric database
  - Metadata database
- Environments

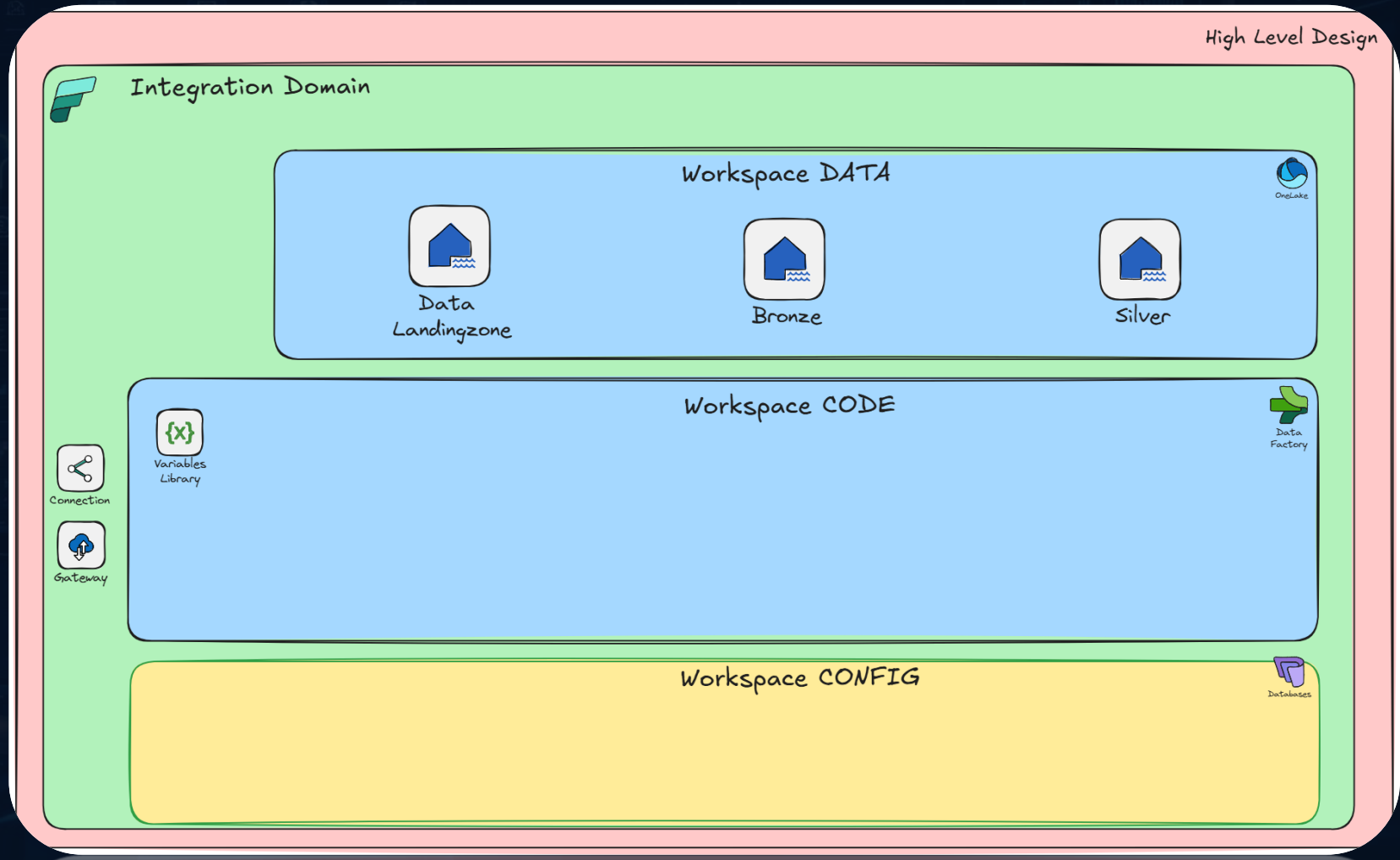


Data-Marc.com


A strong platform enables speed, reliability, and innovation at scale.
BUILDING THE KITCHEN
Data Platform Engineering Items
Data-Marc.com

Platform Engineering (Data Plane)

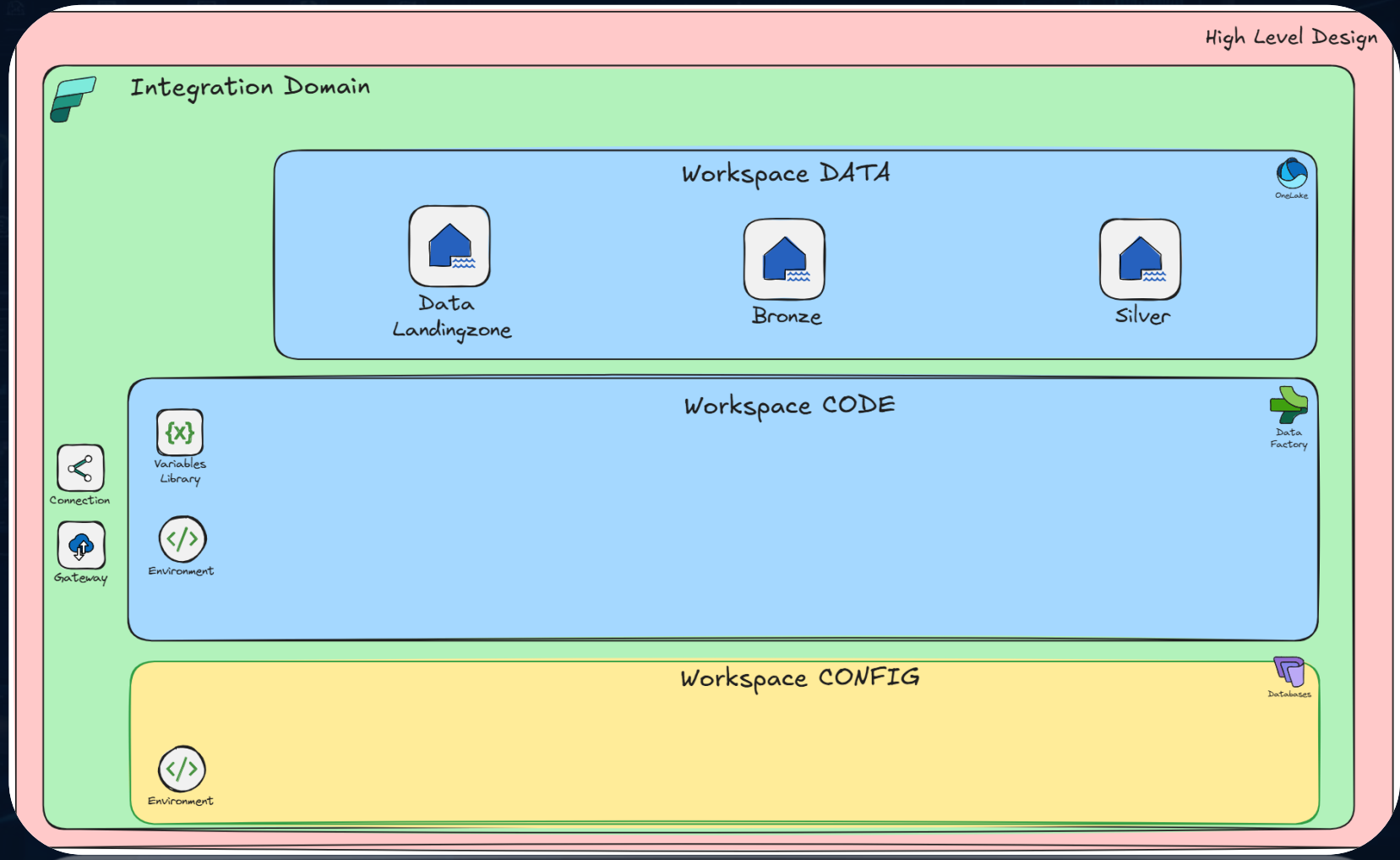
- Lakehouse Architecture
- Variable Library
  - Configuration settings



Data-Marc.com

Platform Engineering (Data Plane)

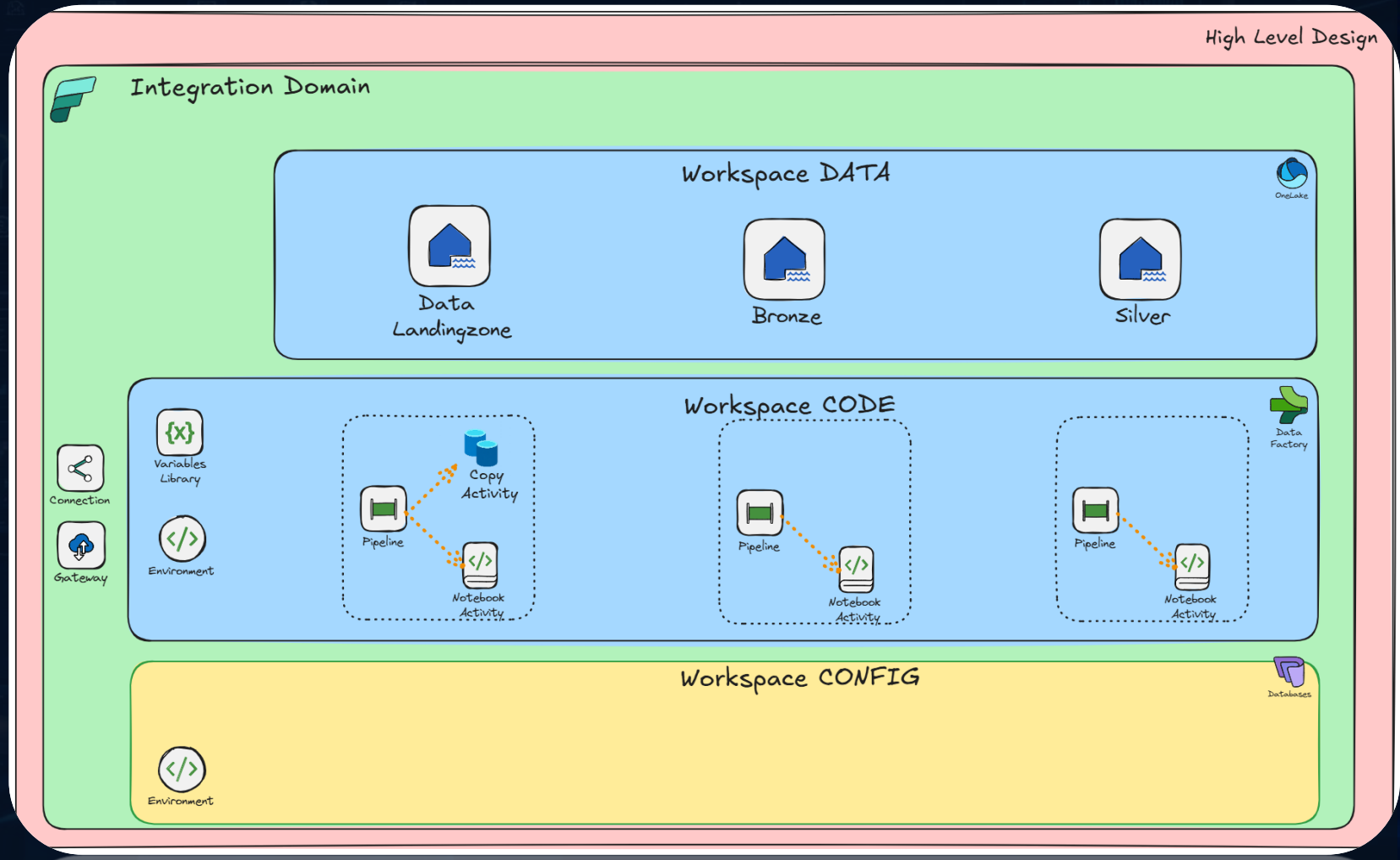
- Lakehouse Architecture
- Variable Library
- Environments
  - Spark settings



Data-Marc.com

Platform Engineering (Data Plane)

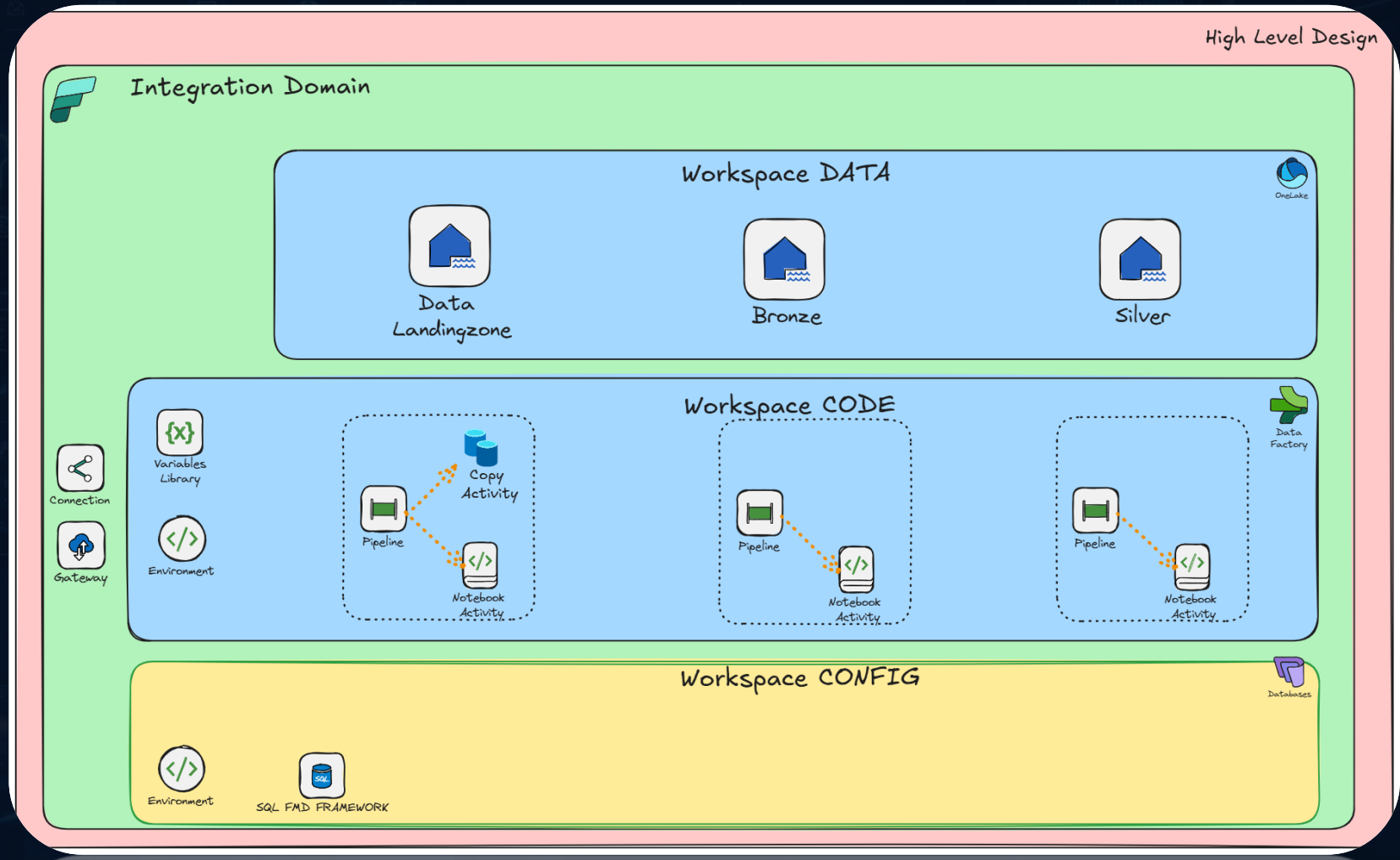
- Lakehouse Architecture
- Variable Library
- Environments
- Data Pipelines
- Notebooks



Data-Marc.com

Platform Engineering (Control Plane)

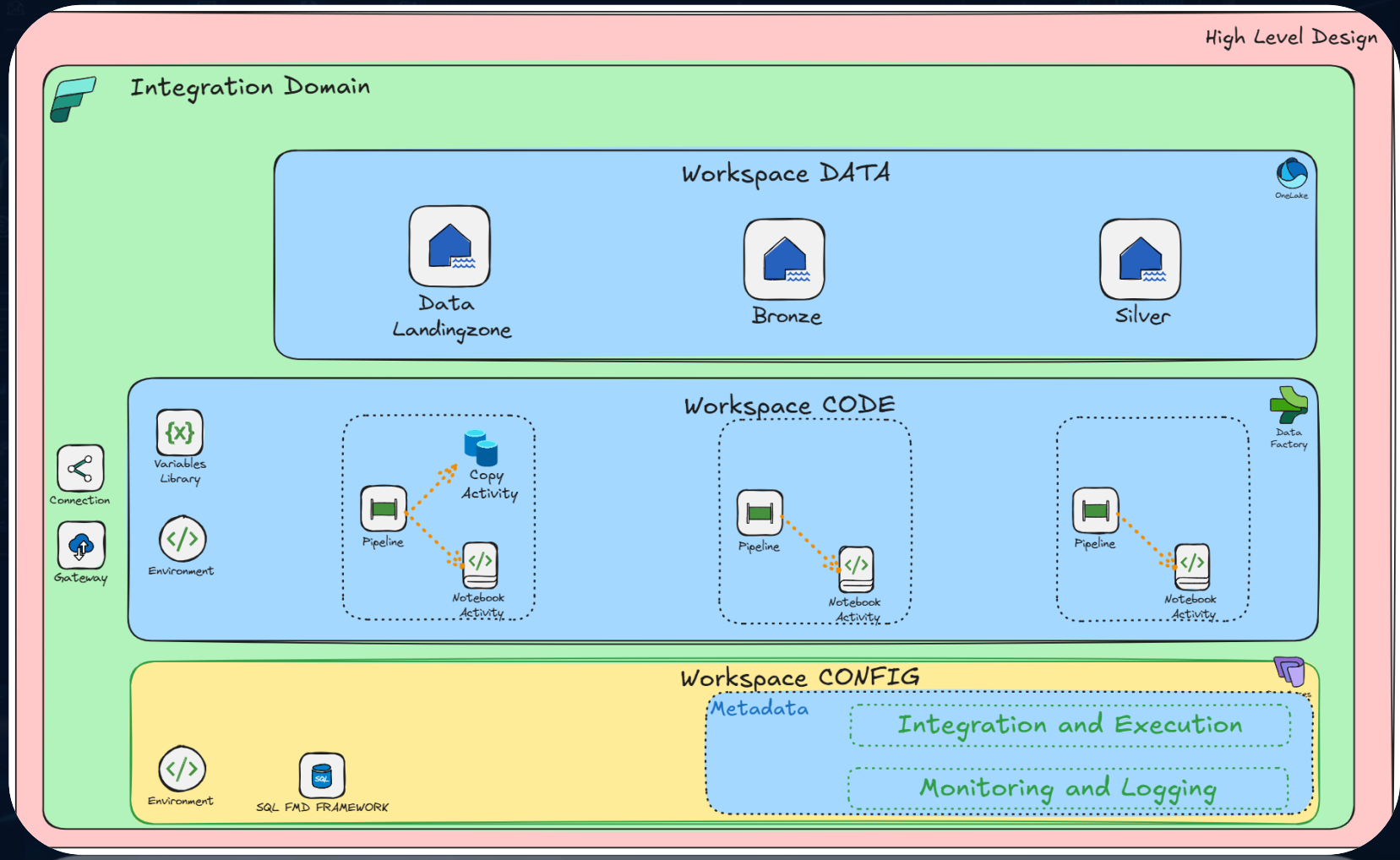
- SQL Database



Data-Marc.com

Platform Engineering (Control Plane)

- SQL Database
  - Metadata database



PARAMETERIZE

Data-Marc.com



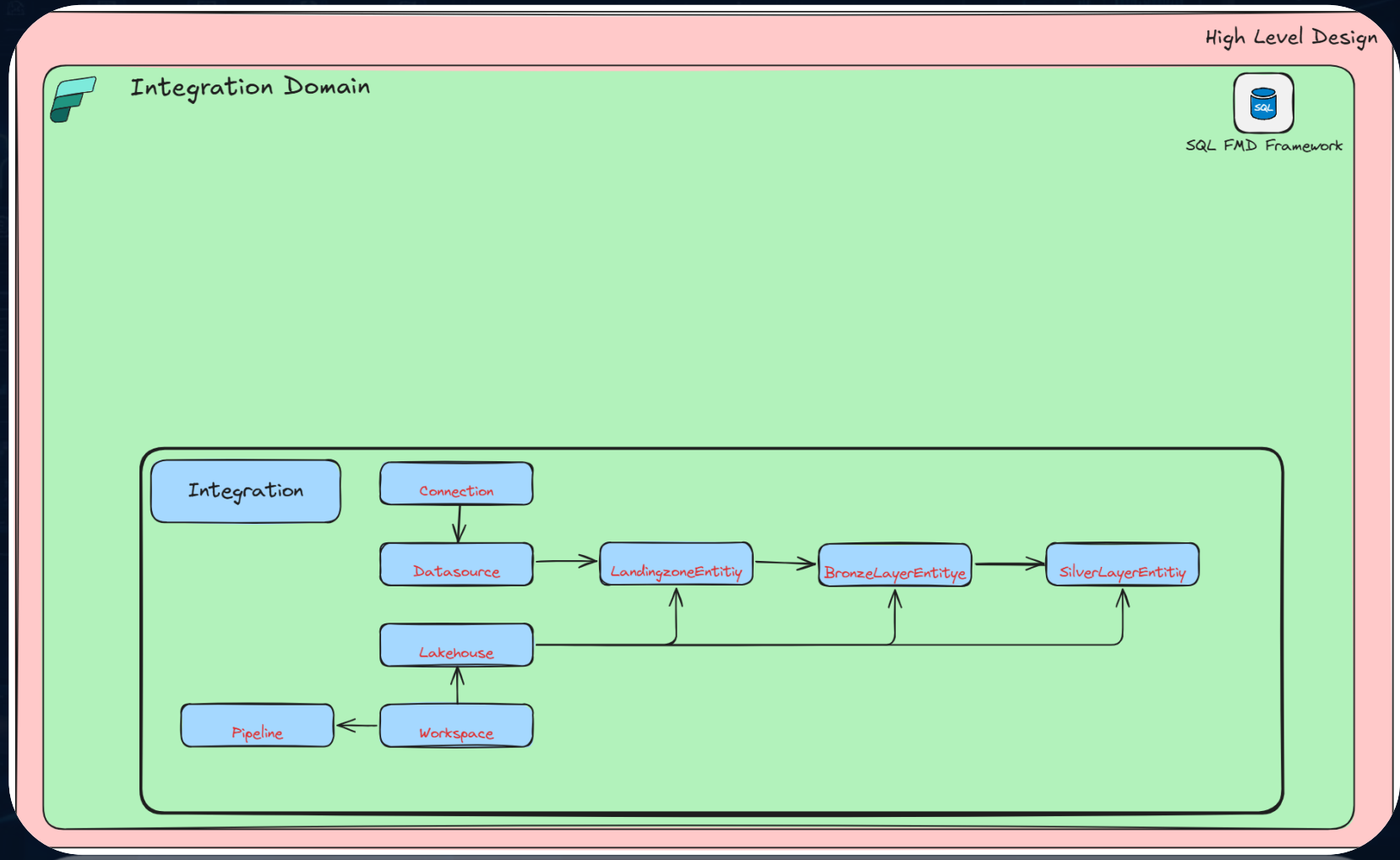
Unified data is created at ingestion, not at the end.

PREPARING AND COMBINING INGREDIENTS

Data Integration

Data Integration (Control Plane)

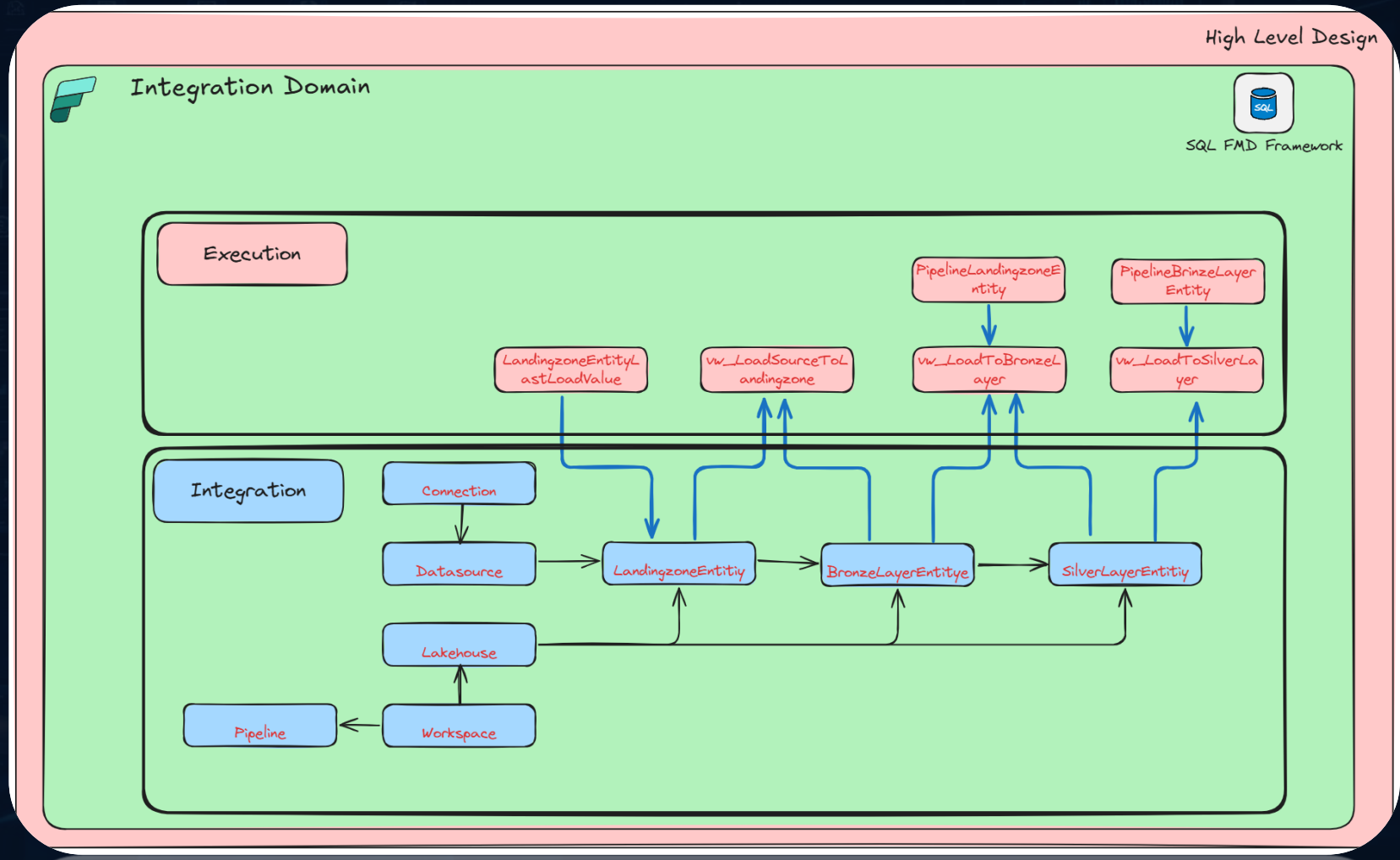
- Integration
  - Tables
  - Views
  - Stored Procedures



Data-Marc.com

Data Integration (Control Plane)

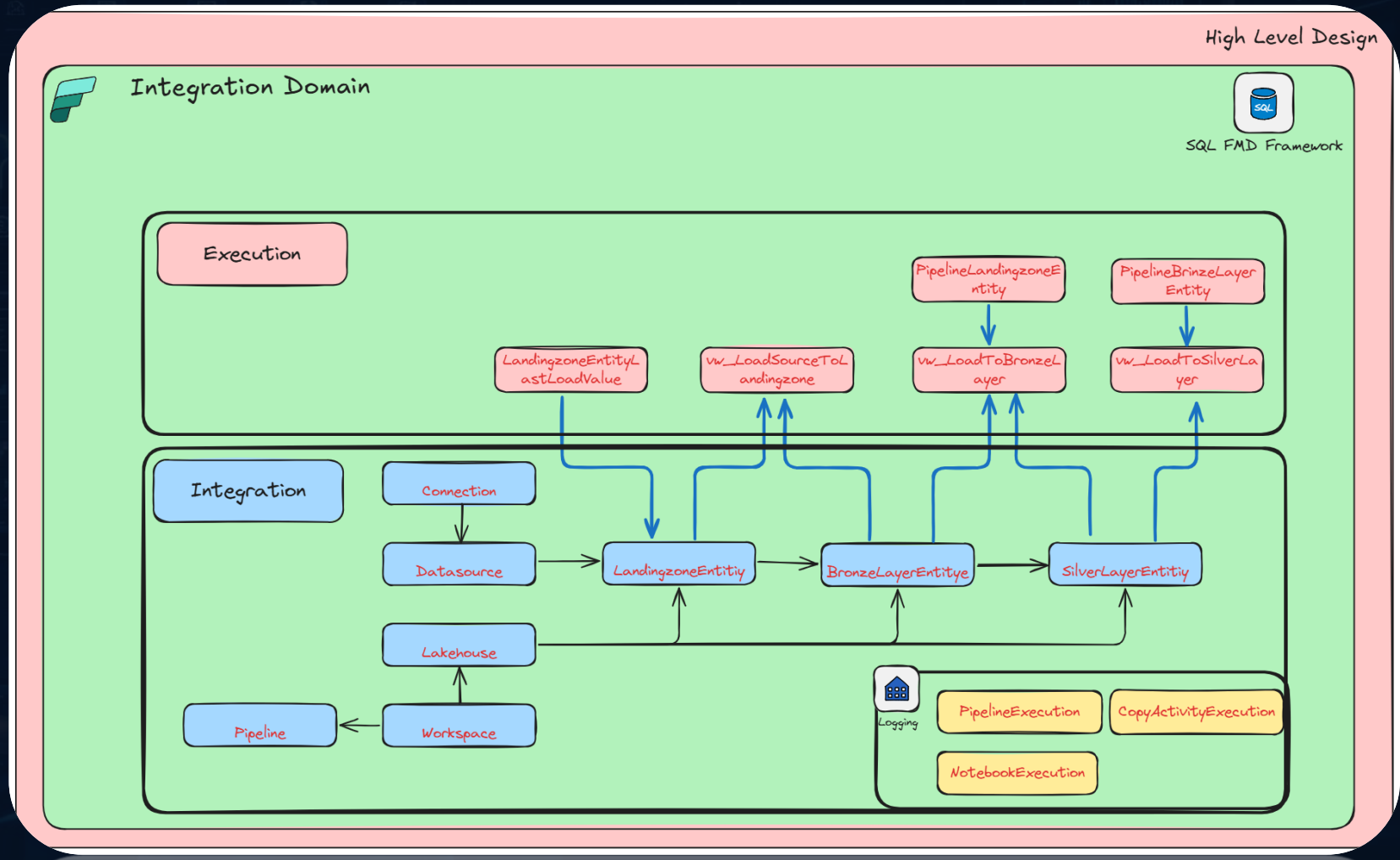
- Execution
  - Tables
  - Views
  - Stored Procedures



Data-Marc.com

Data Integration (Control Plane)

- Logging
  - Tables
  - Views
  - Stored Procedures



Data-Marc.com

The control plane is an executable contract

Metadata is not documentation. It decides what runs and how.

Entity	What to process	SalesOrder
Source	Where and how to read	Azure SQL / table
Target	Where to land	Bronze.SalesOrder
Load strategy	How to detect change	Watermark
Cleansing policy	What must be true	Not null + uniqueness
Observability	What to record	Counts + duration + state

Data-Marc.com



COOKING THE DISH

Data Processing

AUTOMATE

Data-Marc.com

Pipeline or notebook? Use both deliberately

Pipelines: orchestrate

- Connections and movement
- Activity dependencies
- Retries and timeouts
- Parameter passing
- Operational scheduling

Notebooks: compute

- Complex transformations
- Reusable libraries
- Data profiling and rules
- Delta operations
- Developer-testable logic

Data-Marc.com

One orchestrator, many entities



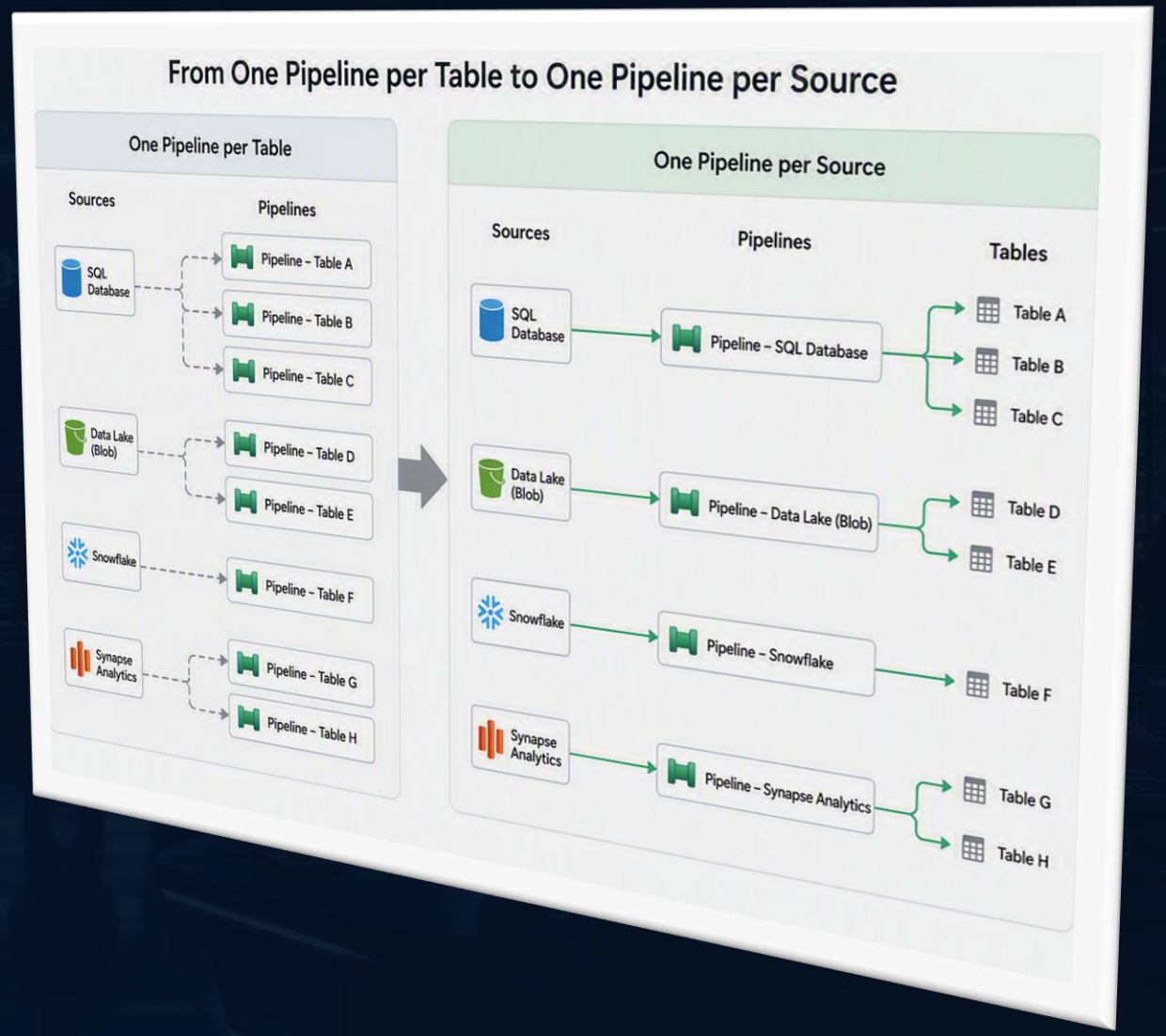
Data-Marc.com

One orchestrator, many entities

foreach
entity

resolve parameters
→ invoke pipeline or notebook
→ capture metrics
→ decide continue / quarantine / stop

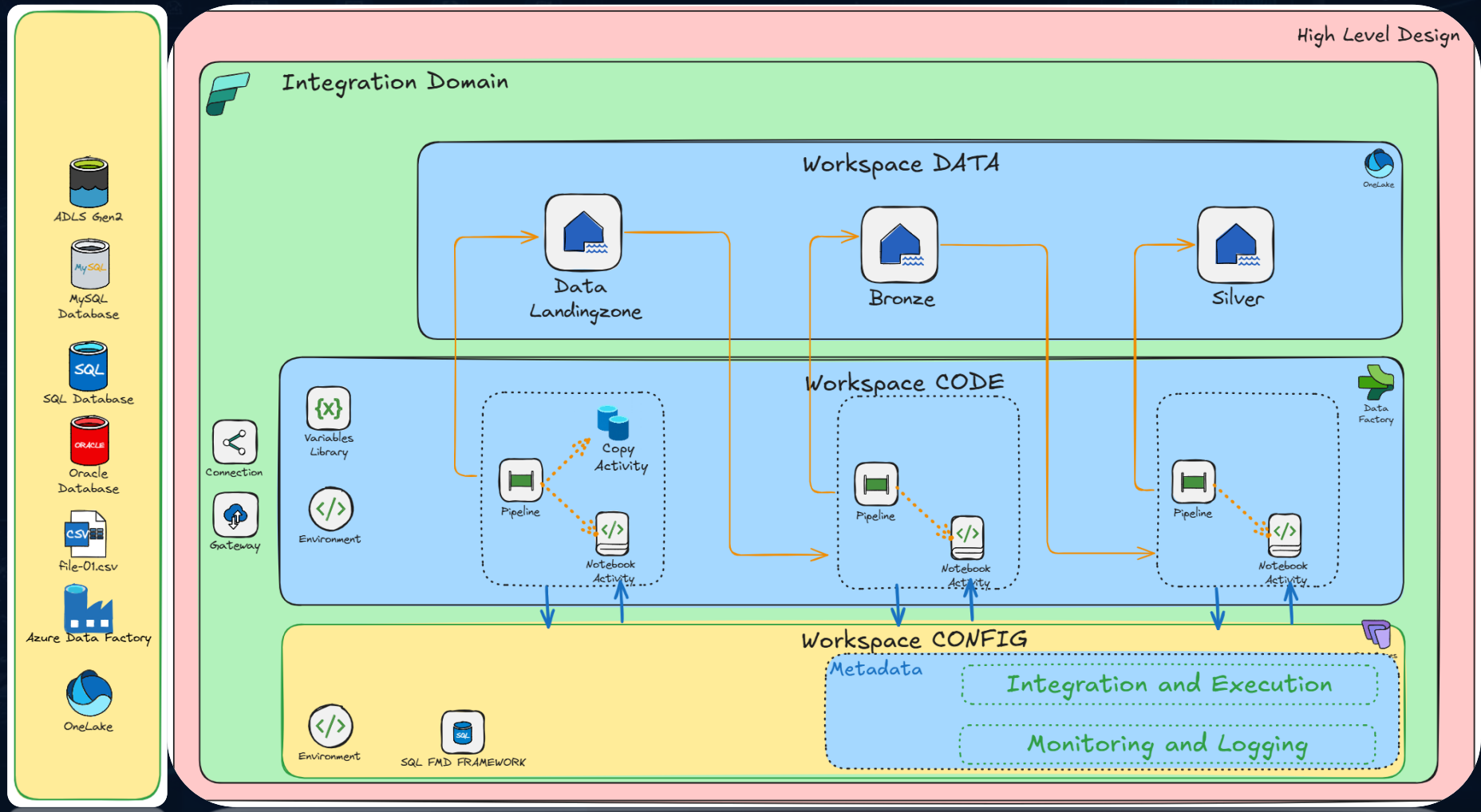
Idempotent execution + explicit run identity + bounded parallelism



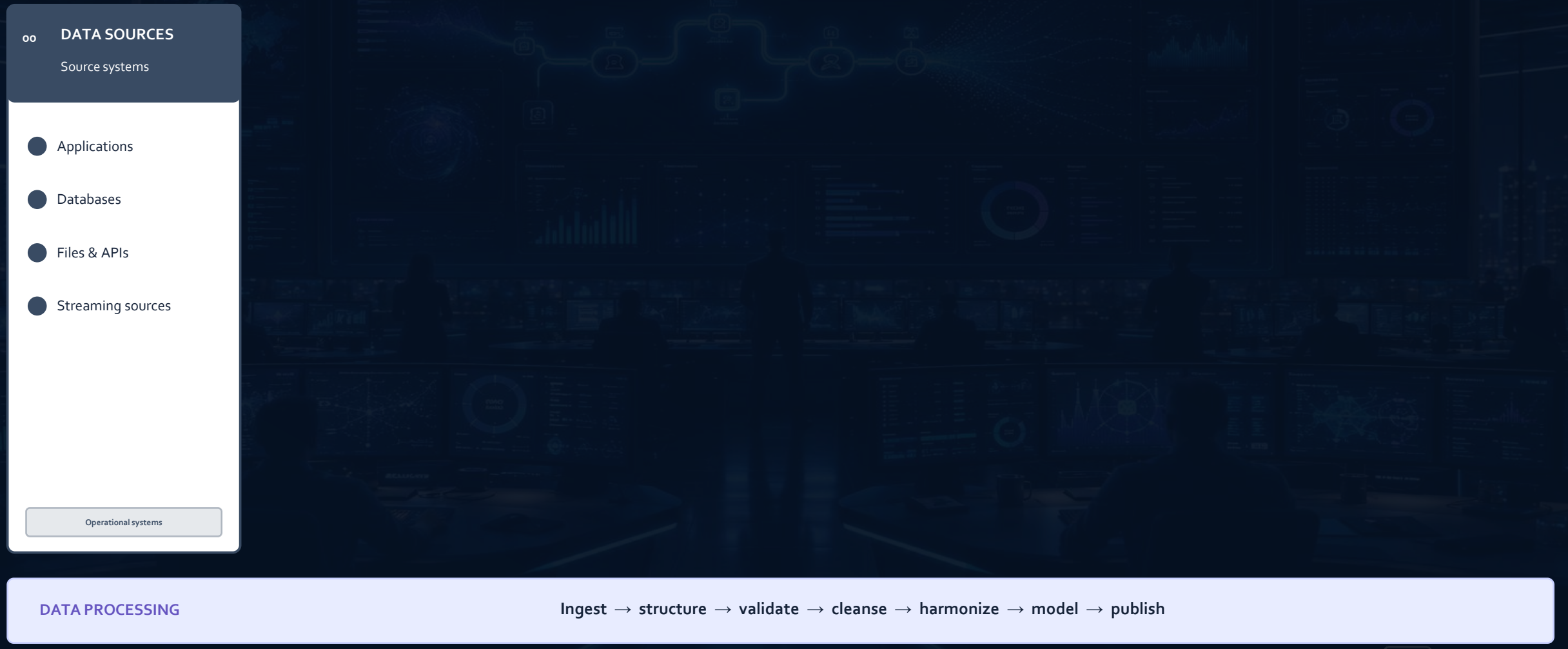
Data-Marc.com

Data Processing

- Data Sources



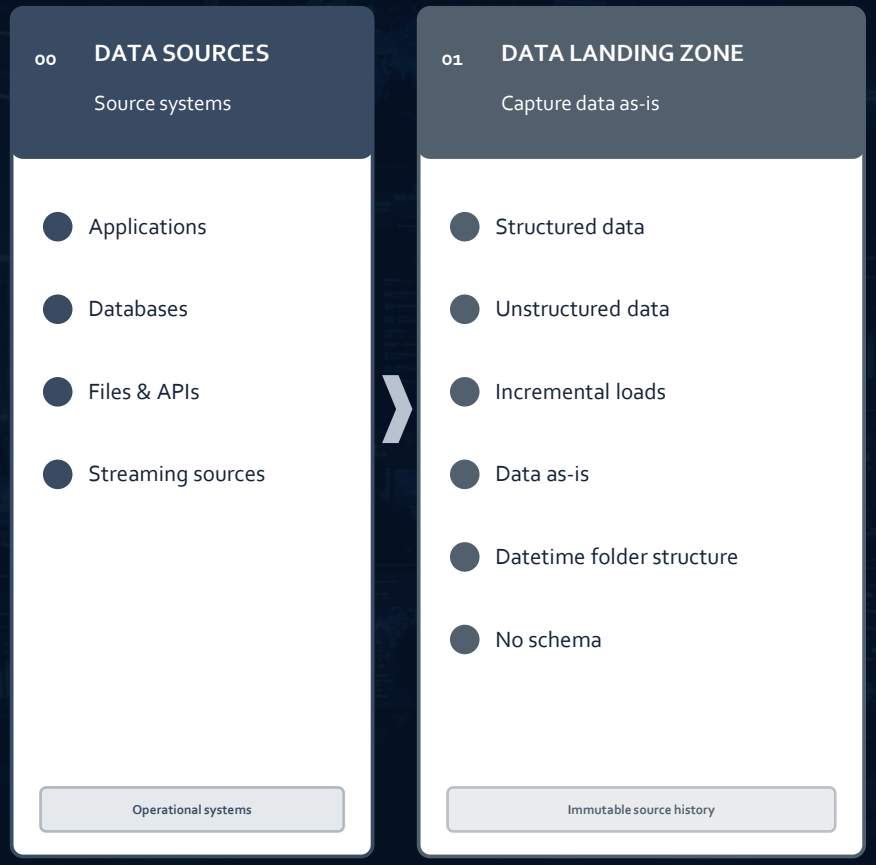
Data Serving



Data-Marc.com

Landing the Data

From Chaos to Control



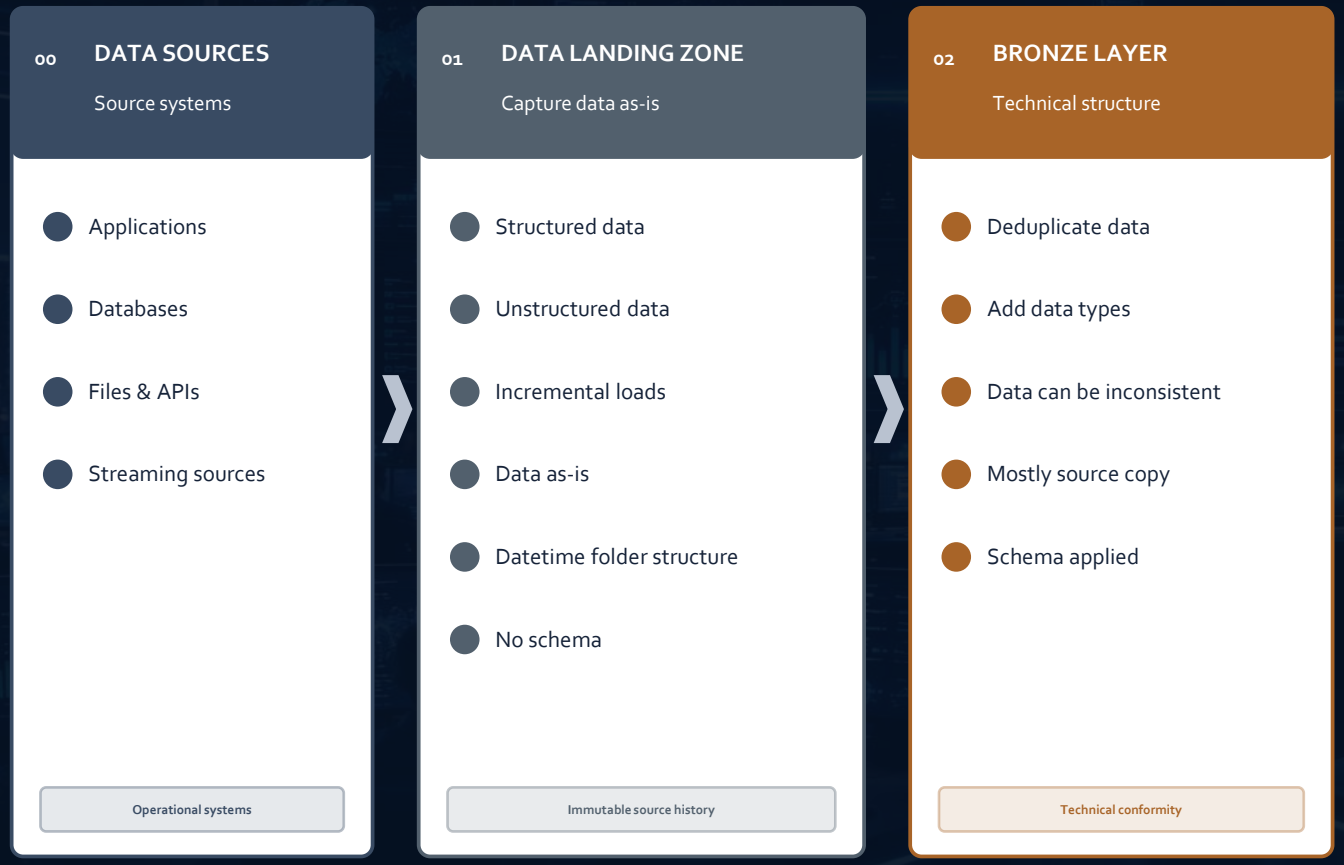
DATA PROCESSING

Don't lose data. Don't change data. Keep the evidence.

Data-Marc.com

Bronze Layer – Preserving the Truth

From Chaos to Control



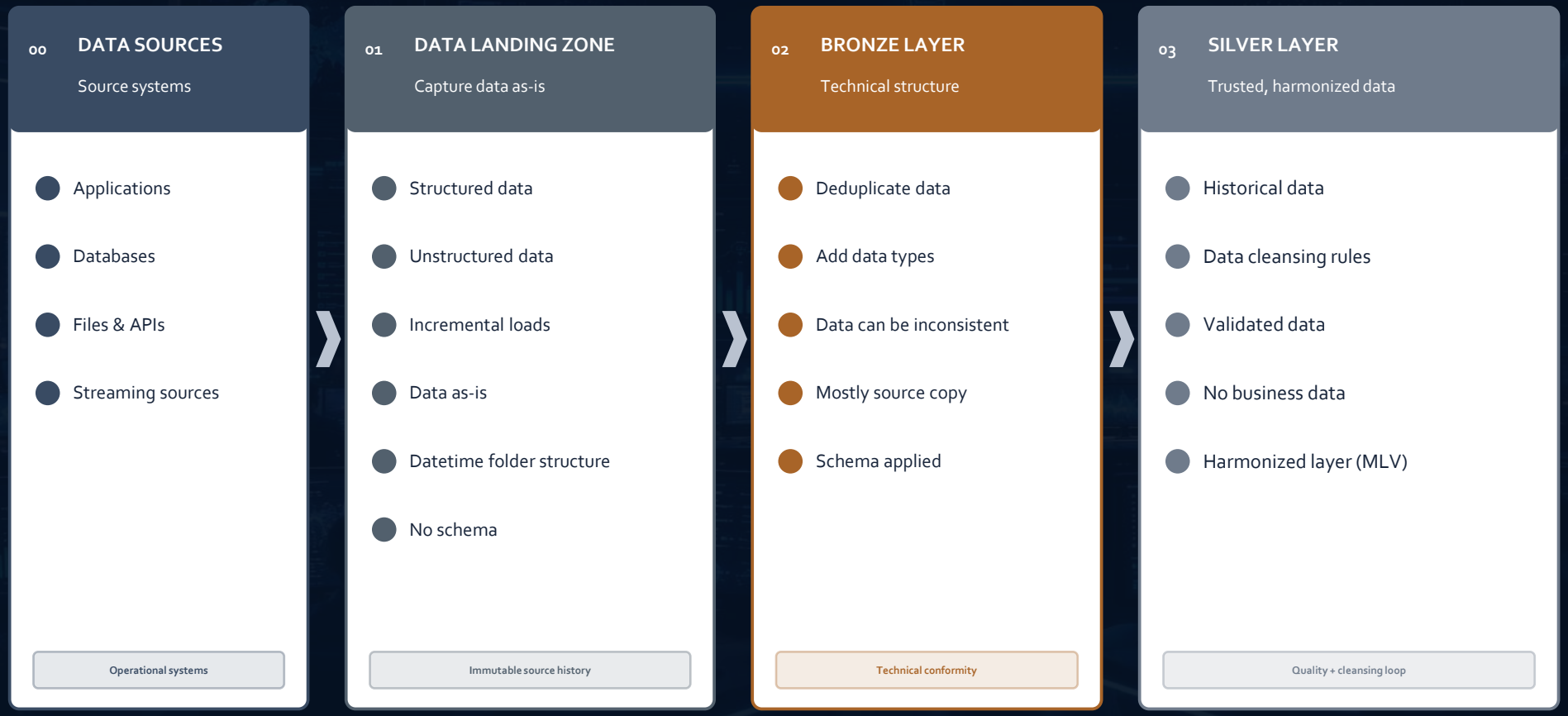
DATA PROCESSING

What exactly came from the source system?

Data-Marc.com

Silver Layer – Creating Business Context

Turning Data into Information



- This is where AI gains understanding.
- Not because of GenAI.
- Because the data finally makes business sense.

DATA PROCESSING

But information is not enough. Business users still need answers.

Silver is the trust boundary: validation identifies violations, cleansing resolves them, and re-validation proves the result before business logic is introduced.

Data-Marc.com



TASTING THE DISH

Data Quality

Data Quality

- Completeness
- Uniqueness
- Validity
- Consistency
- Timeliness
- Accuracy

01 **Completeness**
Required fields are populated
CustomerID is not null

02 **Uniqueness**
Keys and records are unique
No duplicate OrderID

03 **Validity**
Values conform to domain rules
Status in allowed values

04 **Consistency**
Formats and codes agree
Country uses ISO code

05 **Timeliness**
Data meets freshness SLA
Loaded within 24 hours

06 **Accuracy**
Values match trusted references
ProductID exists in master data

Data-Marc.com



Wash it, cut it, and prepare it for cooking.

Data Cleansing

Data-Marc.com

Data Cleansing

- Remove duplicates
- Fix invalid values
- Handle missing values
- Standardize formats
- Normalize codes and labels

STRUCTURE

Schema & type conformity

Required columns • data types • length • parseability

COMPLETENESS

Mandatory values

Null policy • defaults • conditional required fields

CONSISTENCY

Formats & normalization

Trim spaces • case • dates • units • code mapping

INTEGRITY

Keys & relationships

Uniqueness • duplicate detection • reference matching

VALIDITY

Domain & business rules

Allowed values • ranges • thresholds • cross-field logic

EXCEPTIONS

Controlled failure handling

Reject • quarantine • warn • ignore with owner and reason

Data-Marc.com

Data Quality Validation Rules vs Data Cleansing Rules

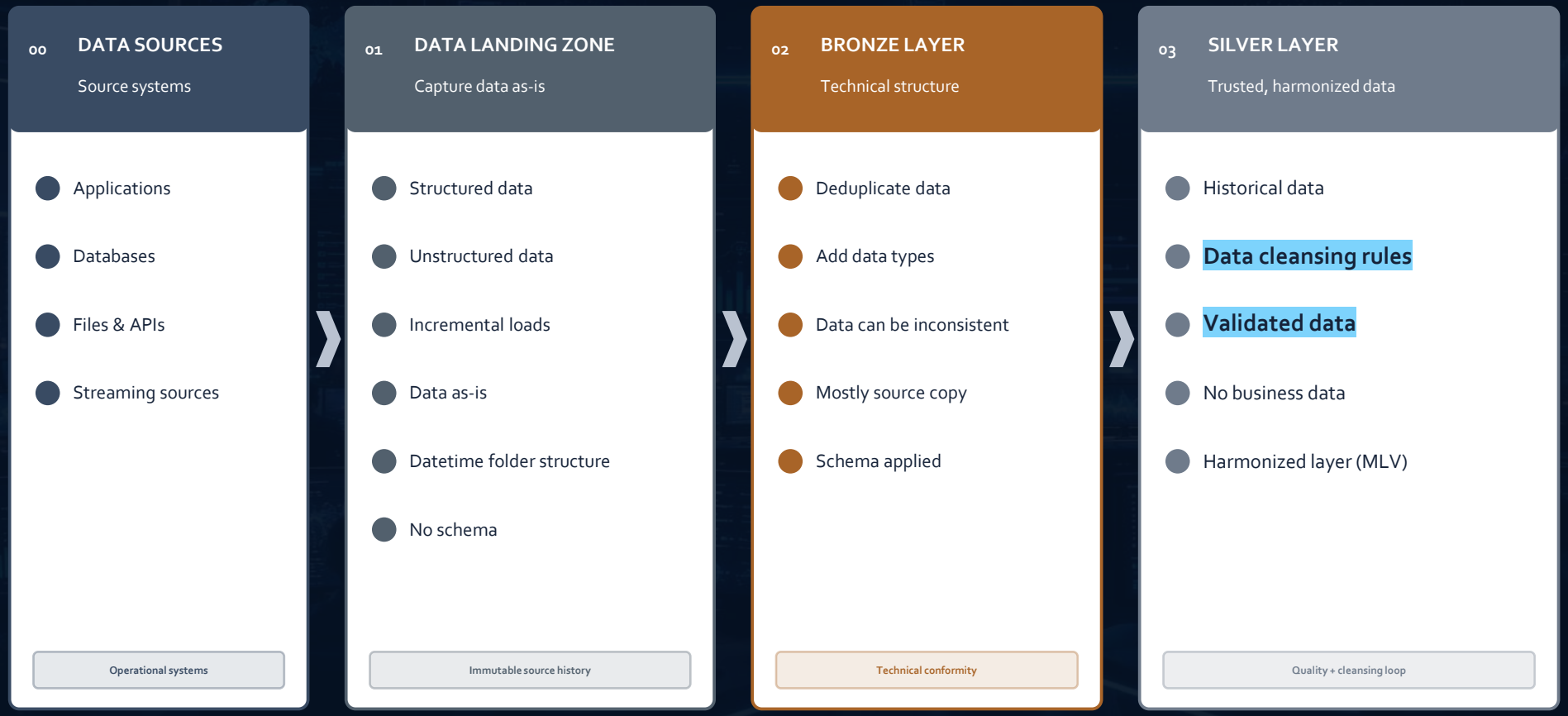
- Validation tells you WHAT is wrong.
- Cleansing determines HOW it gets fixed.

- Validation measures quality
- Cleansing improves quality

Aspect	Data Quality Validation Rules	Data Cleansing Rules
Purpose	Determine whether data is fit for use	Correct data that is not fit for use
Question	*Is the data correct?*	*How do we fix the data?*
Outcome	Pass, Warn, Fail, Drop, Quarantine	Transform, Standardize, Enrich, Repair
When	Before or during processing	During processing
Focus	Detection	Correction
Result	Quality score and rule violations	Improved data quality

Data-Marc.com

Data Processing



DATA PROCESSING — Ingest → structure → validate → cleanse → harmonize → model → publish

Silver is the trust boundary: validation identifies violations, cleansing resolves them, and re-validation proves the result before business logic is introduced.

Data-Marc.com

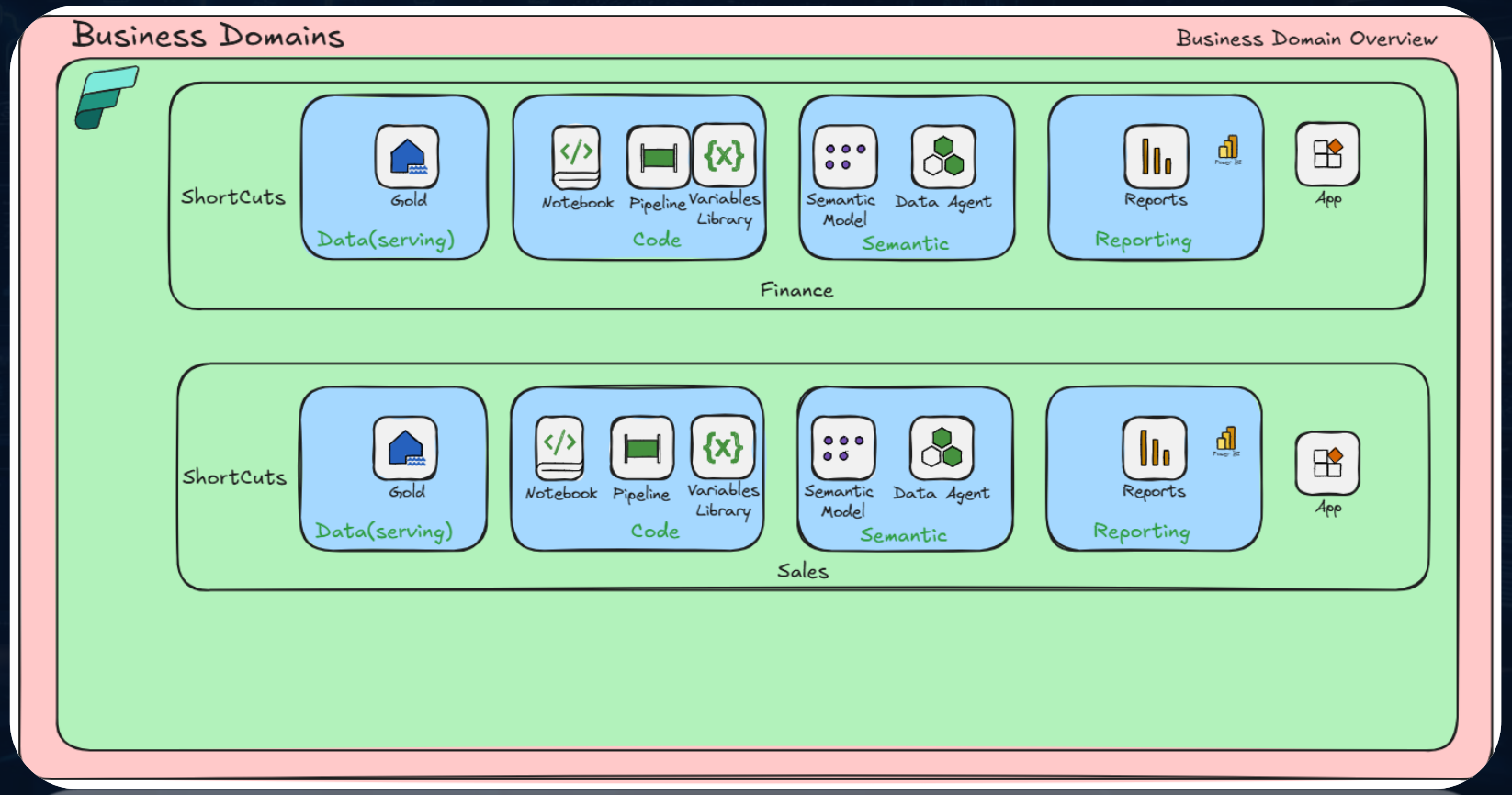


Serve the dishes

Data Serving

Data Serving

- Create serving layer per domain
- Separation of duties
- Governance
- Security



Data-Marc.com

Gold Layer – Creating Business Knowledge

Speaking the Language of the Business

00 DATA SOURCES
Source systems

- Applications
- Databases
- Files & APIs
- Streaming sources

Operational systems

01 DATA LANDING ZONE
Capture data as-is

- Structured data
- Unstructured data
- Incremental loads
- Data as-is
- Datetime folder structure
- No schema

Immutable source history

02 BRONZE LAYER
Technical structure

- Deduplicate data
- Add data types
- Data can be inconsistent
- Mostly source copy
- Schema applied

Technical conformity

03 SILVER LAYER
Trusted, harmonized data

- Historical data
- Data cleansing rules
- Validated data
- No business data
- Harmonized layer (MLV)

Quality + cleansing loop

04 GOLD LAYER
Business-ready data

- Facts and dimensions
- Materialized Lake Views
- Business friendly names
- Calculations and aggregations
- Agent Consume Ready

Analytics and AI

DATA PROCESSING

Gold translates technical data into business meaning.

Silver is the trust boundary: validation identifies violations, cleansing resolves them, and re-validation proves the result before business logic is introduced.

Data-Marc.com



MONITORING AND QUALITY CONTROL

Data Observability

"Who fully trusts every number in every dashboard?"

Data-Marc.com

The answer is correct.

A Data Agent reports:

"Revenue yesterday was" **€4.2M**

✓ A confident, plausible answer

Everyone is happy.

The number looks plausible. The answer sounds certain.



VISIBLE: a successful answer

Everything looked green.

Data-Marc.com

The answer was correct. The data was not.

A Data Agent reports:

"Revenue yesterday was" **€4.2M**

✓ A confident, plausible answer

Everyone is happy.

Until the platform evidence appears...

01	**Pipeline failed**	The orchestration did not complete.
02	**Only 20% of orders loaded**	The reported revenue was incomplete.
03	**Customer data was stale**	The answer mixed current and outdated data.
04	**Wrong notebook parameter**	The execution context was incorrect.



VISIBLE: a successful answer

HIDDEN: broken execution context

Data-Marc.com

The answer was correct. The data was not.

A Data Agent reports:

"Revenue yesterday was" €4.2M

✓ A confident, plausible answer

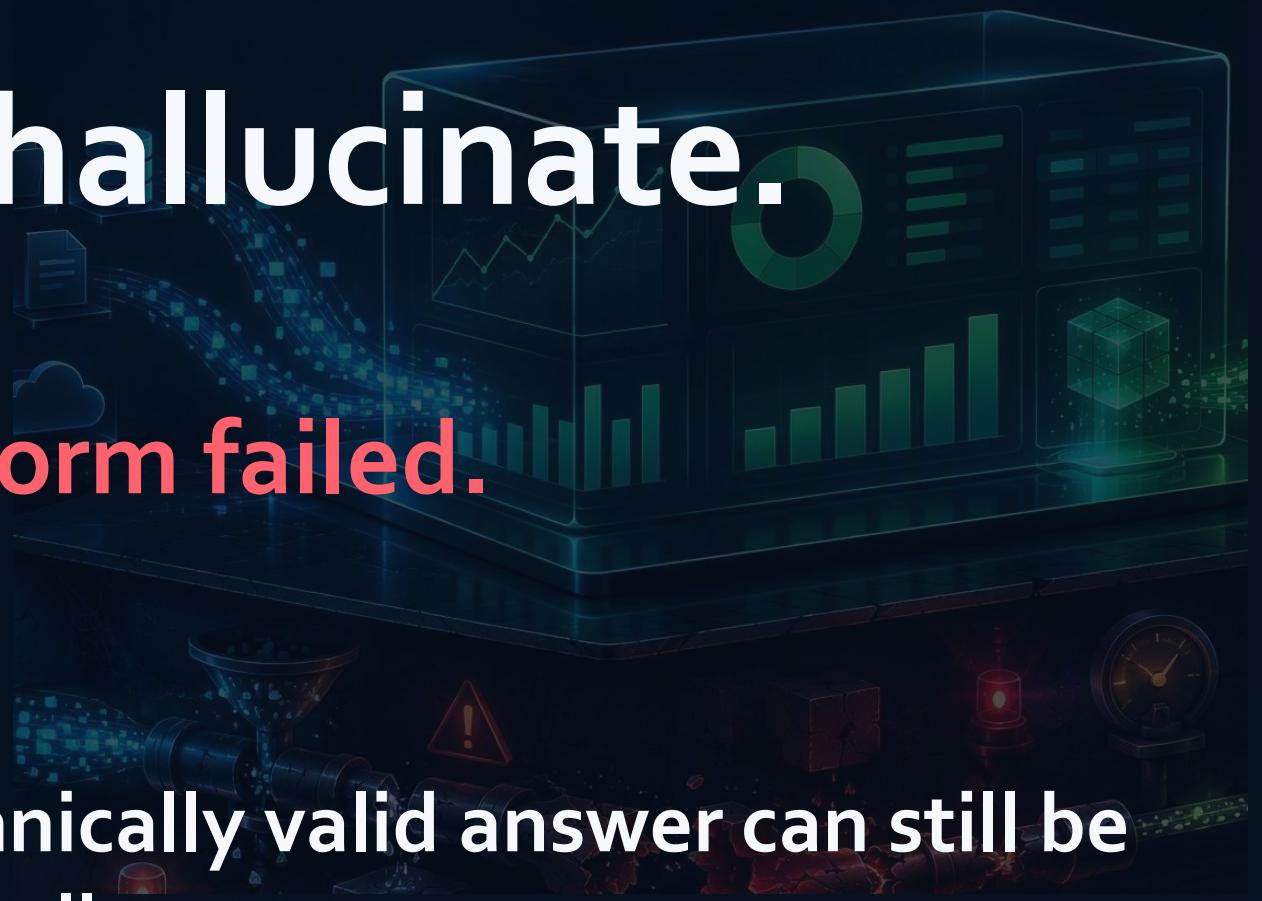
Everyone is happy.

Until the platform evidence appears...

01	Pipeline failed	The orchestration did not complete.
02	Only 20% of orders loaded	The reported revenue was incomplete.
03	Customer data was stale	The answer mixed current and outdated data.
04	Wrong notebook parameter	The execution context was incorrect.

AI did not hallucinate.

The platform failed.



Without observability, a technically valid answer can still be operationally wrong.

VISIBLE: a successful answer

HIDDEN: broken execution context

Data-Marc.com

Data Observability – Maintaining Trust at Scale

- Create serving layer per domain
- Separation of duties
- Governance
- Security

Data-Marc.com

Data Observability – Maintaining Trust at Scale

Who Watches the Data?



Data-Marc.com

The Six Observability Domains

TRUST EVIDENCE

Observability turns operational telemetry into evidence that data and AI answers can be trusted.

1 **START** — Run context

2 **EXECUTE** — Pipeline / notebook

3 **MEASURE** — Records & duration

4 **COMPLETE** — Success or failure

1 WHEN

Start & End Time

QUESTIONS

- Did it run?
- When?
- How long?

Without timing: no freshness, no SLA, no trust.

2 VOLUME

Extracted Records

QUESTIONS

- How much data arrived?
- Is volume normal?
- Are trends changing?

Volume anomalies often become business-answer anomalies.

3 HEALTH

Execution Failure

QUESTIONS

- What failed?
- Why?
- Where?

Go beyond the error message to root-cause analysis.

4 ORCHESTRATION

Pipeline Information

QUESTIONS

- Which pipeline?
- Which workspace?
- Which trigger or activity?

Without orchestration metadata, history cannot be reconstructed.

5 COMPUTE

Notebook Information

QUESTIONS

- Which notebook?
- Which version?
- Which parameter set?

Essential when Fabric notebooks drive transformations.

6 CONTEXT

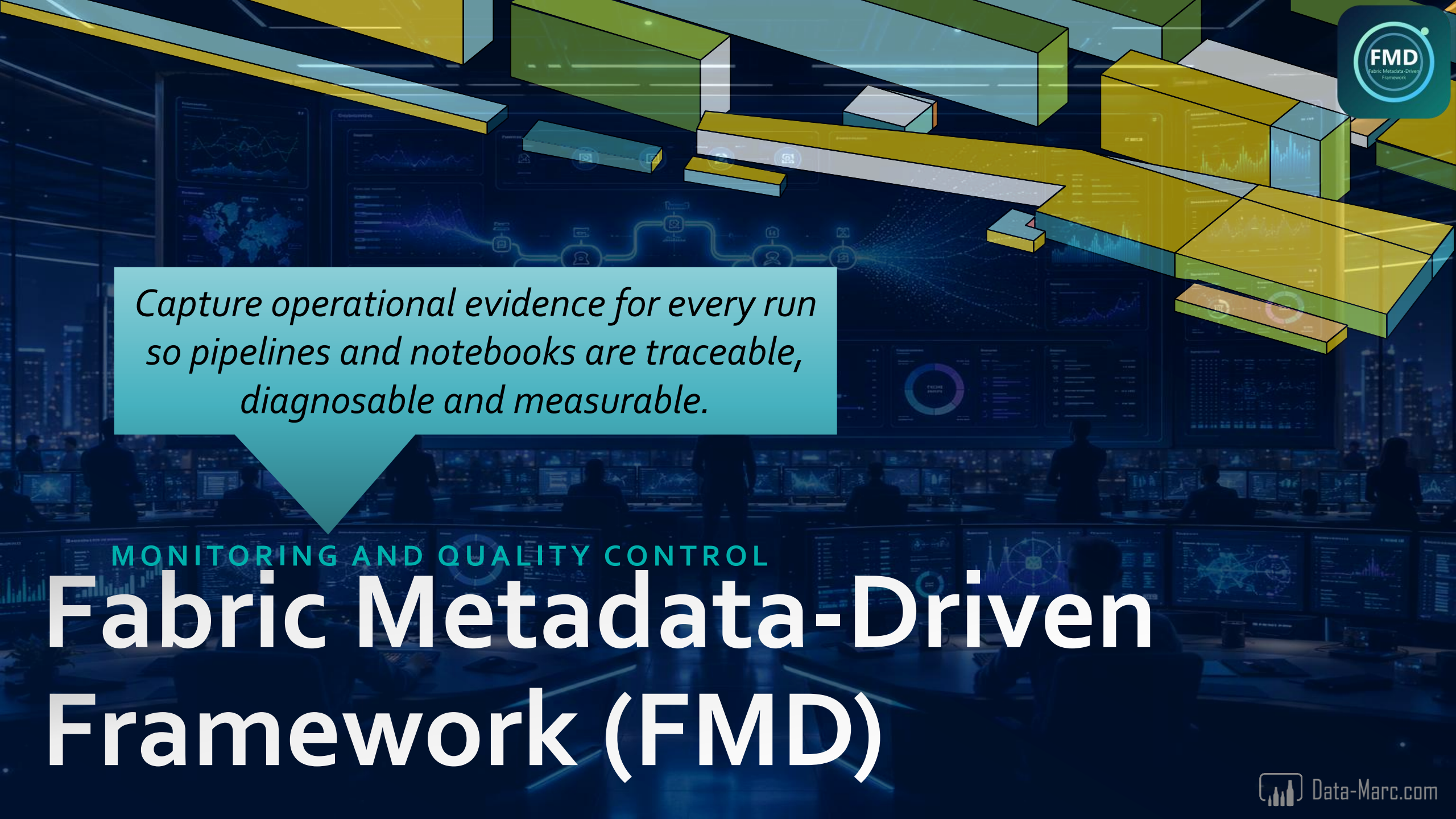
Runtime Variables

QUESTIONS

- Production or test?
- Incremental or full load?
- Source version or watermark?

Context explains execution behaviour.

ta-Marc.com



MONITORING AND QUALITY CONTROL

Fabric Metadata-Driven Framework (FMD)

Data-Marc.com

Fabric Metadata Driven Framework

Modern data platforms demand agility, scalability, and consistency. FMD simplifies these challenges by enabling:

- Dynamic, metadata-driven pipelines
- Consistent orchestration across ingestion, processing, and publishing
- Centralized configuration for all data entities
- Alignment with Microsoft Fabric Lakehouse & Medallion Architecture
- Reduced engineering effort through reusable patterns
- Faster delivery with standardized, tested components

Data-Marc.com

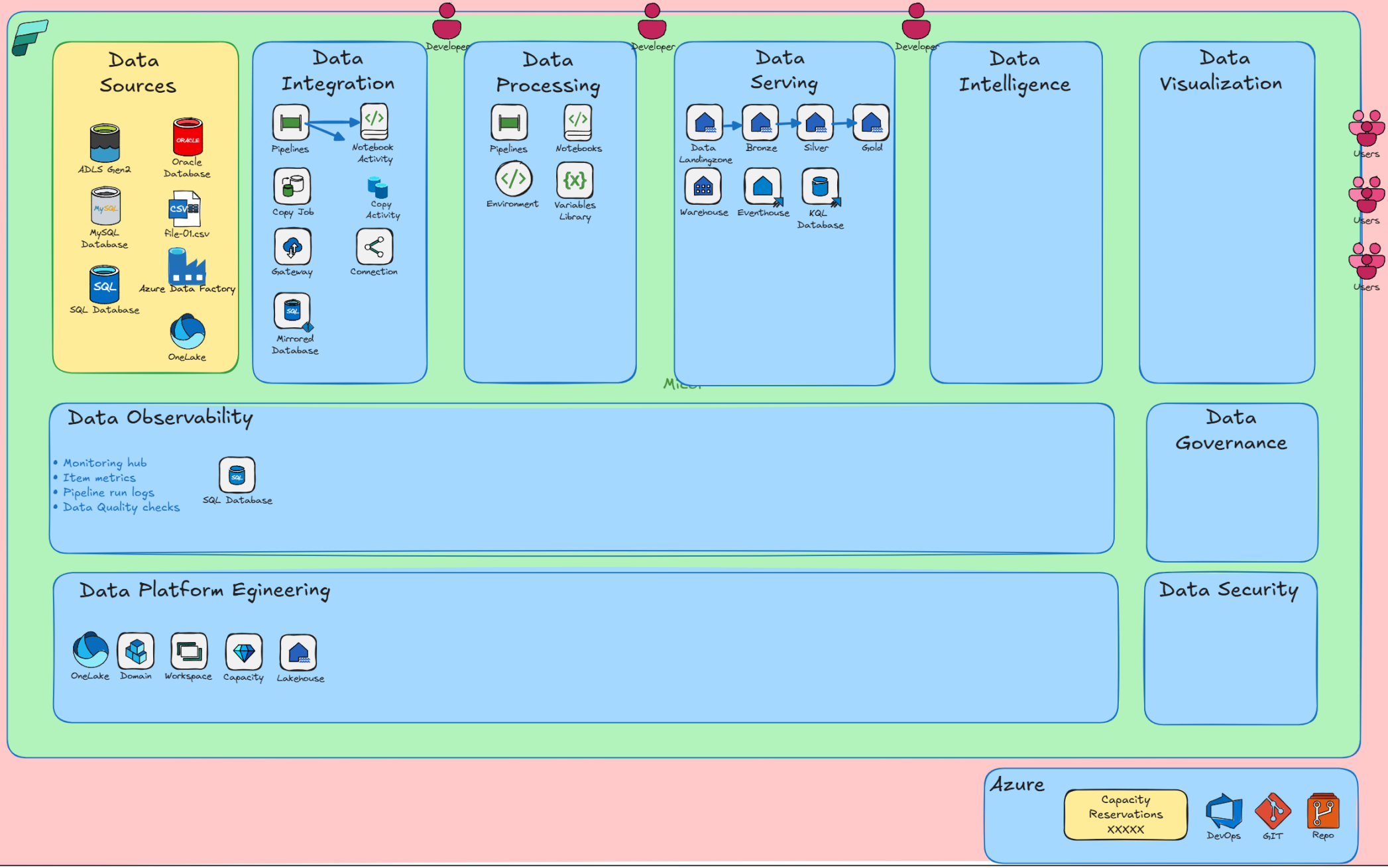
Fabric Metadata Driven Framework

- Meets most of your needs
- Full control over design and features.
- Less development effort.
- Flexibility and extensibility.
- Lower upfront costs.
- 100 % Fabric Native Solution

FMD Framework: Unlock Data Integration - Erwin | Data & Intelligence

Data-Marc.com

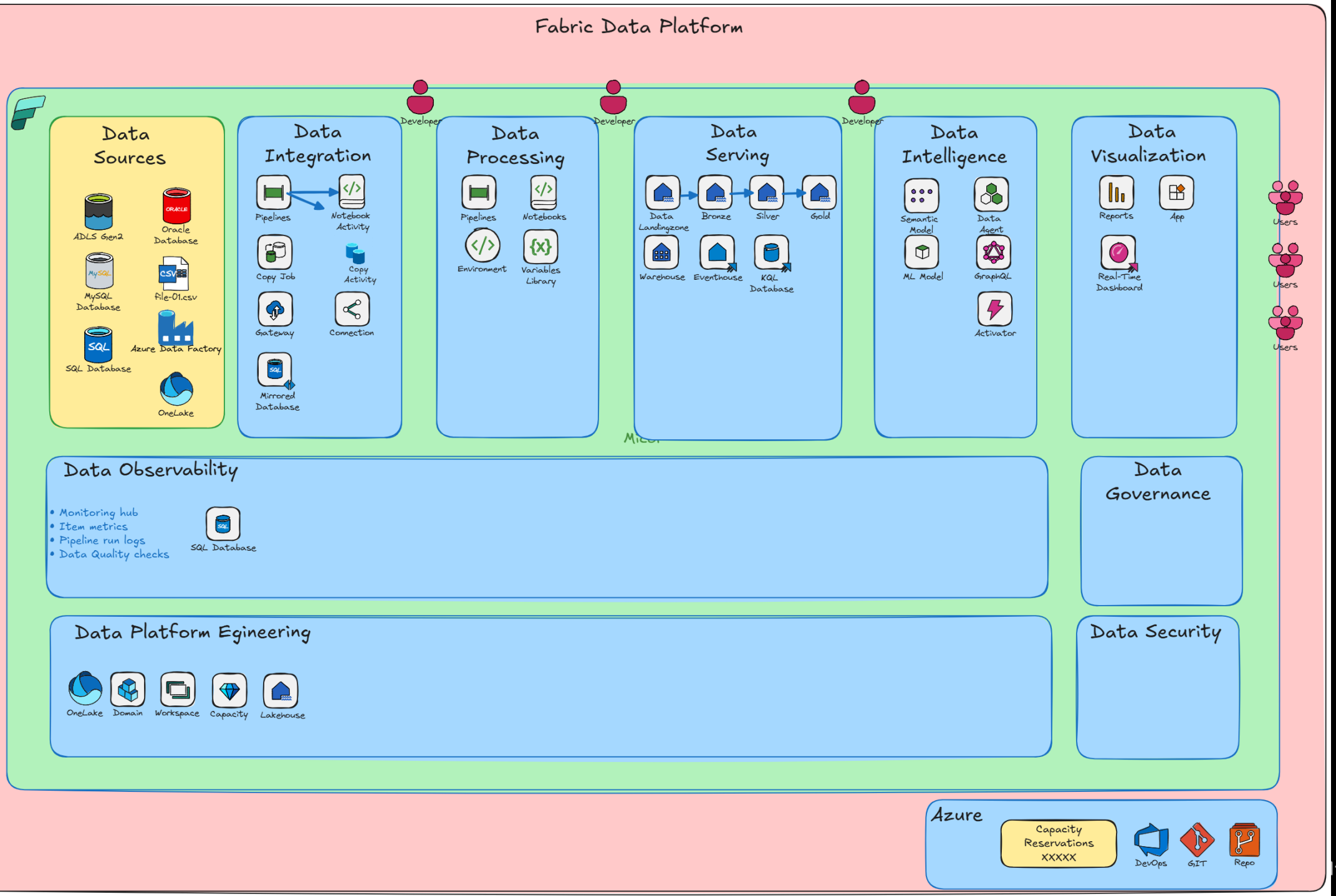
Fabric Data Platform



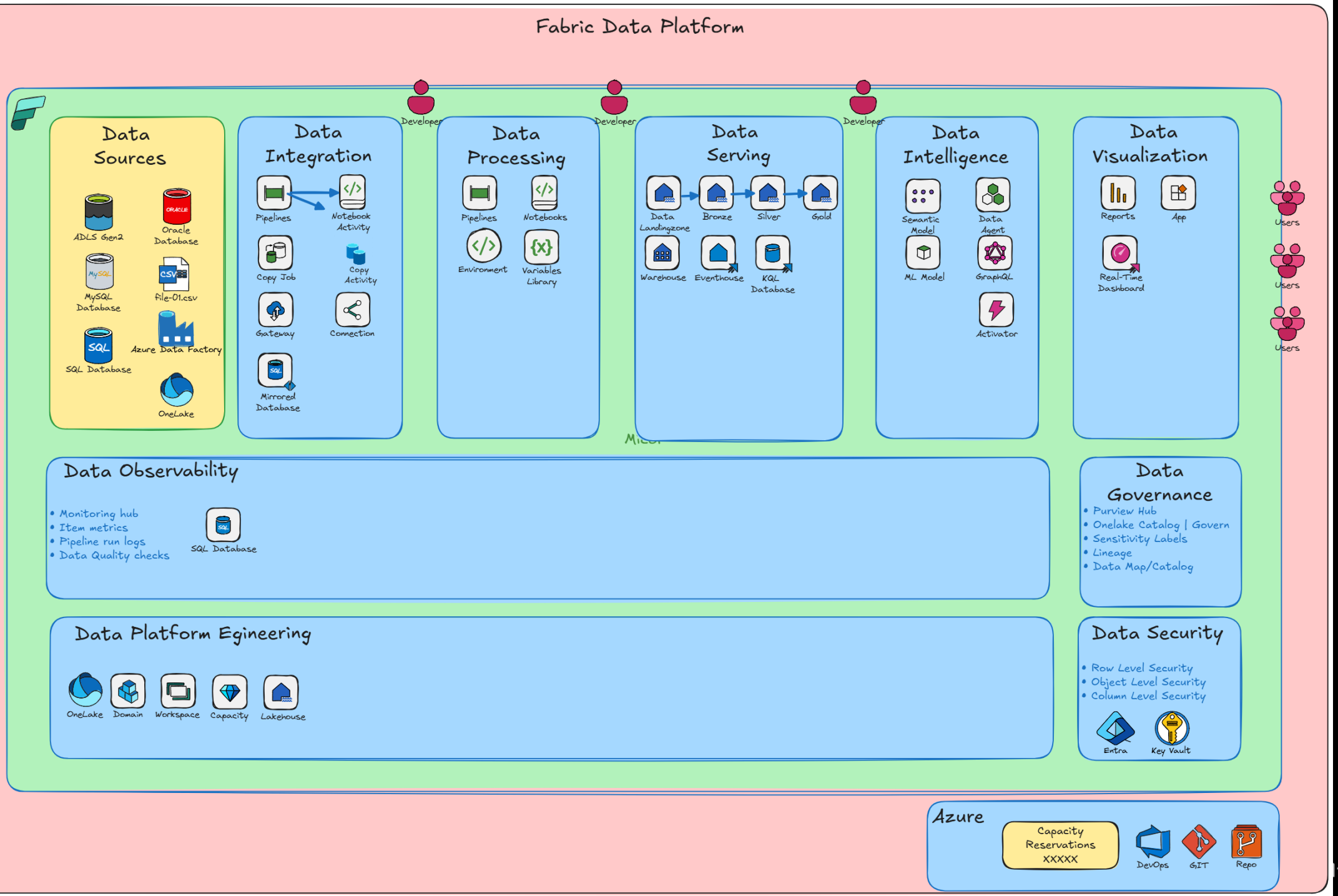
Break

SM/Agents+rest

Fabric Data Platform



Fabric Data Platform



-Marc.com

Every layer reduces uncertainty.
The agent is simply the final consumer
of a
well-designed data platform.

Data-Marc.com

Most organizations start with

How do we build an AI agent?

The better question is:
How do we build data that an AI agent can trust?

Reliable AI is not an AI problem.

It's a data platform problem.

From reliable platform to trusted AI answers



Source: Tim Hiebenthal - https://handsondata.substack.com/p/ontology-everywhere

Fabric IQ

People

Agents

Operational Intelligence

Ontologies

IQ

Business Intelligence

Semantic Models

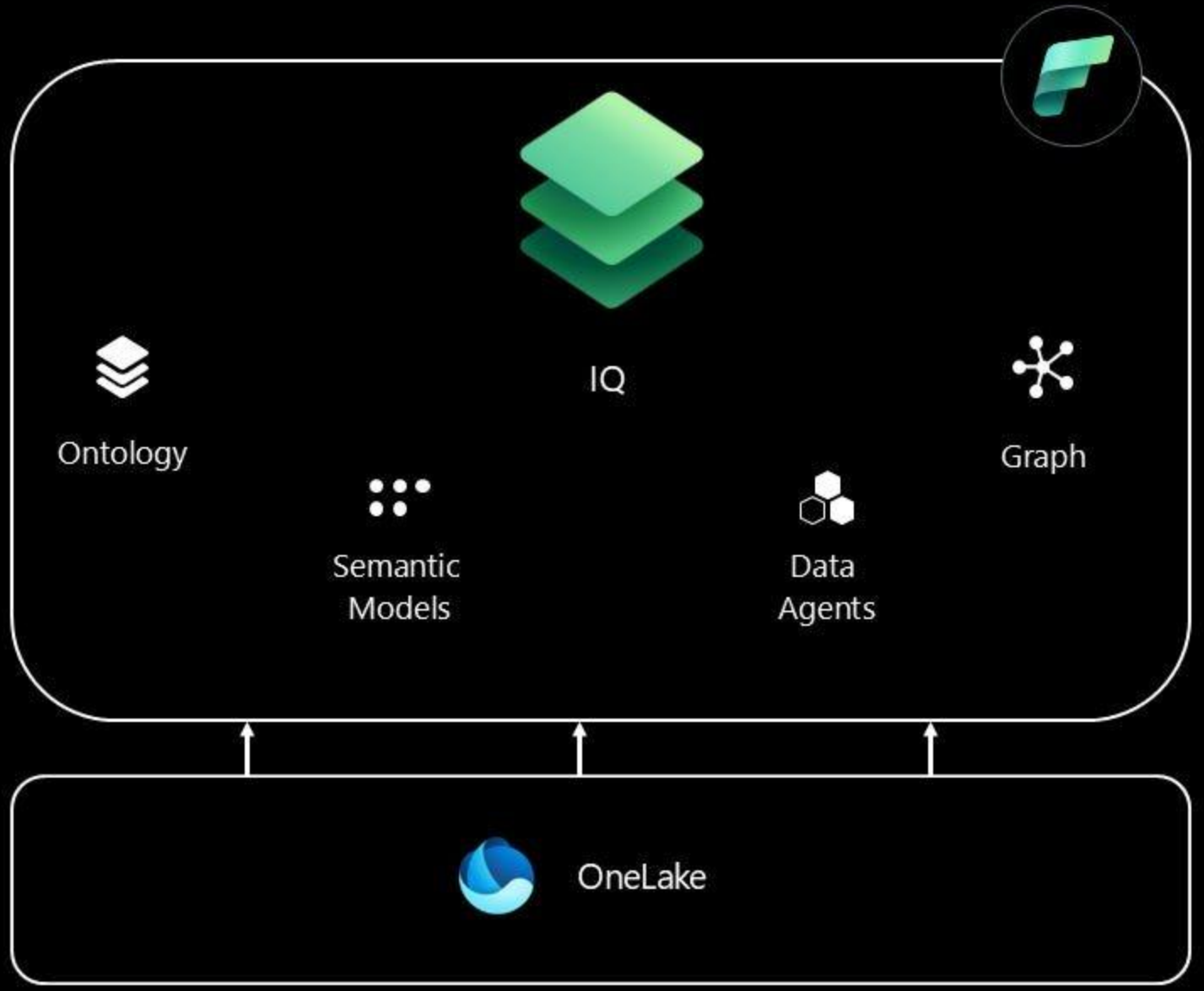
Power BI

Unified Data

Structured, Unstructured,
Real-Time, Graph

OneLake

Data-Marc.com



Data-Marc.com

Why use an ontology?

- Define entities, properties, relationships, rules and metrics.
- Modeled independently of individual tables
  (a definition can cross tables – different than Semantic Models!)
- Add hierarchies where they apply.
- Give AI agents shared context, ground agent experiences in governed definitions.

Model your business through your data!

Should we build an ontology?

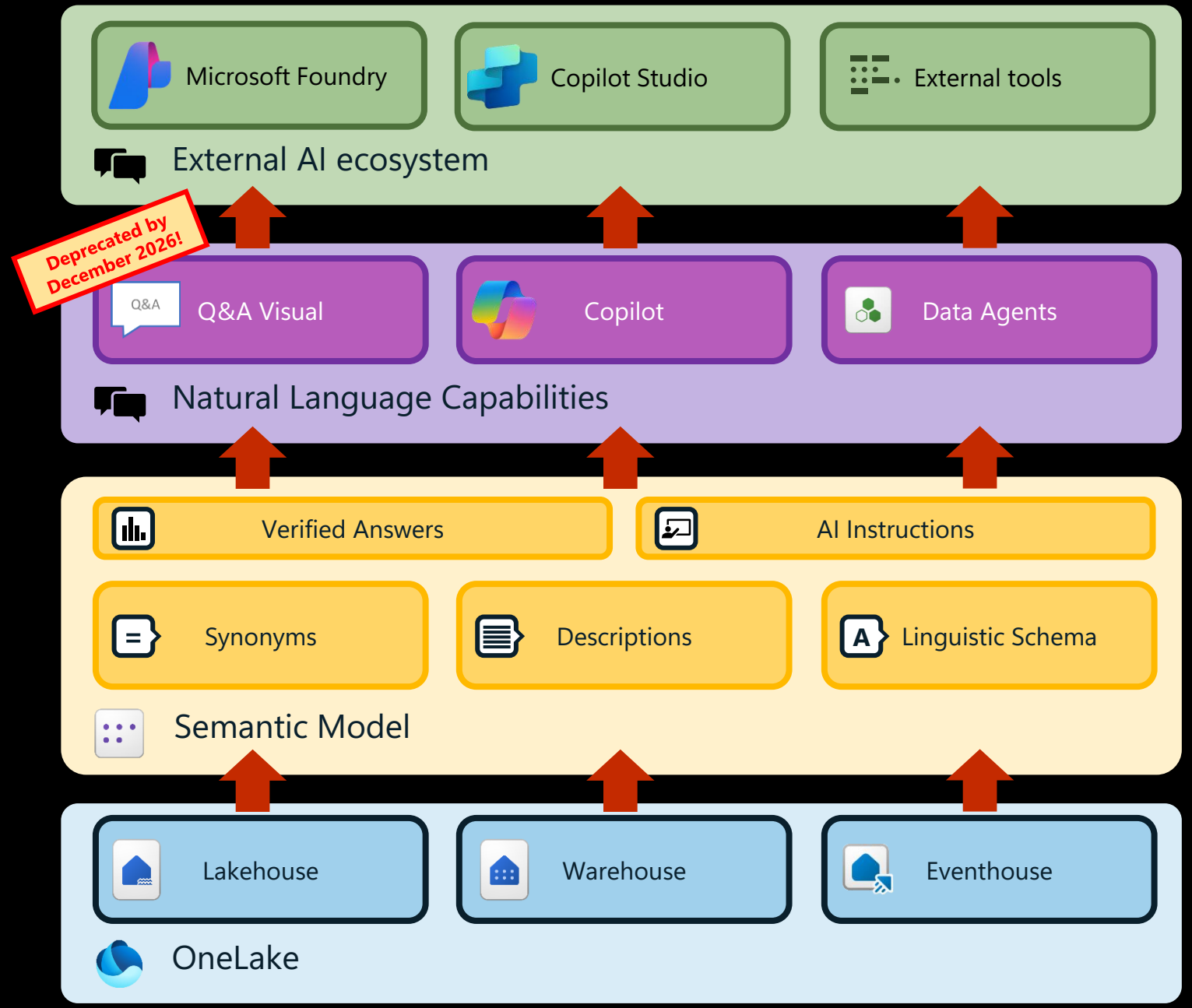
Or do we reuse (existing) semantic models?

Data-Marc.com



The AI ready Semantic Model

Why Semantic Models matter for AI experiences



Data-Marc.com

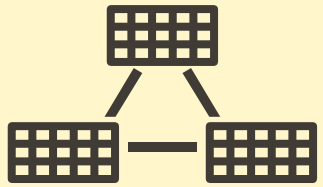
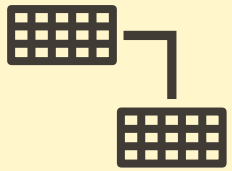
Facing the problem

We have 10+ years of legacy Power BI Semantic Models, which were build in times before AI.

Things that matter today for Large Language Models, were not prioritized in the past!

Modeling fundamentals

A **good** semantic model, covers at least the following items:

Star schema	Relationships	Object metadata	Measures	Hide & see
				

Humans vs AI

The gap between a model that works for humans and one that works well for AI

Semantic Model design:

- **Human interpretation:** Cust, Prod, FactSales, DimDate
- **Context around measures:** Sales, is it with or without VAT? And which measure represents Sales? Revenue, Gross sales, Net Sales, …
- **Ambiguity:** “Show our best customers” what is best?
- **Relationship implicitly:** [Customer] 1:N [Sales] Sales belongs to a customer but might be modeled differently. Inactive relationships, many-to-many, multiple date relationships (orderdate, shipdate, …)

AI challenge:

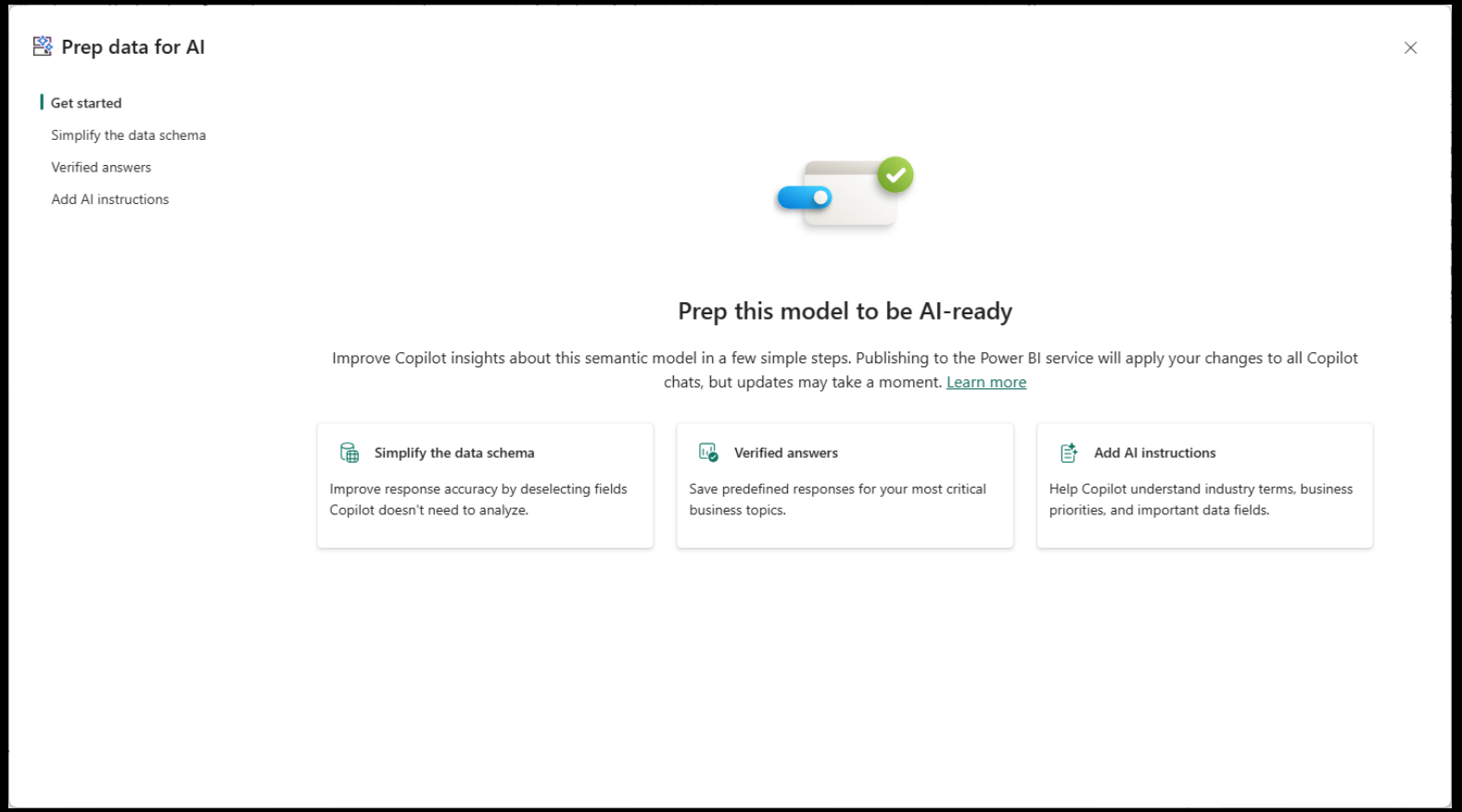
- Business meaning is not always explicitly represented in the model.
- AI must infer meaning from names, descriptions, metadata, relationships and calculations.
- The less explicit the semantics, the more room there is for incorrect interpretation.
- Unable to use features like Visual Calculations, Field parameters and more… (Report design related features)

What makes a Semantic Model AI ready?

Prep data for AI!

Three main pillars:

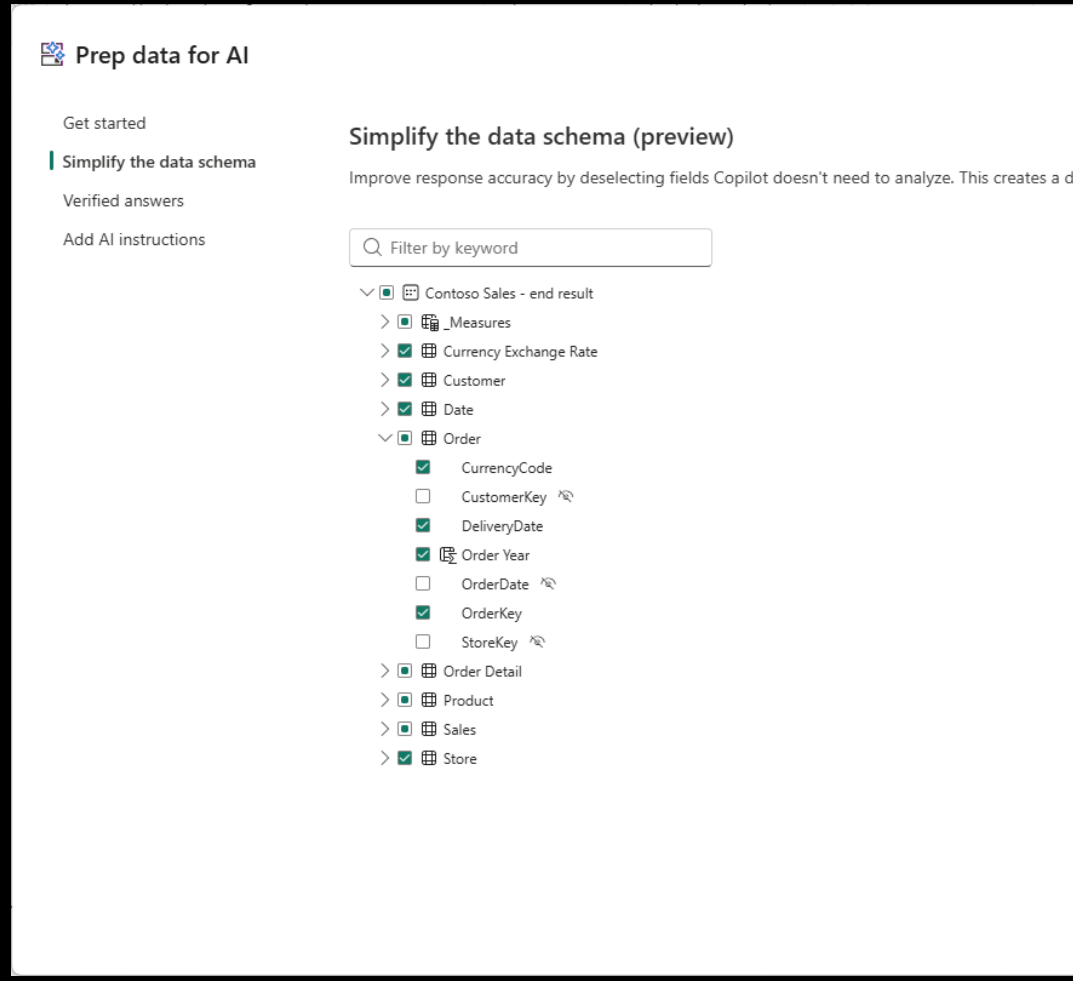
- Simplify schema
- Verified answers (links to visuals)
- AI Instructions



Prep data for AI: Simplify schema

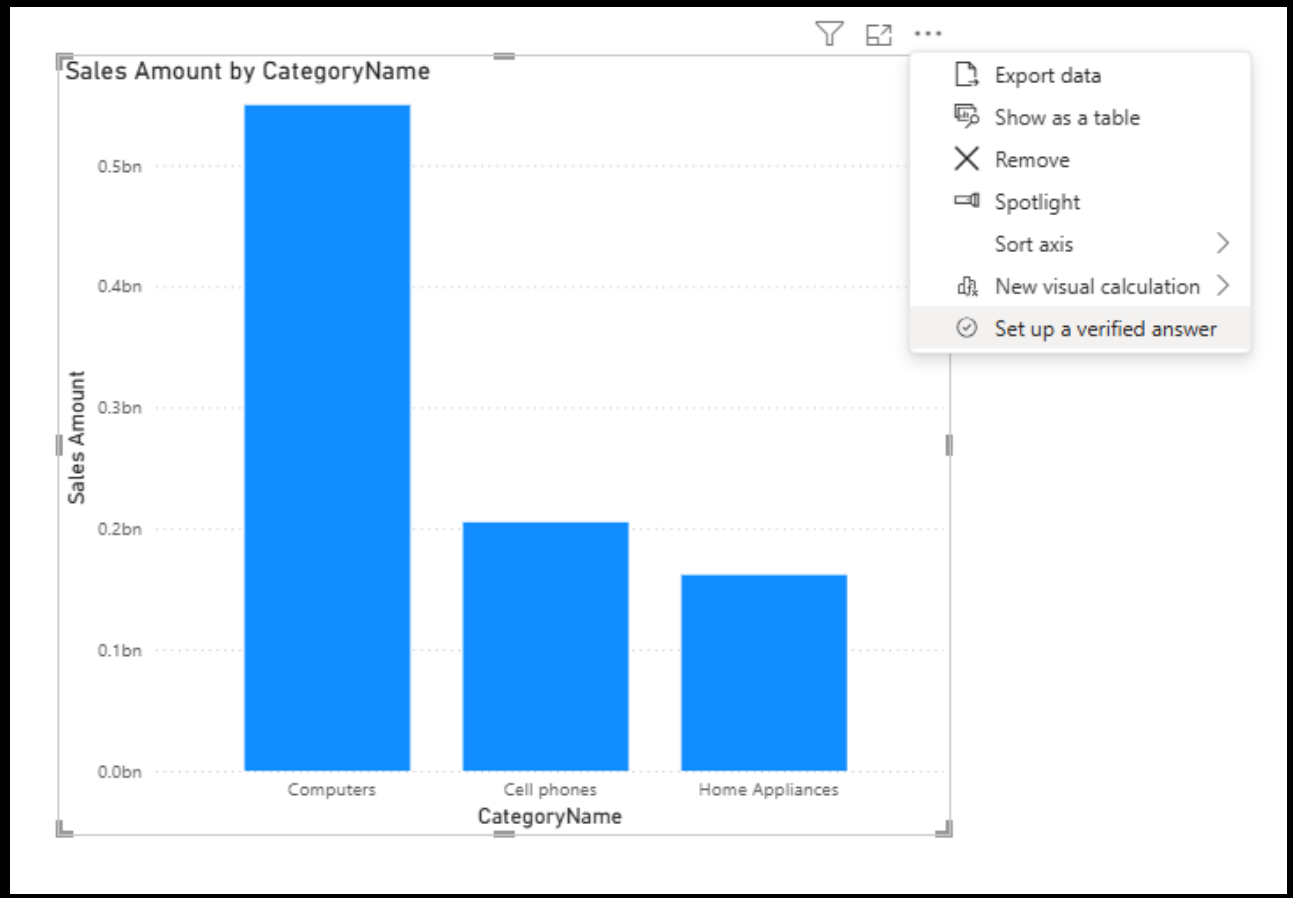
- Only keep what is relevant!
- Hide report-specific measures
  - Conditional formatting
  - Measures outputting texts like visual titles
- Disconnected tables likely cause issues
  → build relationships!

Keep in mind that Visual Calculations are not part of the semantic model!

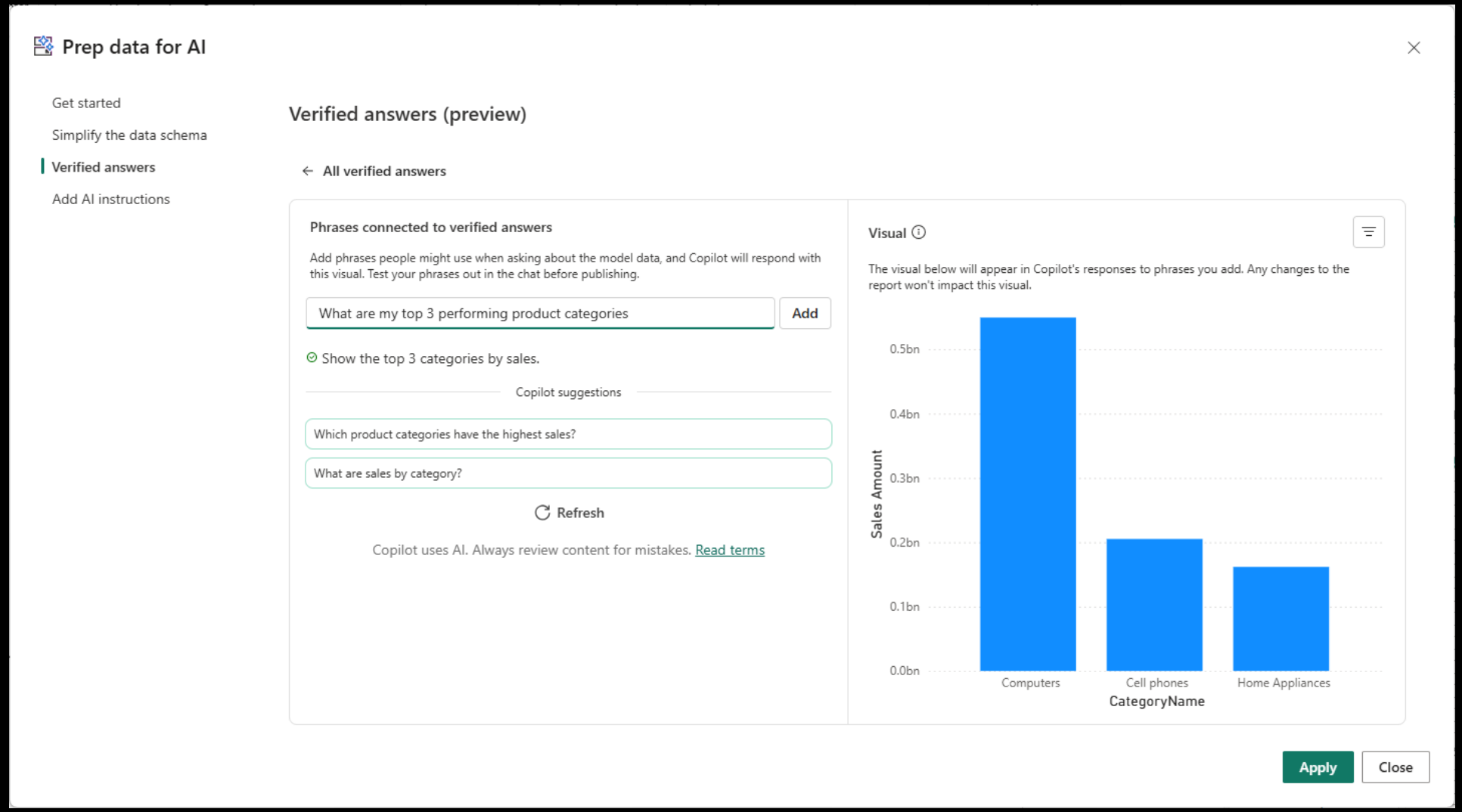


Prep data for AI: Verified answers

- Verified answers will always be referenced first
- Selected on visual level
- Linked to specific prompts
  - Copilot suggestions for relevant prompts
  - Ability to add your own prompts



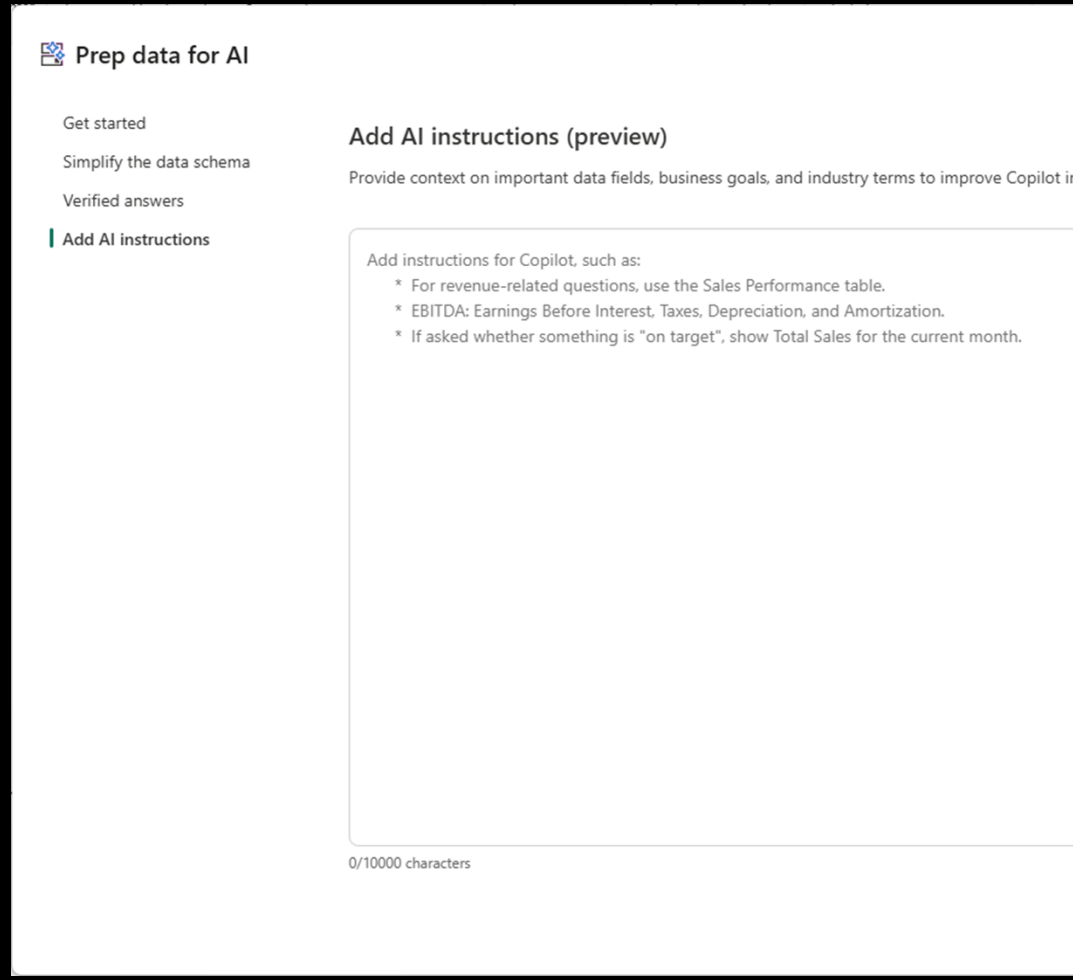
Prep data for AI: Verified answers



Data-Marc.com

Prep data for AI: AI Instructions

- Unstructured guidance (textual)
- No guarantee the guidance is followed exactly – see it as a hint!
- Response formatting
  - Style
  - Tone
- Business specific logic – for example:
  - Define what is a top performer – high or low?
  - Peak / off-season: which months are these?





Measuring AI readiness

Turning it into a measurable criteria

Introducing the **AI readiness score** for Semantic Models!

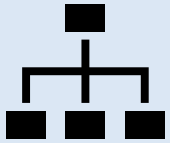
Discoverability & Metadata



Can AI discover and understand what the model contains?

Max score: 20

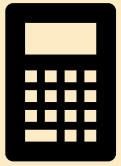
Model Structure & Organization



Is the model organized in a predictable way?

Max score: 20

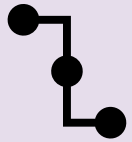
Measures & Calculations



Can AI reliably understand and use the calculations?

Max score: 20

Relationships & Model Logic



Can AI determine how entities relate and how filters flow?

Max score: 20

Business Semantics & Context



Does the model express business meaning not just technical structure?

Max score: 10

Quality & Trust



Are there structural (data-quality) issues that could influence AI answer quality?

Max score: 10

Discoverability & Metadata



Can AI discover and understand what the model contains?

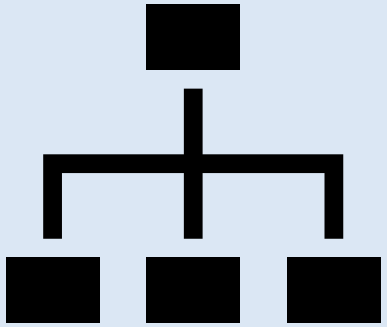
Max score: 20

What	Points	How to measure
Table descriptions	3	% of visible tables with a non-empty description
Column descriptions	4	% of relevant columns with a non-empty description
Measure descriptions	5	% of measures with a non-empty description
Business-friendly names	4	% of objects following naming rules / avoiding technical names and abbreviations
Synonyms defined	4	% of business facing (non-hidden) objects having synonyms defined

Metadata score =

Table coverage
+ column coverage
+ Measure coverage
+ naming quality
+ synonym coverage

Model Structure & Organization



Is the model organized in a predictable way?

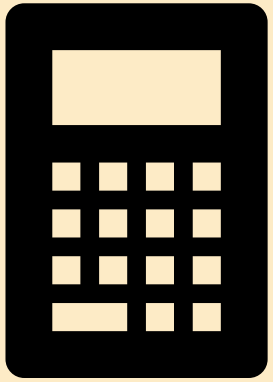
Max score: 20

What	Points	How to measure
Star schema characteristics	5	Dimensions & facts follow defined
Date Table is flagged as such	4	Is there a Date table present and flagged as such?
Facts & dimensions can be identified	3	Tables can be unambiguously identified, and other irrelevant tables are hidden
Technical tables are hidden (for AI)	4	Technical / helper tables are hidden for AI.
Auto summarization for numeric columns is set	4	% of numeric columns used in relationships or measures having auto summarization configured

Model structure score=

Star schema
+ Date table availability
+ Identifiable facts & dimension
+ Hidden technical items
+ Auto summarization config

Measures & Calculations



Can AI reliably understand and use the calculations?

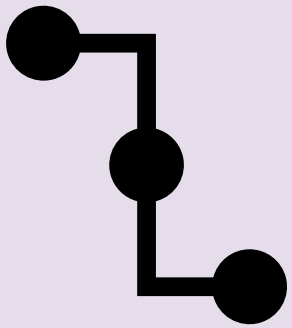
Max score: 20

What	Points	How to measure
Measures clearly named	5	Naming follows semantic/business naming conventions
Measures have descriptions	5	% description coverage
Format strings are applied	4	% measure having format strings applied
Time intelligence available	4	Typical patterns like YTD / LY / SPLY are available in the model
Measures are organized	2	Grouped together logically (with display folders)

Measure & calculations score=

Measure naming
+ Measure Descriptions
+ Measure with Format Strings
+ Time intelligence
+ Measure organization

Relationships & Model Logic



Can AI determine how entities relate and how filters flow?

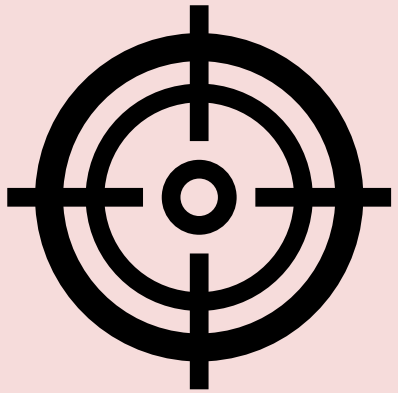
Max score: 20

What	Points	How to measure
Appropriate active relationships	4	% of expected relationships are set to active
Unambiguous filter paths	3	Detect ambiguous paths
Correct cardinality	6	1:N / 1:1 relationships only, bi-directionals should be avoided
Avoid unnecessary bi-directional filter paths	4	Avoid bi-directional filter paths
Relationships are documented (have a description)	3	% Relationship metadata availability

Relationships & Model Logic score=

Active relationships
+ Unambiguous paths
+ Cardinality
+ Bi-directional filter paths
+ Relationship documentation

Business Semantics & Context



Does the model express business meaning not just technical structure?

Max score: 10

What	Points	How to measure
AI Instructions / Notes for AI	5	Context and business specifics are described in the AI Instructions
Calculation groups used	2	Repeated (time intelligence / currency / …) calculation patterns covered in Calculation groups
Business context modelled in hierarchies	1	Logical drill-down structures
Units, currency & formatting defined	2	% / currency / quantity / decimal places / date formatting set for measures and numeric columns

Business Semantics & Context score=

AI Instructions
+ Calculation groups
+ Hieraries
+ Formatting

Quality & Trust



Are there structural (data-quality) issues that could influence AI answer quality?

Max score: 10

What	Points	How to measure
No columns with solely the same value or empty	3	Validate availability of data and cardinality of columns
Data types are set consistency on both ends of the relationship	2	Inconsistent datatypes will make relationships unusable
No duplicate measures	2	Different measures with same definition will result in score 0
Security roles configured	2	Security roles created and have expressions added
Security roles documented	1	Descriptions on roles added

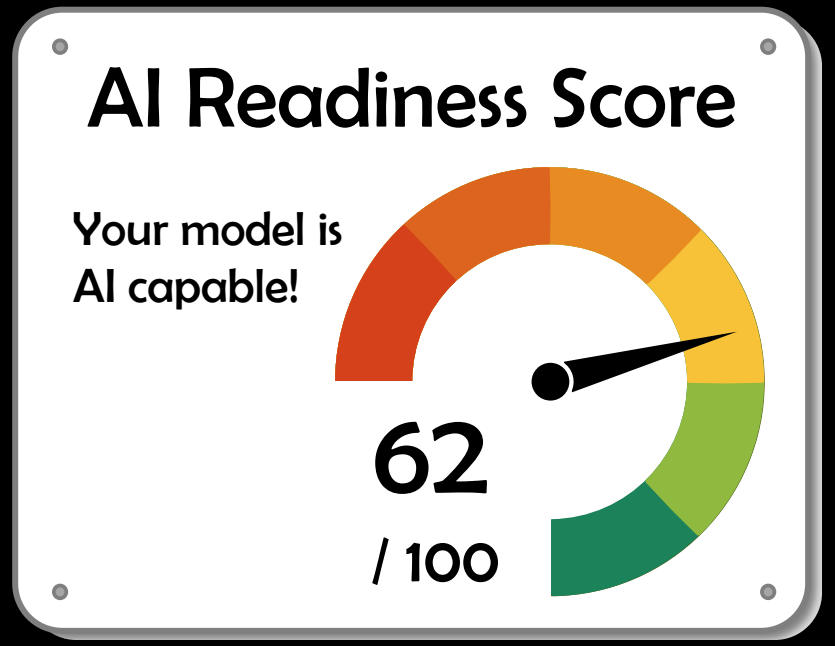
Quality & Trust score=

Columns with one value
+ Data types across relationships
+ Duplicate measures
+ Security roles configured
+ Security roles documented

The AI Readiness Assessment

A semantic model can be perfectly usable for humans while still leaving important context implicit for AI.

The AI Readiness Score makes these gaps **visible, measurable, and actionable.**



0-19	20-39	40-59	60-74	75-89	90-100
AI Unready	AI Limited	AI Emerging	AI Capable	AI Mature	Ready!
Fundamental issues remain	Major gaps for AI	Foundation is taking shape	Usable, improvement advised	Strong foundation for AI	Excellent!

Common issues

- Conflicting or overly complex instructions
- Skipping schema refinement – especially with large semantic models containing many tables
- Not using a star schema (resulting in inefficient DAX)
- Relying on hidden fields – copilot cannot access them
- Including unnecessary measures: Simplify the schema,
  - Remove measures that are not relevant or broken,
  - Hide report measure (for conditional formatting for example),
  - Avoid duplicating logic in different measures
- Non-descriptive naming (technical names for objects)
- Relying on implicit measures: high chance of inconsistency!
- Ambiguous date fields (e.g. Order Date, Ship Date, …): use prep data for AI to specify when to use which

Why manual assessments don't scale?!

Manual assessment	The problem
Open the semantic model manually	Repeat for every semantic model in scope
Check tables & columns	Lots of meta data to process
Review measures	Business logic is harder to assess though, technical assessment only cannot do the job either
Check names & synonyms	Subjective and inconsistent
Record findings	Documentation quickly becomes outdated
Repeatability after every change	Continuous maintenance becomes impossible
Self-service users should be able to run the assessment themselves	Requires understanding across all users

The result?
Slow, subjective, inconsistent and not easily repeatable

Bringing in Semantic Link (Labs)

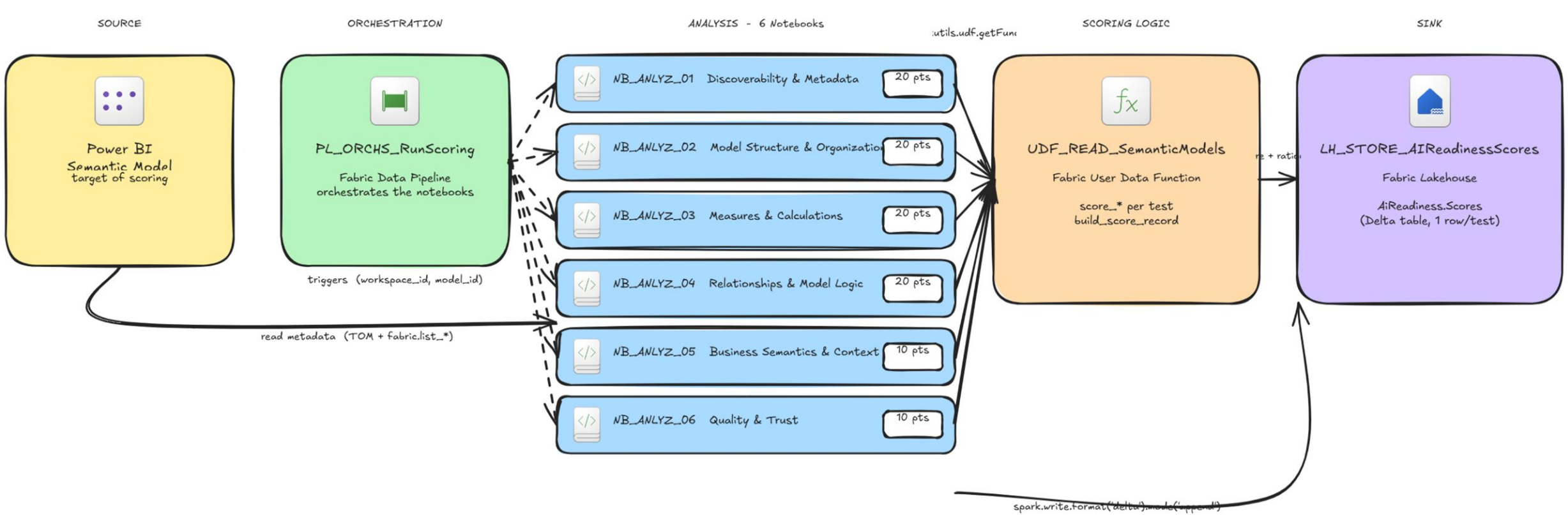
Instead of asking "is this model AI-ready?"
we can start measuring the characteristics that make it IT-ready!

Semantic Link

- Python library for interacting with semantic models
- Used inside of a Fabric notebook
- Pre-installed in the Fabric runtime
- Supported by Microsoft

Semantic Link Labs

- Open-source extension of Semantic Link
- Higher level inspection functions
- Model management capabilities



The architecture

Which components are part of this automated solution?



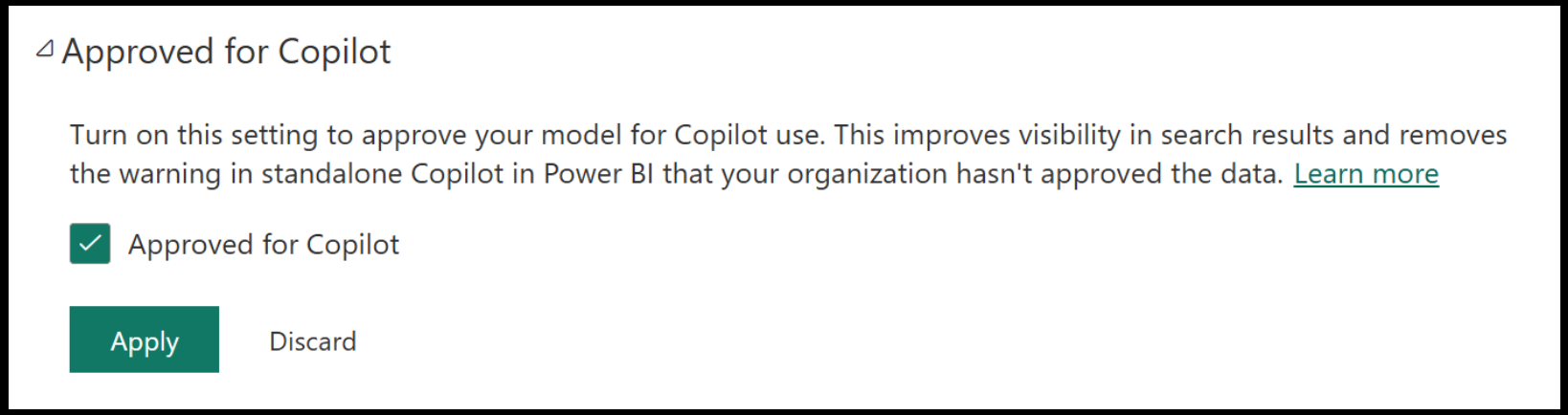
Data-Marc.com

Semantic models evolve over time,
and so should your AI readiness score

Data-Marc.com

Assessment integration

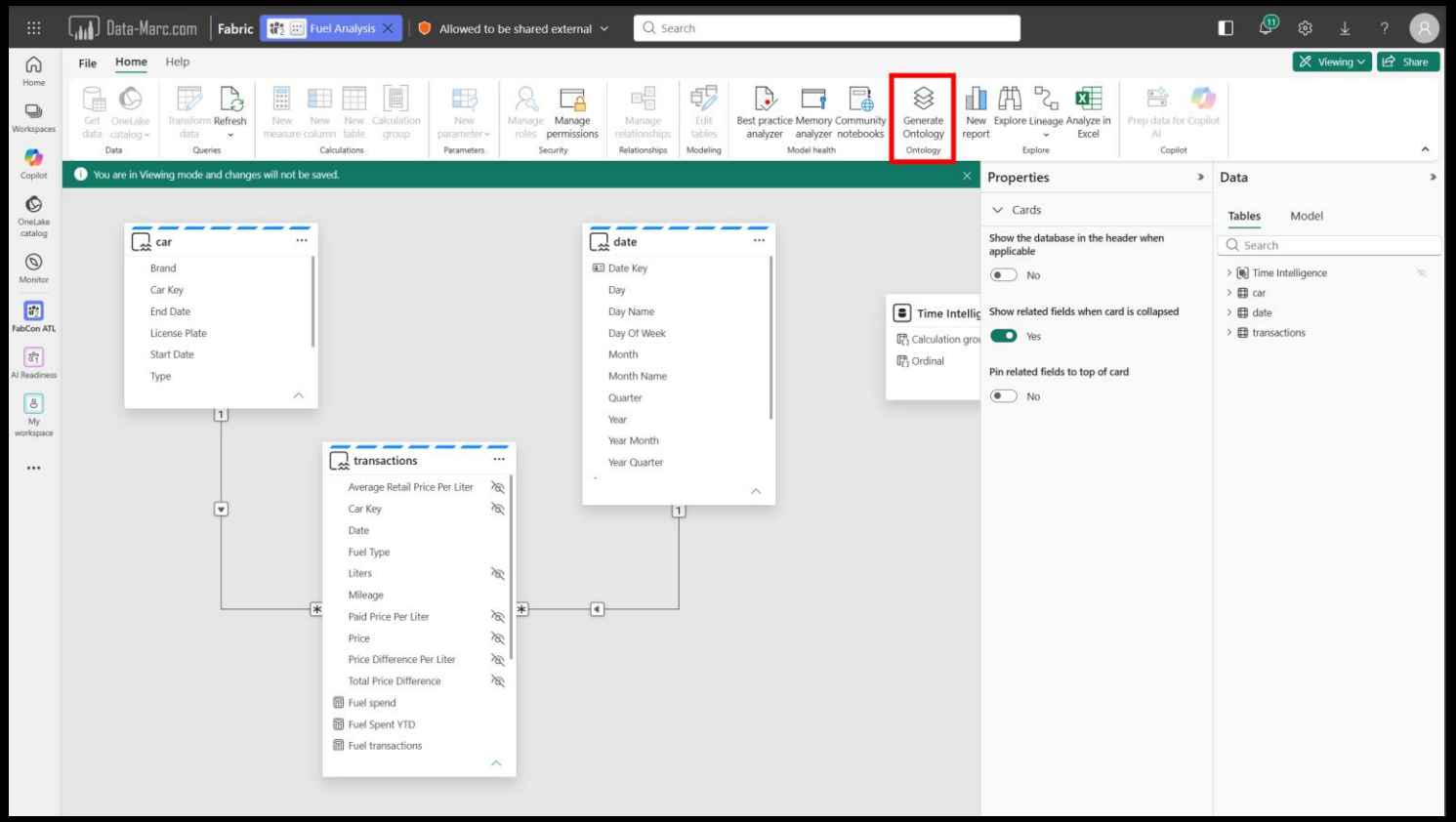
- Availability of Semantic Models (feature flag) for AI
- As part of regular governance / monitoring processes
- Automation of Copilot capacities



How ontologies fit in

Ontologies can be created from Semantic Models.

- Reuse existing definitions
- Include calculated columns and measures (which are not always part of upstream source)

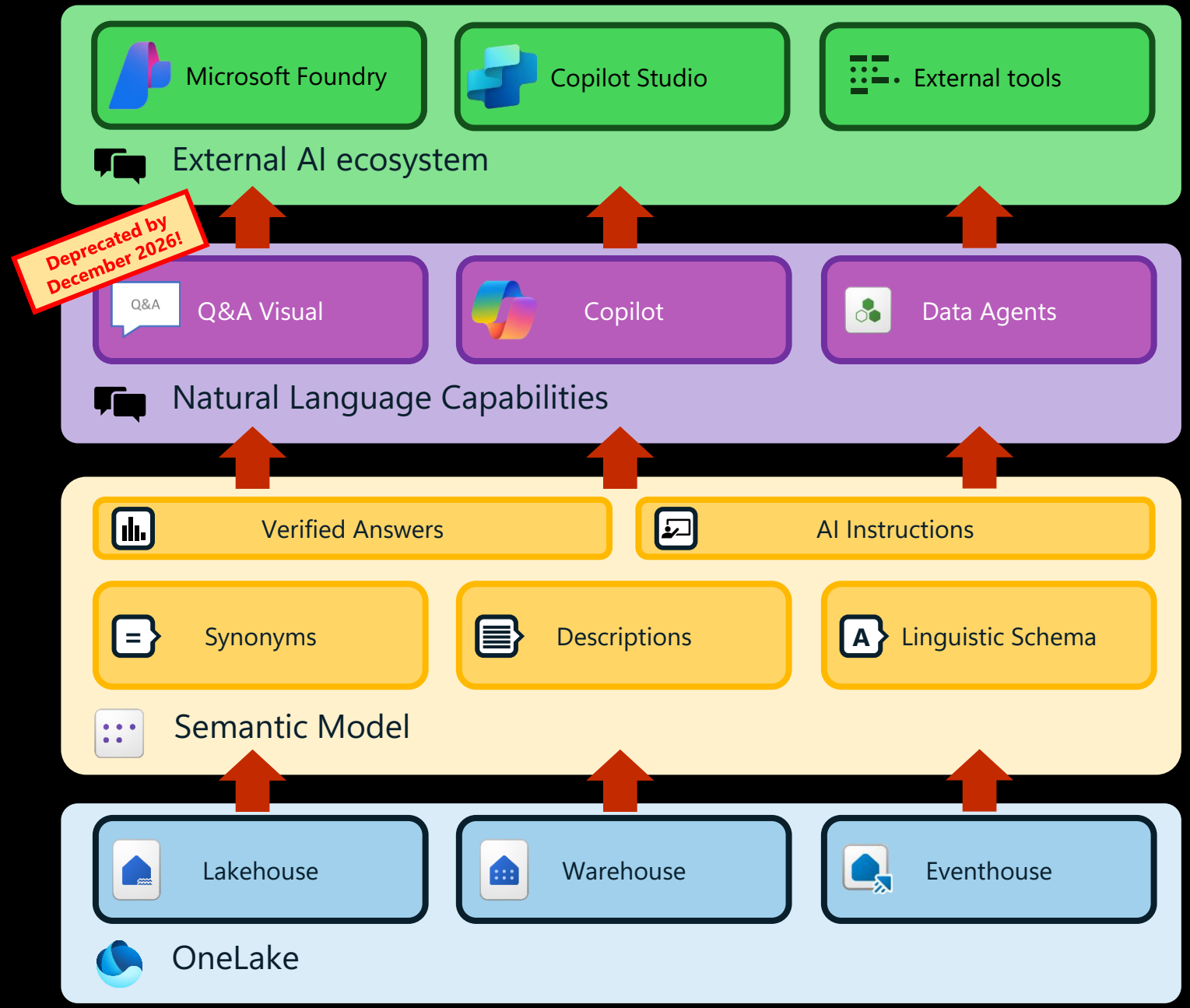


Mastering Fabric Data Agents

From Setup to Success



Where does natural language interactions fit in?



Data-Marc.com

Exploring solution areas
What are the alternatives to copilot?

What can you conclude from the orders placed versus the orders shipped?

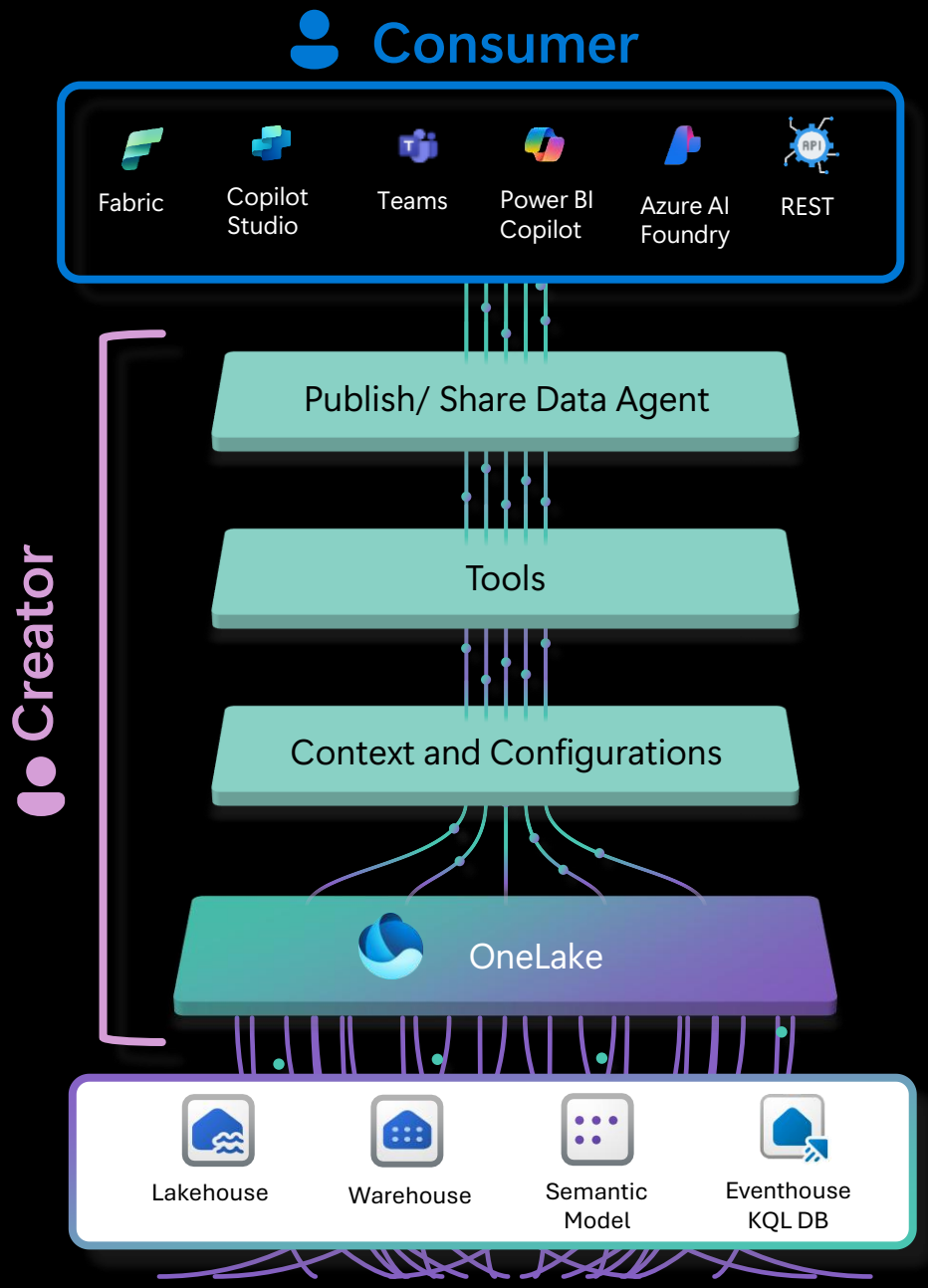


Prerequisites for Data Agents

- Fabric paid capacity (any size)
- Fabric Data Agent tenant setting is enabled
- Copilot tenant switch is enabled
- A valid source (Warehouse, Lakehouse, Power BI semantic model or KQL Database)
- Power BI semantic models via XMLA endpoints tenant switch is enabled.

Introducing Data Agents

- The **conversational Data Agent** allows users to interact with data naturally, enhancing accessibility and usability.
- Improved chat canvas for creators with **new debugging capabilities**, making it easier to understand and refine responses.
- Seamlessly reason over multiple data sources, including **Lakehouses, warehouses, semantic models and Eventhouse KQL databases**.
- Your Data Agent can be consumed inside and outside of Fabric. Stay tuned for **integrations** with Copilot Studio, Teams, Azure AI Foundry and your own custom applications.



Different types of supported sources



Lakehouse



Warehouse



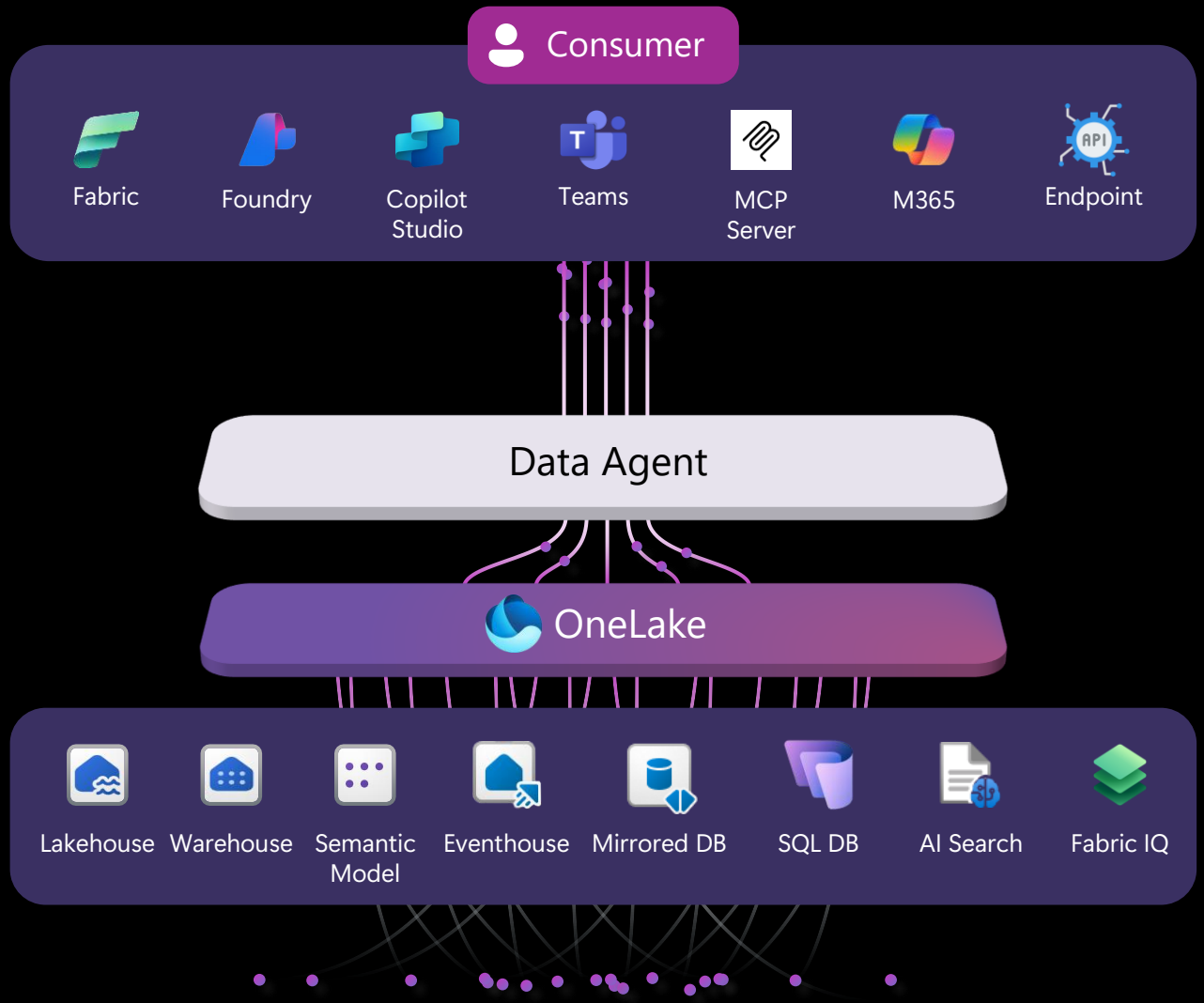
Eventhouse



Semantic Model

And related sources (like mirrored sources)

Fabric Data Agents



A **virtual analyst** that allows users to interact with and gain insights from enterprise data in OneLake

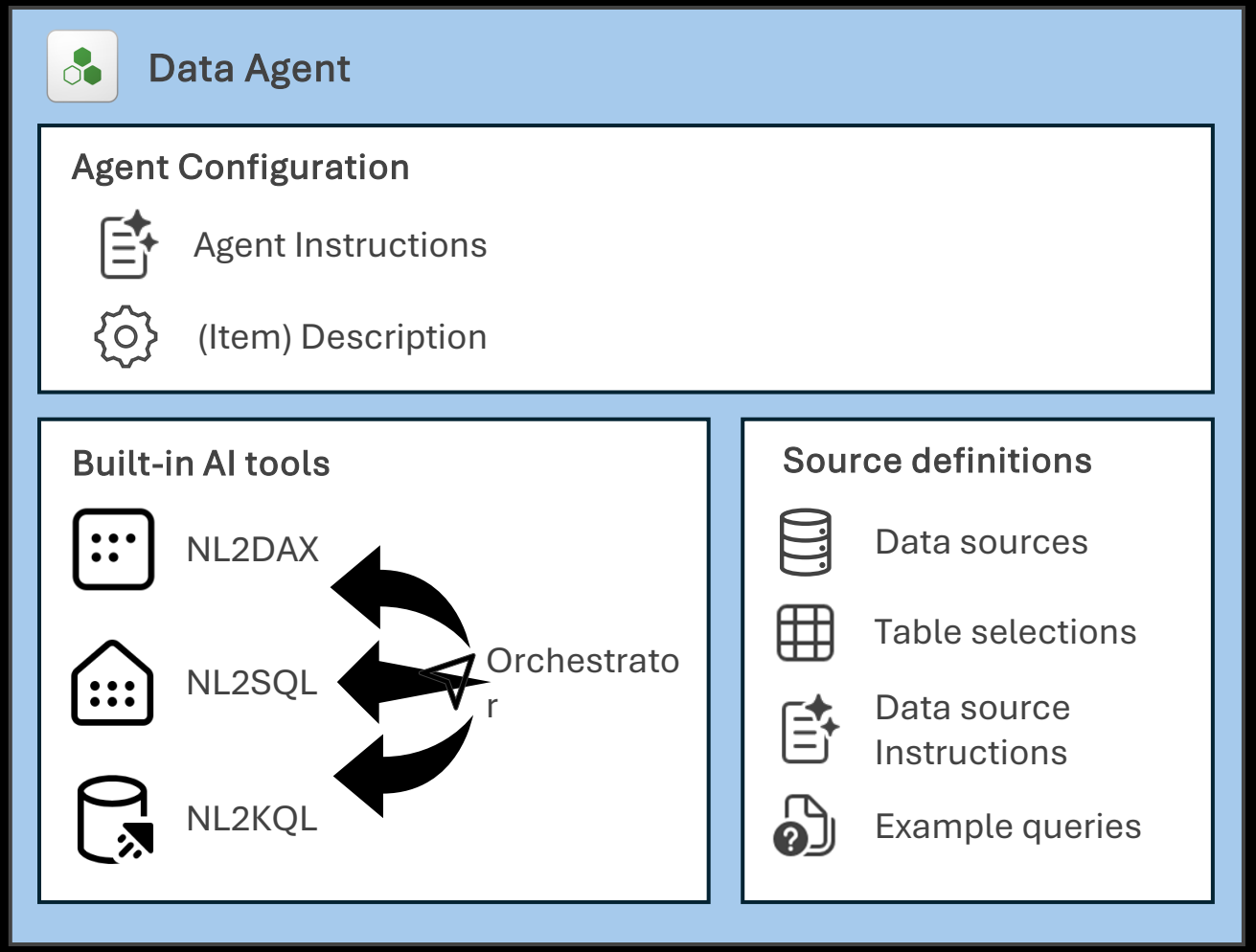
New Consume your data agents in **M365 Copilot** and as **remote MCP Server in VS Code**

New Data agents now support unstructured data through **Azure AI Search**

New Data agents now support **Fabric IQ Workload through Ontology**

Data-Marc.com

Data Agent building blocks



Creator configurations



Instructions (schema)

Verified answers

Agent Instructions

Data Source Instructions

Example Queries

Intended Use Cases:

- Specify which data sources to route for specific questions
- Guide how the agent interprets user intent

What to include:

- Map terminology to user intent
- Define steps for handling key questions
- Set response style and tone

Intended Use Cases:

- Provide clarity on how to use each source
- Guide table selection for topics
- Ensure correct query logic

What to include:

- Descriptions of key tables & columns
- Recommended table usage for certain topics
- Example column values and filter patterns

Intended Use Cases:

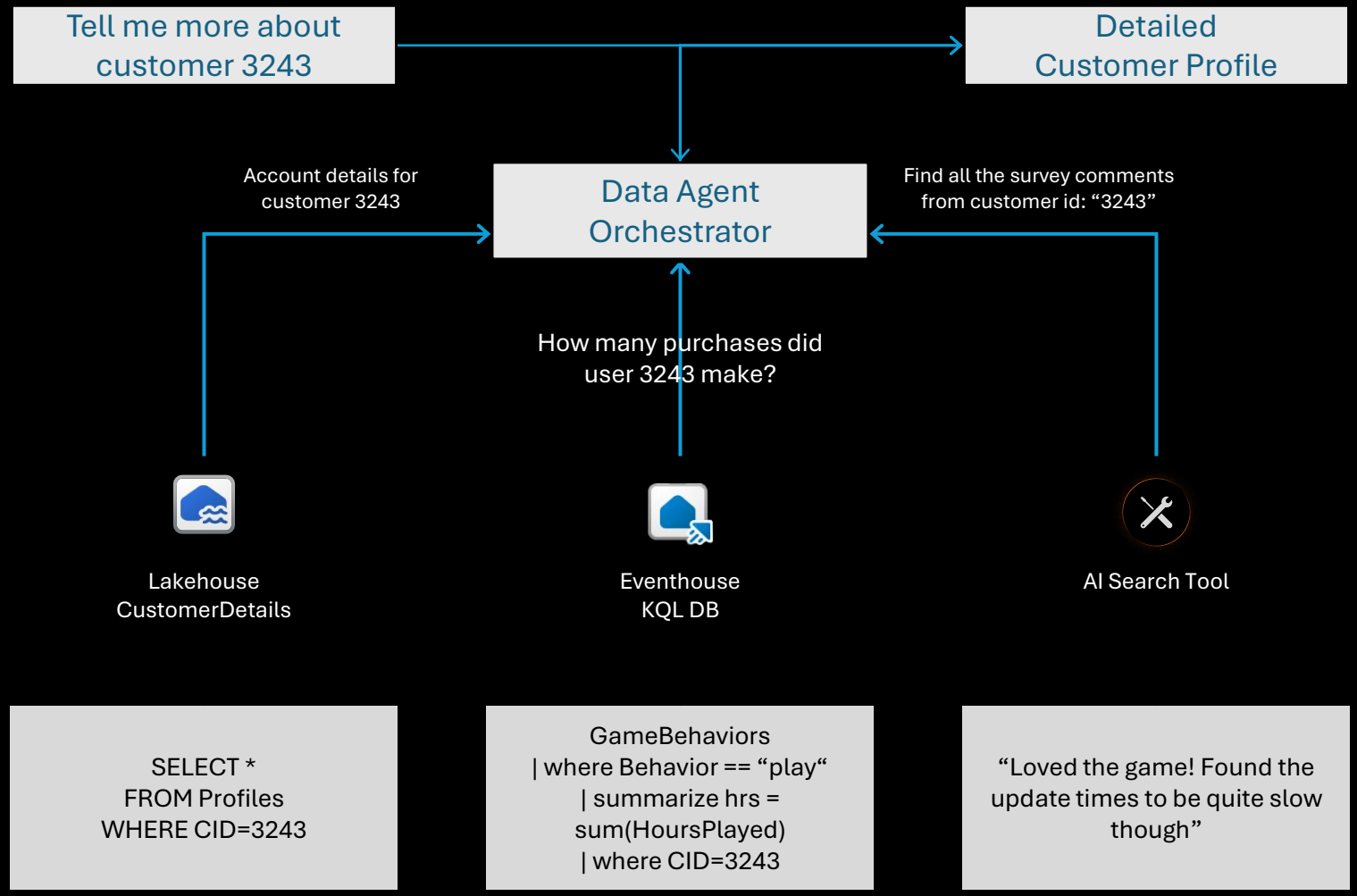
- Reflect advanced logic
- Examples should highlight join patterns or expected values

What to include:

- Representative set of questions a consumer may ask

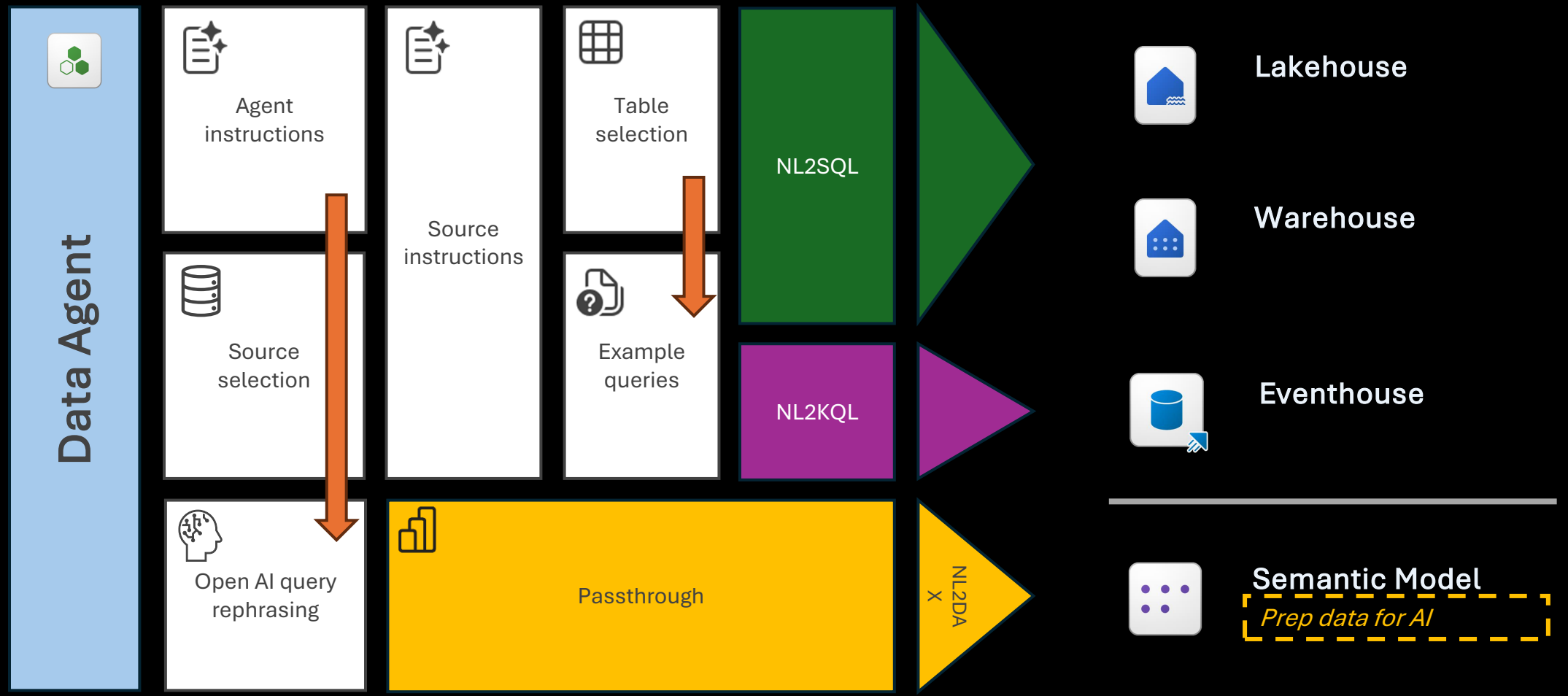
Data-Marc.com

The Data Agent flow

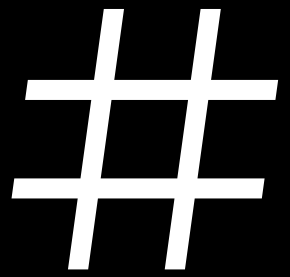


1. The data agent delegates questions to specific tools or data sources.
2. Data Sources are specialized at taking a natural language question and generating a response – query or result.
3. Data Sources & Tools supply their response back to the orchestrator – allowing it to process the information and think about the next step or return the summarized result.

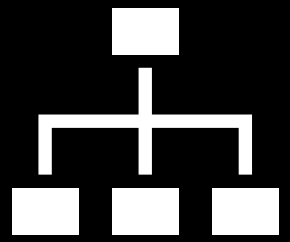
Data Agent Orchestrator - Inner workings



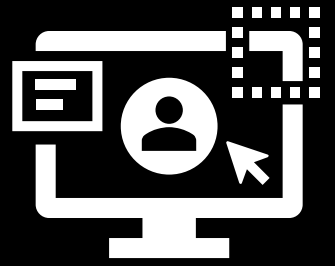
Advantages of using a Semantic Model as a data source



Allows metadata enrichment



Business logic (measures)



User friendly and terminology



Single source of truth (incl reports)

Keep in mind!

The answers will only be as good as your data!

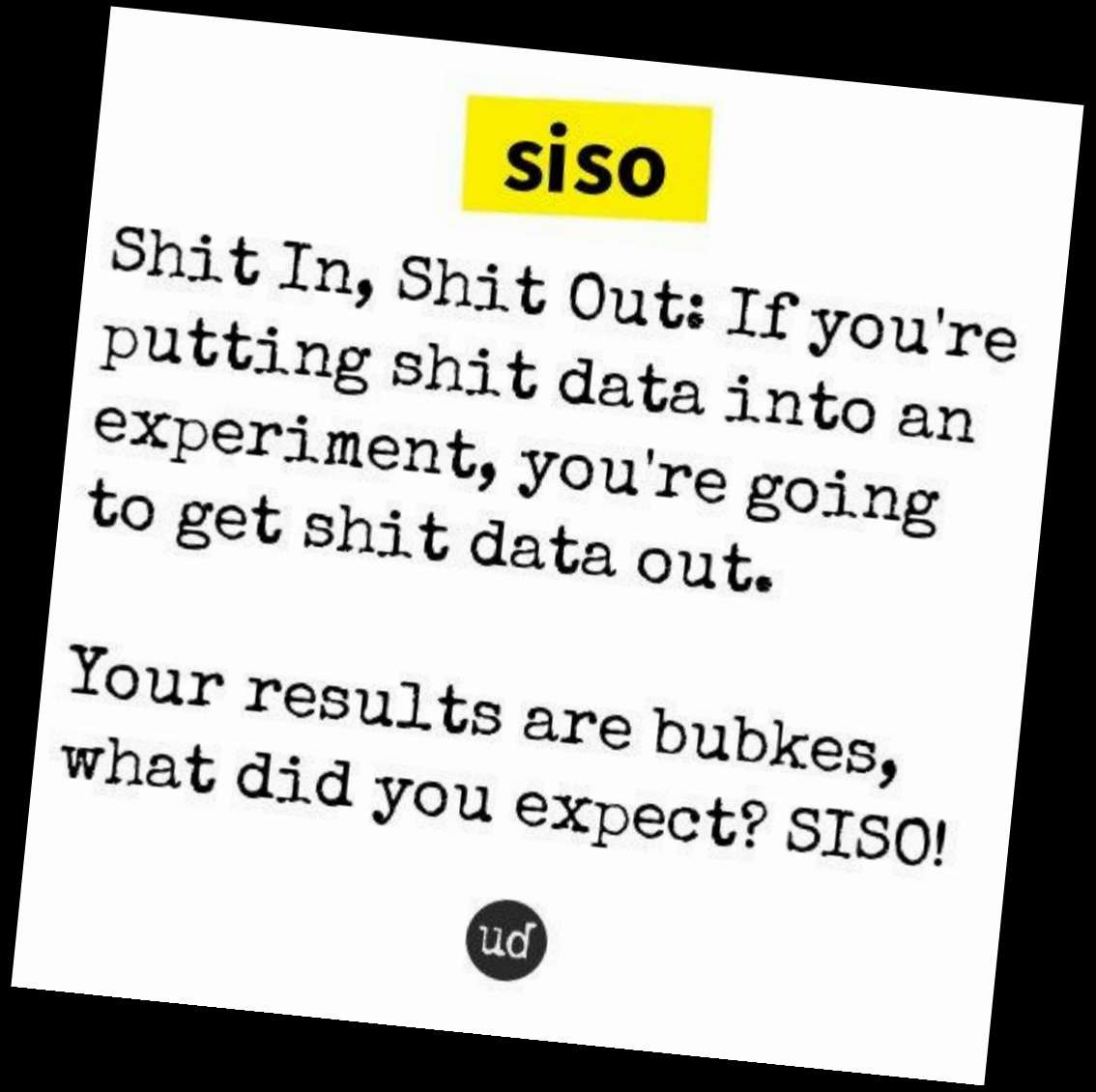


Image from Urban Dictionary: https://x.com/urbandictionary/status/958044898212630528

Diagnostics

“AI is wrong” to structured root cause analysis!

Where to start your investigation when the answer is not what you’re looking for

Permissions

- RLS / CLS
- Item level access
- governance policies

Prompt & grounding

- Clear question
- Proper metadata

Source grounding

- Semantic model, lakehouse, warehouse, eventhouse
- Relationships and item naming

Notes for model

- Tone & voice
- Context (what is a good performer, high or low?)
- Hints (example queries)

Generated query analysis

- Diagnostics

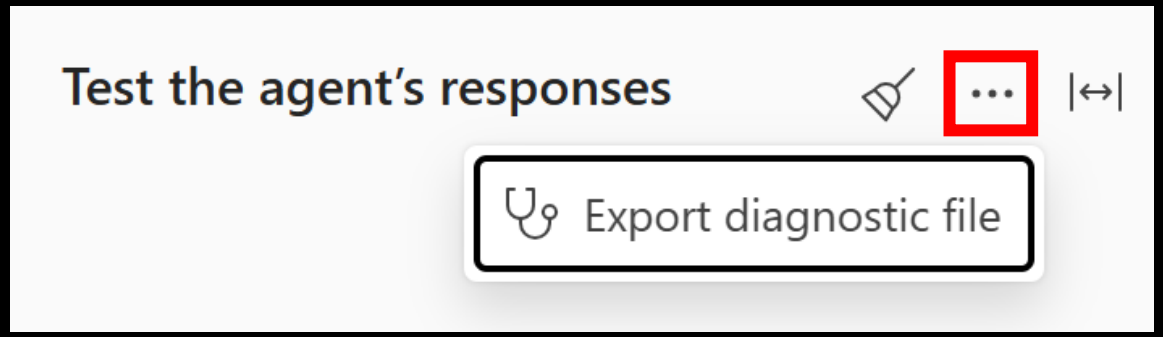
Capacity & performance

- Storage mode (for semantic models)
- Model size
- Load on the capacity

Diagnostics

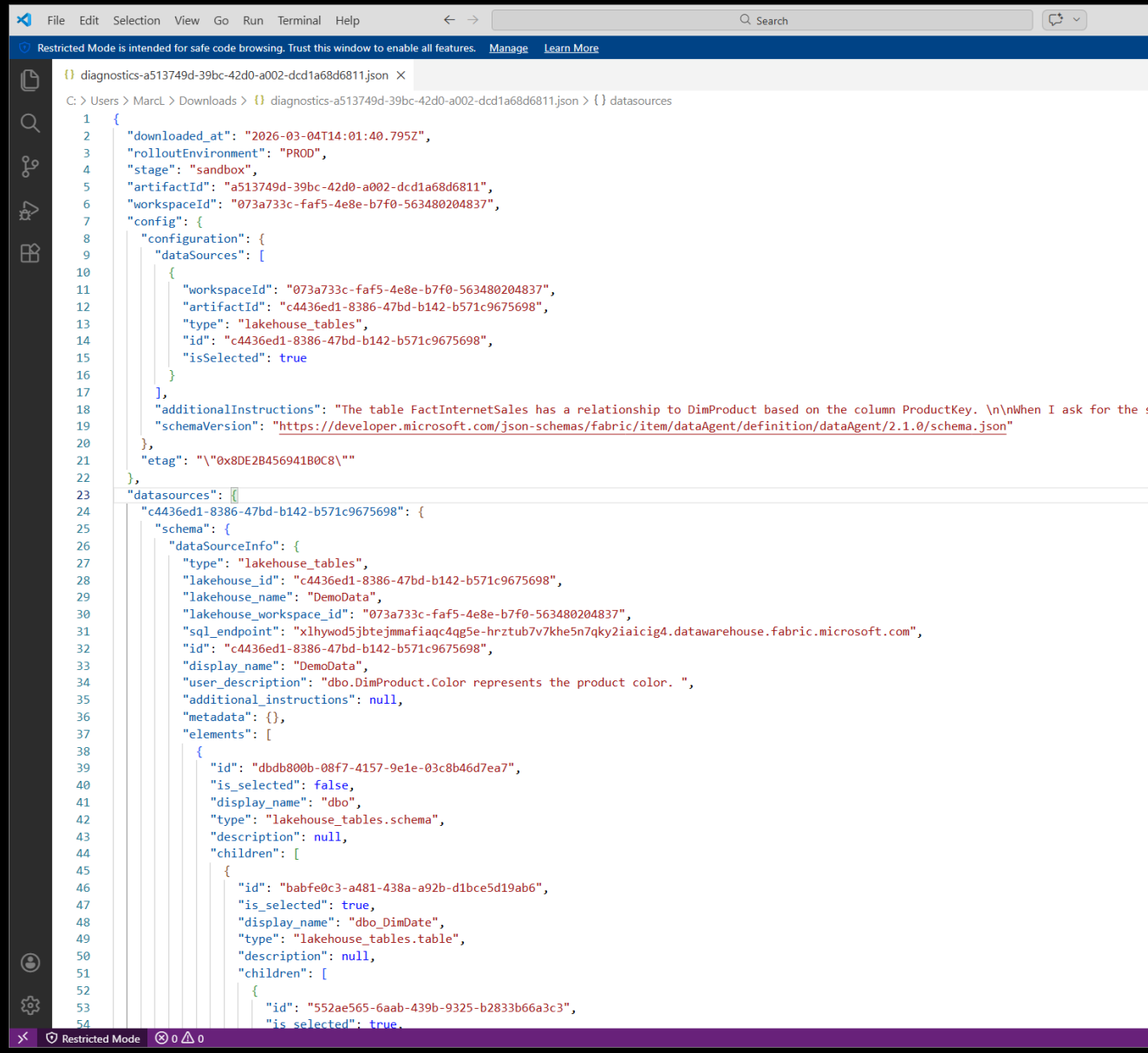
“AI is wrong” to structured root cause analysis!

Where to start your investigation when the answer is not what you’re looking for?



- User prompts
- Data source schemas and configurations
- File metadata and contents
- Environment and artifact identifiers (GUIDS)
- Any proprietary information you may have entered

Demo
Data Agent Diagnostics

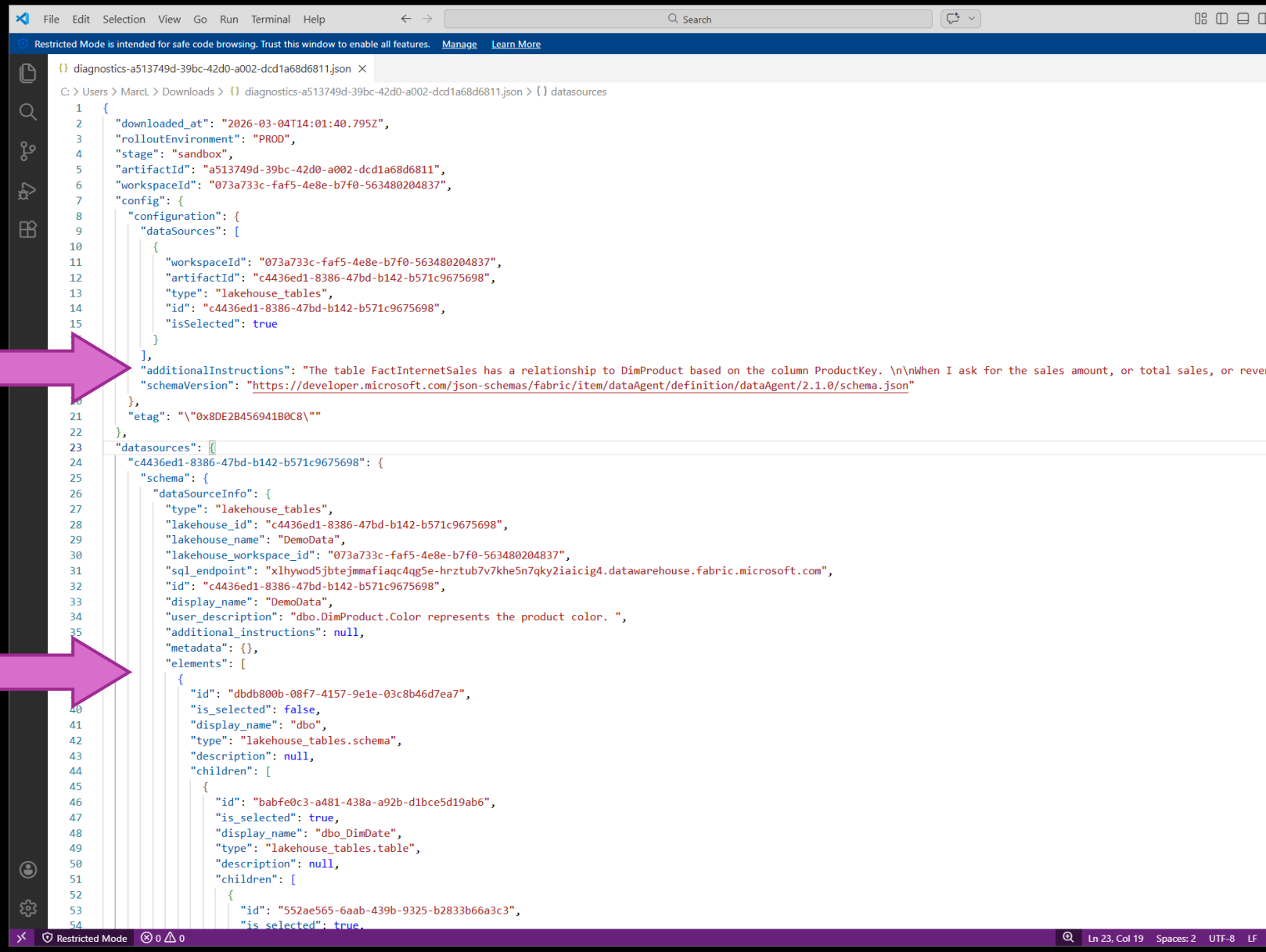


Diagnostics

Source instructions

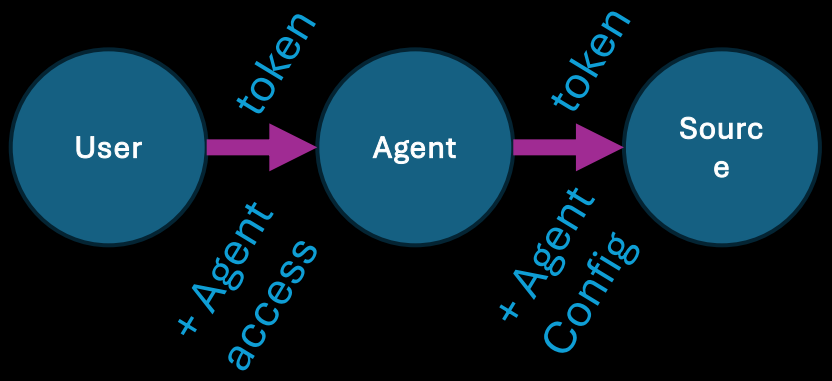
Schema information

...and much more!



Permissions

Data access



Pass-through mechanics

- User credentials flow to the data source
- Permission evaluated per-user, per-query
- RLS/OLS etc. are baked-in

Service principals (SPN) are ~~**NOT**~~ supported, ~~Access requires a user identity.~~

Data source permission requirements

Semantic model

Read permissions only! *(previously also build)*

Agent access

Permission	Description	View Config	Edit Agent
Default (None)	N/A	❌	❌
View details	View the configuration and settings, but make no changes.	✅	❌
Edit and view details	Edit the configuration and settings.	✅	✅

Data Agent SDK

Data Agent SDK

Programmatic Management: Create, update, and delete Data Agent artifacts seamlessly.

Data Source Integration: Easily connect to and integrate multiple data sources, for enhanced data analysis and insight generation.

OpenAI Assistants API Support: Use the OpenAI Assistants API for rapid prototyping and experimentation.

Workflow Automation: Automate routine tasks to reduce manual efforts and improve operational efficiency.

Resource Optimization: Optimize configuration and management of Data Agent resources to better align with specific customer needs.

Copy

```
%pip install fabric-data-agent-sdk
```

Data-Marc.com

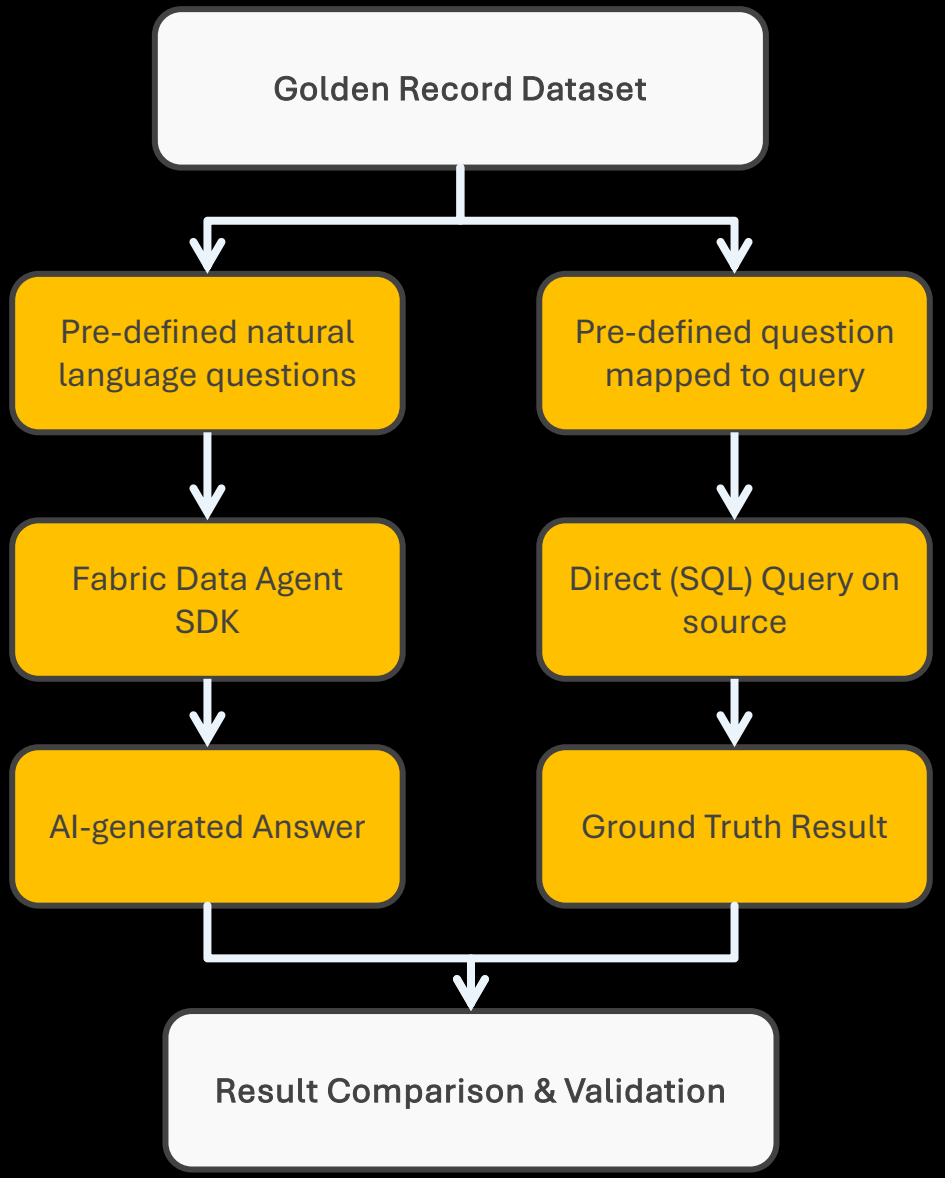
Programmatically create Data Agents

Data Agent Testing

Golden record testing

- Establishes trust by validating your Data Agent against a known, authoritative dataset.
- Results are compared one-to-one
- Enables objective evaluation

Gains confidence in the data agent, produces accurate, explainable and repeatable answers



Bringing in Semantic Link (Labs)

Instead of asking "is this model AI-ready?"
we can start measuring the characteristics that make it IT-ready!

Semantic Link

- Python library for interacting with semantic models
- Used inside of a Fabric notebook
- Pre-installed in the Fabric runtime
- Supported by Microsoft

Semantic Link Labs

- Open-source extension of Semantic Link
- Higher level inspection functions
- Model management capabilities

What is Semantic Link?

- Python library for interacting with semantic models
- Used inside of a Fabric notebook
- Pre-installed in the Fabric runtime
- Supported by Microsoft
- List_* functions to view semantic model metadata
- Functions to execute DAX, TMSL, XMLA, DMVs
- Basic API wrapper functions (list_capacities, list_workspaces, list_reports, list_items)
- And more…

Semantic Link – Use cases

Power BI general

- Documenting Power BI items
- Move Power BI items across workspaces
- Detect broken reports
- Rebind reports
- Set a report theme
- Migration of report-level measures to the semantic model
- Tenant Settings tracking

Semantic Models

- Best Practice Analyzer
- Vertipaq Analyzer
- Semantic model edits (TOM)
- Metadata translations
- Semantic model refresh
- Visualize a refresh
- Semantic model backups
- Run DAX with impersonation
- Manage Query Scale Out

Direct Lake

- Migration to Direct Lake
- Check Direct Lake guardrails
- Warm the cache for Direct Lake
- Analyze Delta tables for Direct Lake
- Fallback to DirectQuery diagnostics
- Update connection of a Direct Lake semantic model

Capacities

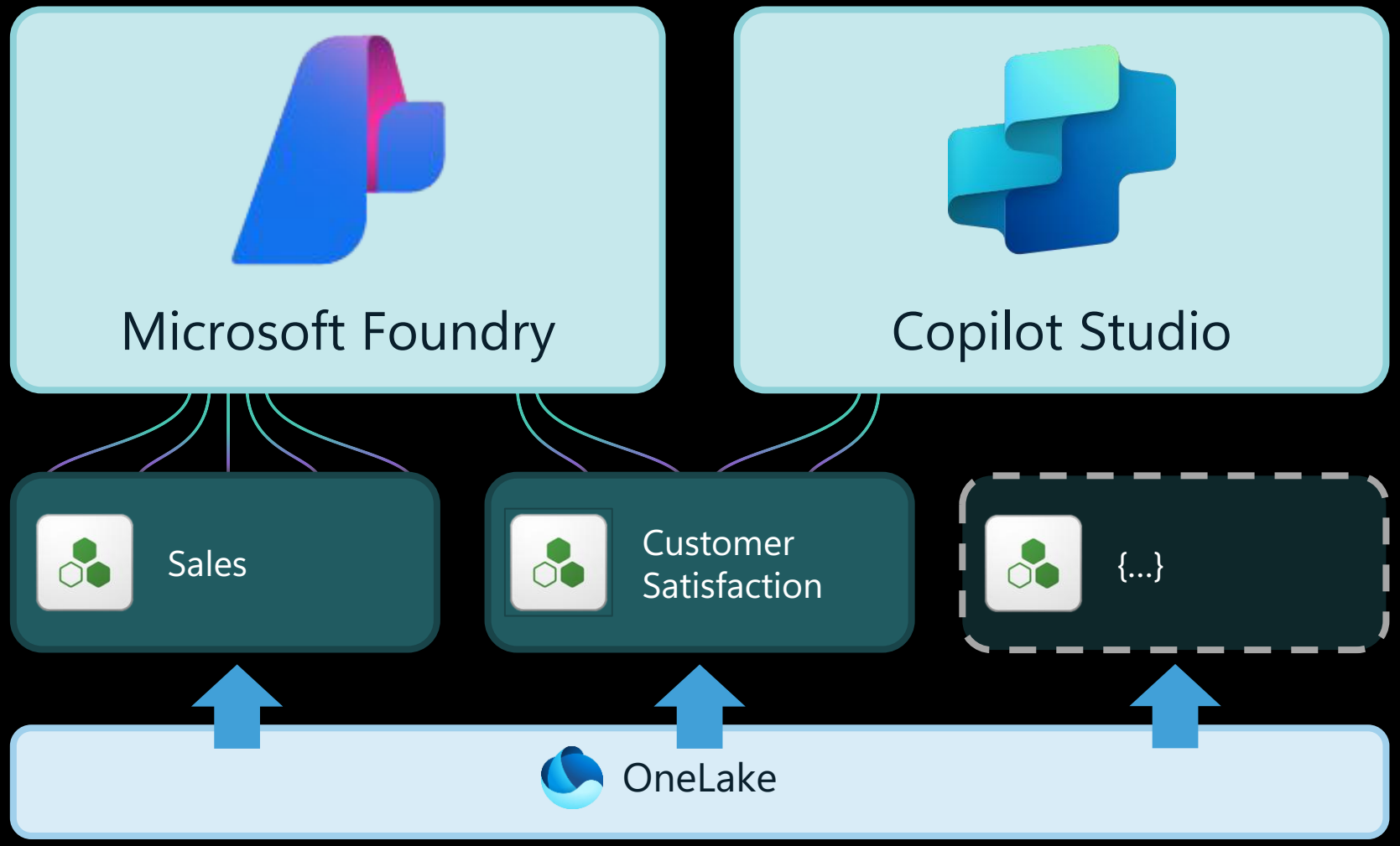
- Migration from P SKUs to F SKUs
- Migration from FT SKUs to F SKUs
- Capacity management

And more…

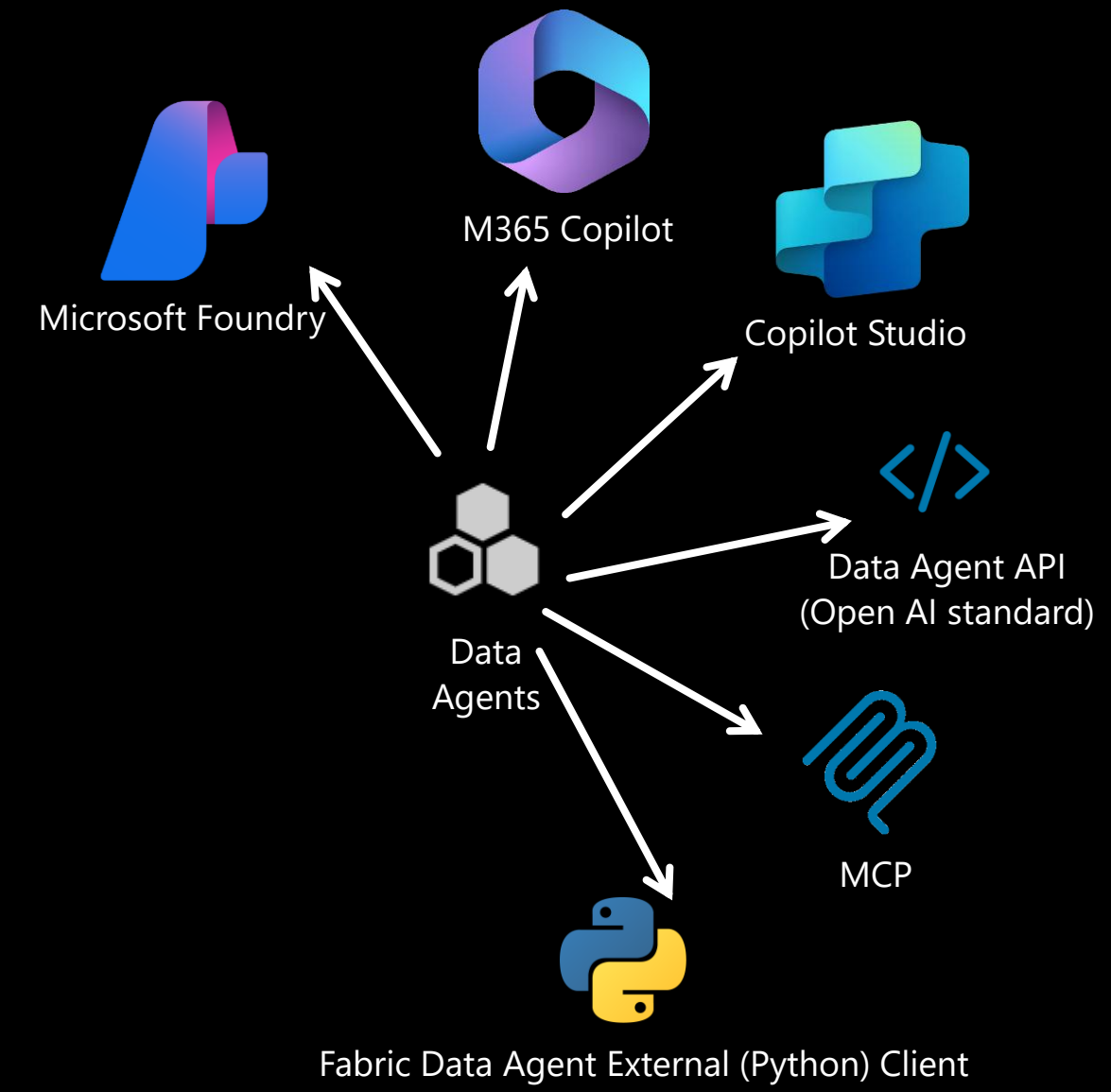
Agent Architecture

Agent Architecture

One to rule them all?



Part of a larger **ecosystem**

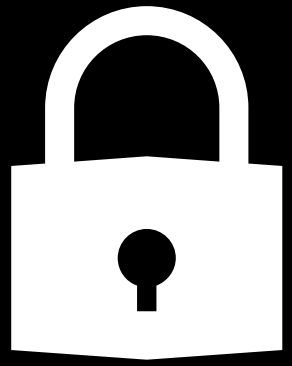


Recipe to successful implementation



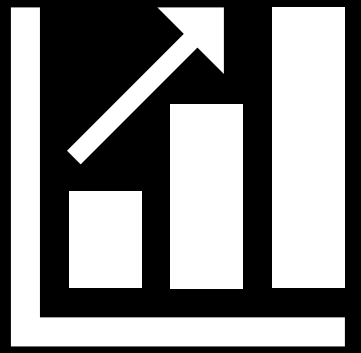
Reliable answers

Through decent preparation & (automated) testing



Secured

Inheritance and passthrough of permissions



Well adopted

Measured and tested with diagnostics

Recap & Questions

Data-Marc.com

Thanks for attending!



Marc Lelijveld
Architect & Technical Advocate
Macaw Netherlands

Microsoft® Most Valuable Professional

linkedin.com/in/MarcLelijveld

Data-Marc.com

From Source to a Working Agent in 3 Hours: Building an AI Ready Lakehouse in Microsoft Fabric

Session Evals

These slides + the code
https://github.com/marclelijveld/Slide-decks
https://github.com/marclelijveld/fabric-automation