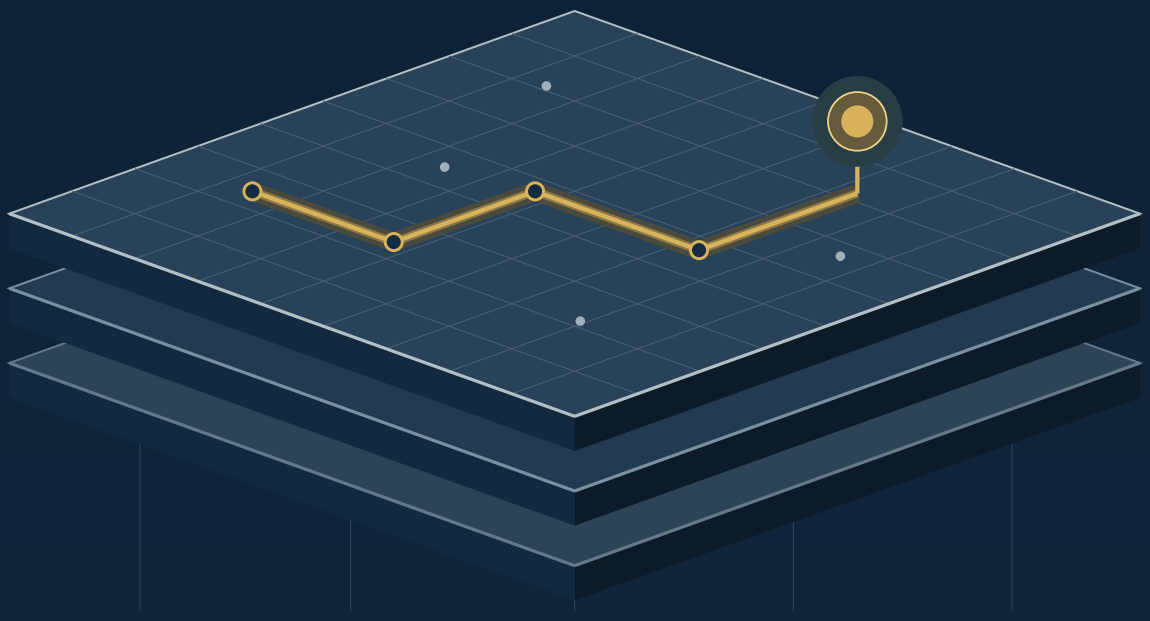
EXECUTIVE EDITION / OCTOBER 2026

From Data Foundations to Decisions

Salesforce, AWS, Databricks and Snowflake
in the architecture of an AI-enabled enterprise



ANDRÉ MAGRINI

TRUSTED DATA. EXPLICIT AUTHORITY. VERIFIABLE VALUE.

EDITORIAL NOTE

A book for people who make decisions

This book is for executives who need to connect investments in data with decisions about revenue, customer service and operations. It starts with a practical question: what will the company be able to decide better once the architecture is working?

The chapters combine data fundamentals, a view of the platforms and an execution model. Salesforce, AWS, Databricks and Snowflake appear as possible components of that architecture. A brand in a diagram represents a potential role; it does not imply that the company should purchase all four platforms.

What supports the text

Descriptions of named capabilities refer to official documentation consulted on October 19, 2026. Numbered references distinguish documented functionality from editorial guidance. Names, regional availability, licensing and commercial terms can change. A purchasing or implementation decision requires confirmation in the actual environment.

The operating scenarios illustrate a method. They are not measured customer cases or estimates of return. The ninety-day roadmap is a proposed sequence of work, subject to access, capacity and budget. First-person reflections express André Magrini's executive perspective, informed by his documented professional background and learning records.

Independence and authorship

This is an original editorial work. It does not reproduce the illustrations in the supplied reference guide or claim endorsement, coauthorship or certification by the vendors. Course certificates, academy accreditations and badges are described within their documented scope.

READING THIS BOOK

Use the references to distinguish documented capabilities from editorial guidance, and adapt the working templates to your organization. Professional contact: contact@andremagrini.com.

CONTENTS

From the business problem to the operating model

01 DECISION ECONOMICS	06
02 DATA CONTRACTS	09
03 ZERO COPY	12
04 SALESFORCE AND AWS	15
05 DATABRICKS	18
06 SNOWFLAKE	21
07 SELECTION AND COEXISTENCE	24
08 AGENTS AND TRUST	27
09 REVENUE AND INDUSTRY	30
10 COST AND VALUE	33
11 OPERATING MODEL	36
12 EVALUATION AND SCALE	39
TOOL 01	42
ABOUT THE AUTHOR	46
GLOSSARY	48
REFERENCES / 1	50

LETTER TO THE READER

Start with the decision

A company can consolidate customer data, install a modern platform and introduce an AI assistant without changing a single consequential decision. Salespeople may still prioritize accounts by instinct. Service teams may still ask for information the organization already holds. Executives may still debate which spreadsheet represents reality.

The gap between technical capability and business consequence is the subject of this book. An investment earns attention when it changes a choice: which account to serve, which risk to escalate, which commitment to make, which offer to prepare or which action to stop. A more polished answer becomes an improvement when someone can use it responsibly.

The perspective I bring

I approach data through the lens of revenue leadership. That leads me to ask who will use the information, in what situation, with what authority and under which constraints. Learning in analytics and AI expands the technical options. Learning in governance reinforces the distinction between being able to execute and being authorized to execute.

That combination shapes the chapters ahead. Data must represent a business situation. A model must support a recommendation. The process must identify who is accountable. Operations must record what happened next. When one of these connections remains implicit, the demonstration may work while the company still lacks a dependable decision process.

How to use this book

Read the opening chapters with a real decision in mind. In the platform chapters, identify the role each component would play in that decision. In the operating chapters, turn the hypothesis into a bounded experiment. The final templates help document the reasoning in a form that stands on its own.

The goal is an investment decision that can be defended, implemented and revisited. A sound architecture also helps the organization recognize when a project lacks the evidence needed to proceed.

> **CENTRAL IDEA**
> A data foundation creates value when it reduces uncertainty in a decision the company can execute and measure.

CONCEPTUAL MAP

Five layers, one line of accountability



Figure 1. Original conceptual architecture. Labels describe possible roles, not a mandatory product bundle or a universally configured integration.

The diagram begins with sources and ends with an operating consequence. Between them sit choices about identity, quality, semantics, inference and authorization. A failure in any layer may appear to the user as an incorrect AI answer, even when its cause lies far from the model.

Treat governance, observability and cost as responsibilities across the layers. The access contract must exist before a query runs. The action record must exist before execution is automated. The budget must exist before the workload is opened to use.

The feedback path matters just as much: the outcome of an action should return to the evidence base. Without that connection, an organization accumulates recommendations while losing the ability to distinguish a plausible idea from a decision that helped.

01 / DECISION ECONOMICS

Begin with a concrete choice

A data initiative becomes difficult to evaluate when its objective is “a single view of the customer.” That phrase might mean correcting records, reducing rework, improving service or preparing an offer. Each purpose requires different evidence, freshness and operating authority.

Write the decision as a sentence containing an action. For example: “Prioritize the accounts the renewal team will contact this week.” The project now has an observable choice, a user and a time window. The team can investigate what information is missing and what the current process delivers.

Identify the existing alternative

Before proposing AI, describe how the decision is made today. A report, a commercial rule, a manager's judgment or a combination of these mechanisms provides the initial reference. It may have limitations, but the proposed change must be evaluated against it.

The project must demonstrate what it adds. A model-generated account list can look sophisticated while contributing less than a rule based on contract expiration and unresolved critical tickets. Compare the same population, decision horizon and capacity constraints.

Find the bottleneck after the answer

A recommendation does not guarantee action. The team may already have more opportunities than it can address. A required discount may need approval. A customer may sit outside service coverage. Better information can reveal a need without creating the capacity to respond.

Connect the decision to the operating queue that will carry it out. Record the accountable person, the work that will be displaced and the stop condition. A useful project must fit the working day of the person receiving its output.

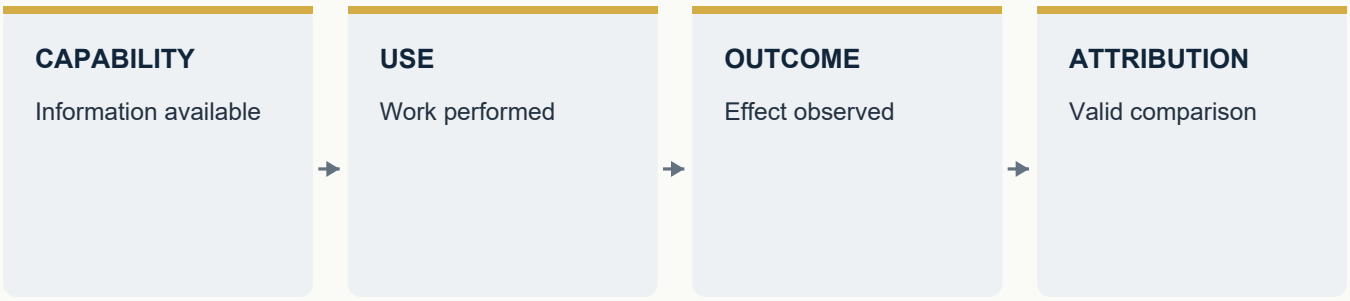
> **AUTHOR'S PERSPECTIVE**
> I would begin with a decision leadership can actually change. A platform can expand the company's capabilities. An accountable decision owner gives those capabilities a path to results.

01 / DECISION ECONOMICS

Separate efficiency, revenue and evidence

Time saved, risk reduced and revenue growth require different chains of evidence. If an assistant prepares an answer faster, that may demonstrate a task-level time saving. To claim a cost reduction, observe the effect on operations. To claim incremental revenue, use a comparison that supports attribution.

Combining these levels produces projects that appear profitable before they have been tested. The same minute saved becomes a personnel saving, added capacity and another sale. The calculation counts the benefit repeatedly without showing that those consequences occurred.



Evidence at one stage does not prove the next stage.

Figure 2. Proposed value chain. Each transition requires its own evidence. The arrows do not represent automatic conversion or measured returns.

Choose a primary metric

For account prioritization, one candidate is the outcome for accounts actually contacted within the defined window. For support, it might be correct resolution without reopening. For proposal preparation, it might be time to an approved version, alongside material errors.

Record a safeguard as well. A reduction in time should not conceal increased rework. Higher conversion should not conceal margin erosion. Define the primary measure and its safeguard before comparing system versions.

INVESTMENT QUESTION

Which outcome must change, which harm must not increase, and which observation would make the company abandon the hypothesis?

01 / DECISION ECONOMICS

Define the decision product

A decision product is a bounded operating unit: it receives evidence, provides guidance, serves a user and records a consequence. It may use a dashboard, a rule, a predictive model, an assistant or a combination of these approaches.

The label keeps the discussion concrete. “Account renewal priority” describes a deliverable. “Data-driven transformation” expresses an intention. The former makes coverage, freshness, errors and accountability discussable. The latter still needs to be translated into work.

The usage contract

Document who will see the output and what they may do with it. A prediction may support meeting preparation without authorizing automatic outreach. A summary may guide an analyst without constituting an approved contract interpretation. An alert may open an investigation without changing a customer record.

Include the context that must accompany the guidance: consulted source, evidence date, limitations, missing information and reason for the recommendation. Users must understand when the output is suitable for their work and when it requires review.

The exit condition

Projects often define entry into production while leaving exit undefined. Specify what happens when a source stops updating, performance declines, the owner changes role or costs exceed the budget. The organization may suspend automation, return to the earlier rule or narrow the population served.

Reversibility belongs in the design. A decision product can be a useful experiment even if it does not become a permanent portfolio asset. It may show that the opportunity was smaller, that the process needed to change or that a simpler solution was sufficient.

By the end of this chapter, the investment should fit on one page: decision, user, current reference, required evidence, expected benefit as a hypothesis and stop condition. If a technology map is still required to explain its value, the business definition needs further work.

02 / DATA CONTRACTS

Make customer identity meaningful

“Customer” may refer to a corporate group, a contracting company, a branch, a buyer, a distributor or an end user. Combining records without deciding which entity matters can create a view that looks consistent on screen and misrepresents the operating relationship.

Consider an industrial business selling through channels. A distributor buys parts, a producer operates the equipment, a branch receives the invoice and a service provider performs maintenance. All four may appear as contacts related to an opportunity. Their roles remain distinct.

Define entities and relationships

Begin with the object of the decision. If the purpose is preparing a service visit, the installed asset and its location may matter more than the historical buyer. If the purpose is contract renewal, the contracting entity and current term may take priority.

Create explicit relationships between these entities. An asset belongs to an installation during a defined period. A contact represents a company for a particular purpose. A channel has a bounded commercial authorization. These relationships need effective dates, provenance and conflict rules.

Identity resolution has consequences

Combining two records may expose information to someone who should not see it. Keeping them separate may duplicate outreach or distort analysis. The policy should address both errors and provide a way to review and reverse decisions.

Preserve uncertainty when the evidence is insufficient to establish a match. The record can enter an investigation queue. A documented gap is more useful than a silent association that contaminates offers, service interactions and models.

> **AUTHOR'S PERSPECTIVE**
> From the commercial perspective I propose, identity is an operating responsibility. Distinguishing who buys, who uses and who decides helps the architecture represent real relationships.

02 / DATA CONTRACTS

Define quality through intended use

A populated field may still be unsuitable for a decision. A visit date may be correct but too old to guide equipment availability. An opportunity value may be current in the CRM while representing something different from recognized revenue. Quality begins with the meaning the consumer will assign to the data.

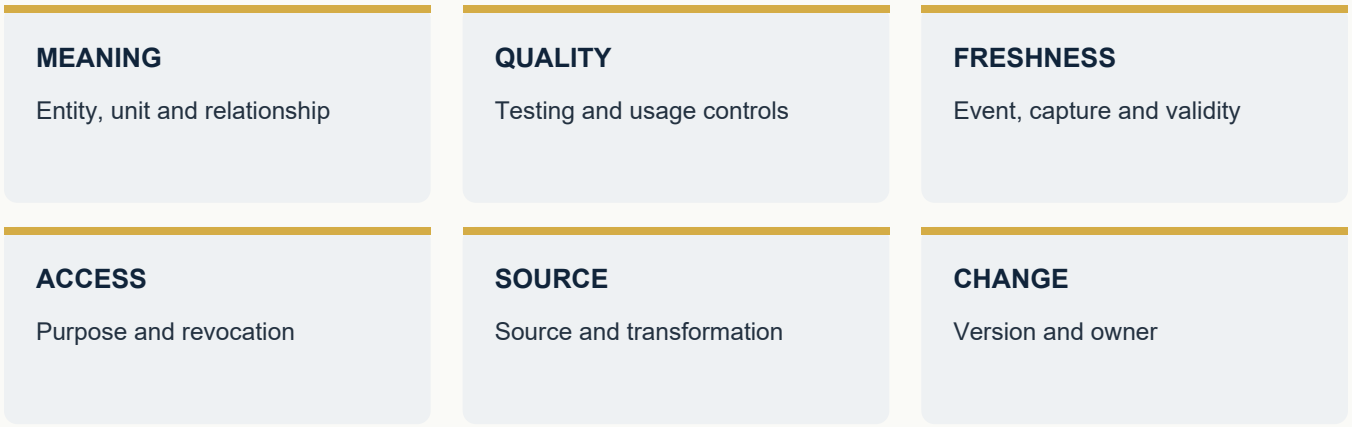


Figure 3. Minimum data-product contract. The six dimensions are editorial guidance and should be adapted to the decision and its risk.

Quality conditions that block use

Decide which problems prevent use and which allow a qualified output. An account without an assigned owner may remain in an aggregate analysis. The same gap may prevent automatic creation of a sales task. A document without a current effective date may be useful as history and unsuitable for answering a policy question.

Put that distinction into execution. When freshness exceeds the permitted window, the consumer receives an explicit unavailable state. Old information must not appear with the same assurance as a valid update.

Changes need a contract too

When a producer changes a column, the responsibility includes assessing consumers. Version the definition, communicate the change and test its interpretation. A new status code may be technically valid while completely changing the accounts selected by a model.

EXIT CRITERION

The consumer can explain the meaning, source, freshness and usage restriction of each decision-critical field, including when that information is missing.

02 / DATA CONTRACTS

Documents are data products too

PDFs, transcripts and messages carry evidence that does not fit neatly into a table. Their value depends on context. A warranty passage may belong to an older version, a particular product family or a contractual exception. Recovering it without those limits can produce a persuasive but unsuitable answer.

Maintain a record for each document with identity, source, version, effective dates, permitted audience and owner. A retrieval unit may be smaller than the document, but it needs to retain that connection. An isolated paragraph loses value if no one can establish where it came from.

Prepare content for the question

Segmentation should follow the use case. A technical manual requires preserving warnings and relationships between steps. A commercial table requires headers, units and conditions. A transcript requires identifying speakers and distinguishing a commitment from a hypothesis or objection.

Evaluate extraction as part of the product. Test tables, scanned documents, attachments and conflicting revisions. Record what the preparation stage could not read. A search tool must not present a failure to extract as proof that the original document contains no relevant information.

Updates and withdrawal

Removing an expired policy from a folder does not prove that it has stopped appearing in retrieval. The operating design must account for indexes, derived copies and references. The relevant test asks whether the withdrawn version still influences an actual user's answer.

Apply the same discipline to access. A document available in one function may be restricted in another. Allowing a search service to read a library does not authorize every user to receive every passage. The policy must reach retrieval and response generation.

With these controls, “using company knowledge” becomes a manageable proposition: a bounded, maintained and identifiable collection, authorized for the questions it is meant to answer.

03 / ZERO COPY

Map the movement behind the label

Zero copy can reduce replication in a particular integration. The term alone does not describe the entire architecture. Identify where the data remains, which mechanism accesses it, where processing occurs and which derivatives are produced.

This matters when a demonstration combines federation, a search index and a model. Table access may avoid a persistent full copy. The rest of the workflow may still create results, metadata, logs, vectors or materializations that require their own policies.

Four different choices

A federated query accesses a remote system through a defined mechanism. File sharing may let a consuming engine read storage associated with a catalog. Ingestion creates a representation at the destination. Materialization prepares a view for a workload. The choice depends on load, isolation, history and latency.

Zero ETL also requires a precise reading. Amazon Redshift's documentation describes an initial load and managed replication from source to destination. The operational work changes, while replication remains part of the integration. **[03]**

Describe the path in one sentence

Write the arrow as a statement: “Databricks reads objects shared by Data 360.” Or: “Data 360 queries an external source.” These paths have different directions, access identities and responsibilities. A generic double arrow leaves those distinctions unresolved.

Then add freshness, authentication, revocation, cost and failure behavior. The design is incomplete until the team can explain what the consumer will see when the source becomes unavailable.

> **AUTHOR'S PERSPECTIVE**
> I would treat zero copy as an engineering option to prove for each flow. An executive decision needs to identify both the replication avoided and the dependencies introduced.

03 / ZERO COPY

An integration has direction and boundaries

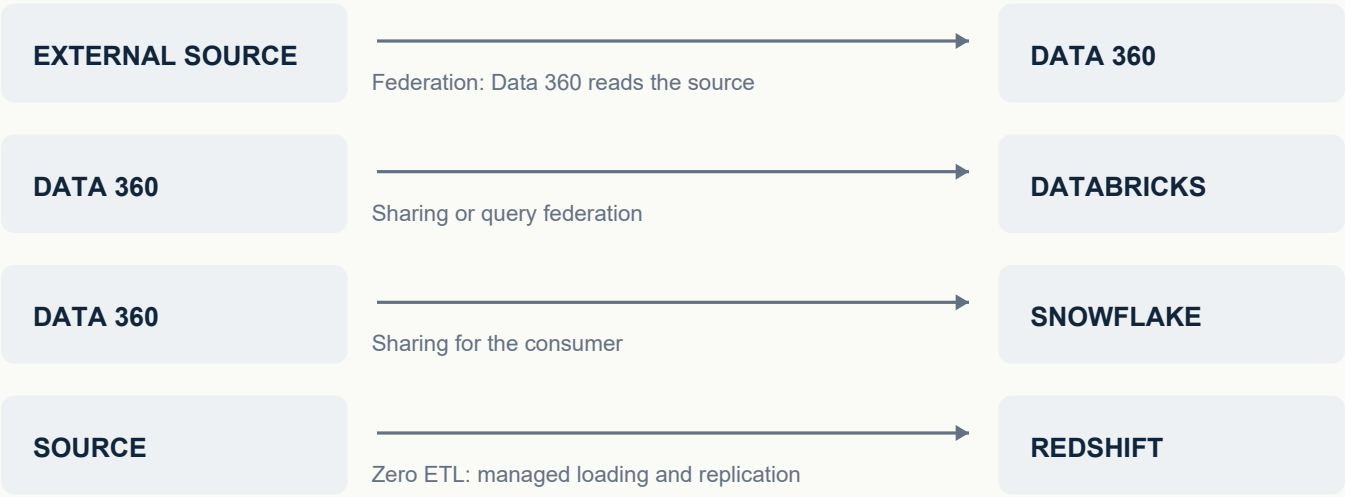


Figure 4. Conceptual access patterns. Arrows connect the data source to its consumer; they do not show query submission. Each connector requires its own configuration and prerequisites. **[03][04][05][06]**

Databricks documentation distinguishes Data 360 connectors for file sharing and query federation, alongside a Salesforce ingestion connector. File sharing reads through the sharing path, query federation submits queries to the source, and ingestion copies data. **[04]**

In the path documented by Snowflake, Salesforce data products become queryable through a Zerocopy Connector and catalog-linked databases. That describes Salesforce to Snowflake. Federation in the reverse direction through Data 360 requires its own configuration. **[05][06]**

Test the authorization boundary

Confirm which objects and rows the consuming principal receives, how revocation works and whether source permissions produce the intended behavior at the destination. Evaluate the integration using an identity representative of the actual audience, alongside the administrator who configured it.

The objective is a verifiable record of the flow. A technically connected integration can still be unsuitable for the population, purpose or freshness the decision requires.

03 / ZERO COPY

Open formats still require operating choices

Apache Iceberg is an open table format. Its documentation describes capabilities including schema and partition evolution, snapshots and reading through compatible engines. This helps separate the data format from the engine using it. [07]

That separation deserves evaluation, alongside the remaining interoperability work. Access also depends on the catalog, read and write versions, policies, storage and supported operations. Two tools recognizing a format does not prove equivalent guarantees for a particular workload.

Include an exit test

A portability experiment should attempt to recover representative tables, interpret types, reconstruct relationships and run relevant queries. Include policies, metadata and transformations in the inventory. Exporting files while losing their meaning leaves a significant part of the system behind.

Test writing authority too. A shared-read architecture does not establish that every engine can safely modify the same table. Define the write owner and compatibility conditions before enabling concurrent changes.

Understand distributed dependencies

Reduced duplication may increase dependence on source availability. The team must decide whether that trade-off is acceptable for the decision. An investigative dashboard and an operating authorization may require different behavior during an outage.

Record the deliberate choices at the end of the design. Which data will be copied for history or isolation? Which will remain at the source? Which derivatives will be retained, and when will they be removed? An architecture is easier to defend when it explains the copies it needs.

EXIT CRITERION

Every arrow has a documented source, destination, identity, operation, freshness, cost, revocation path and failure behavior. The open format is one part of that contract.

04 / SALESFORCE AND AWS

Connect context to the work

Salesforce Data 360 is the current name of the product formerly called Data Cloud. The documentation records the change in October 2025 and describes data ingestion, harmonization, unification and analysis, including identity resolution and federated access. **[02]**

For an executive, the task is to connect these capabilities to the point where work happens. Account information must reach a person or workflow for a defined purpose. Architecture becomes useful when the required context supports the preparation, guidance or execution of an authorized action.

The activation layer's role

An activation layer organizes the use of information within a journey. It should establish which information enters an interaction, how it is interpreted and which action may follow. That responsibility remains necessary when the primary data resides in another platform.

A unified profile still needs clear business definitions. A customer object can bring together valid events without distinguishing an invoiced sale from purchase intent. Meaning must travel through the integration into the user's experience.

What the reference guide contributes

The supplied guide presents Salesforce and AWS working together to connect data and support AI experiences. It discusses structured and unstructured context, sharing and model use. That provides a useful foundation for an activation discussion. It does not establish results for the company reading this book. **[01]**

The executive project must complete that foundation with its own questions: which decision will change, which component owns each function, and which benefit will be observed? An available integration path creates an option. The company's priorities determine whether that option deserves investment.

> **AUTHOR'S PERSPECTIVE**
> I would ask where information starts to change the work. That point needs an accountable person, an authorization rule and an outcome that can return to analysis.

04 / SALESFORCE AND AWS

Assign functions and accountable owners

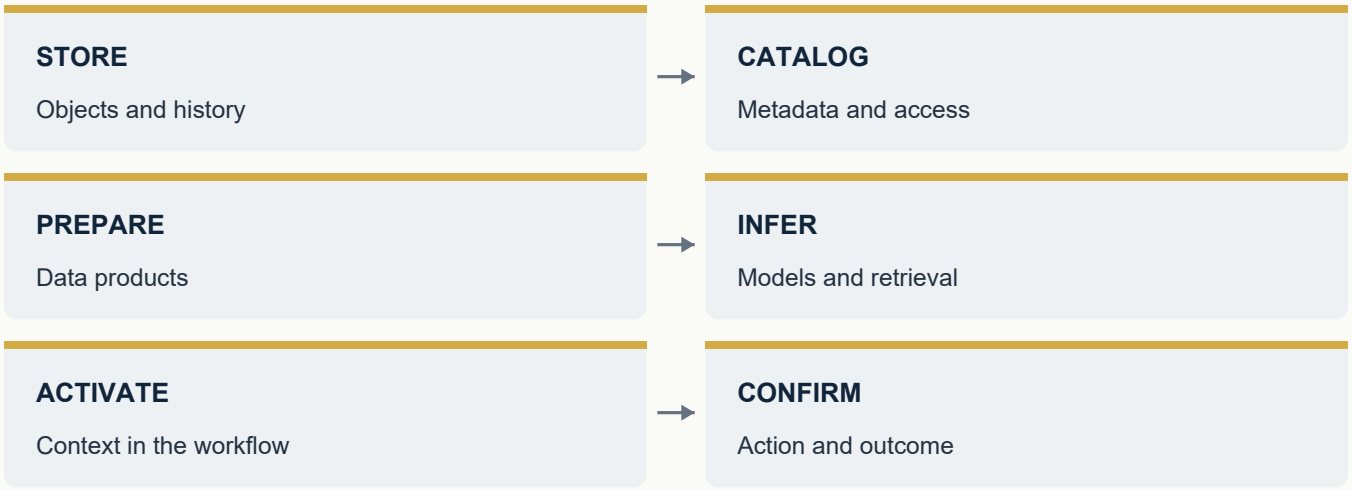


Figure 5. Proposed functional reference. Connections represent contracts to implement and validate; they do not imply automatic integration among all services.

An architecture may use AWS storage, an analytical platform to prepare data products and Salesforce for a commercial interaction. The design must name the owner of each transformation and control the versions of business rules used across components.

For retrieval-augmented generation applications, Amazon Bedrock Knowledge Bases provides mechanisms to retrieve information from sources and use it in responses, including source references. The service can form part of a RAG solution. Application quality and authorization still require testing. **[08]**

Specify each component's authority

Storage retains objects. A catalog describes assets and organizes access. Preparation creates a usable representation. A model generates or estimates. The work system manages the action. These functions may share a vendor, but their responsibilities must remain identifiable.

If two components calculate purchase propensity, establish which version guides the workflow. If an external model provides a prediction, define how it is versioned and exposed to consumers. Its appearance inside a familiar interface is not evidence of validity.

PRACTICAL DECISION

Approve the design when every function has an owner, input, output and expected failure state. The service inventory follows from that division of responsibilities.

04 / SALESFORCE AND AWS

End the integration with a verifiable action

Consider an assistant helping a service professional prepare a maintenance response. The CRM may provide the account and current interaction. A data product may identify installed equipment. A document collection may explain valid procedures. An operating system confirms availability and scheduling.

The assistant must preserve these distinctions. A manual describes a procedure; it does not confirm inventory. A ticket records a report; it does not establish a fault's cause. A prediction of need does not authorize a service commitment. The wording and available tools must reflect these boundaries.

Test the difficult conditions

Remove the availability source and observe whether the assistant acknowledges the limitation. Present a manual for another equipment family. Use an account without a proven link to the asset. Revoke a permission and confirm that the content is no longer retrieved. A useful integration behaves acceptably in these states.

The test must include the final action. Creating a draft differs from sending a confirmation. A suggested appointment differs from a recorded reservation. The application should return evidence from the accountable destination system, alongside the model's explanation.

Close the operating loop

After the interaction, record which evidence was used, which guidance was presented, who decided and what the destination system confirmed. These records support investigation and measurement of actual use. They help distinguish generated answers from resolved problems.

Expose pending status when execution cannot be confirmed. Users need to know whether to wait, repeat or escalate. The repetition mechanism must avoid creating duplicate tasks or commitments.

The connection between data and the customer journey becomes dependable when the organization can reconstruct an interaction from request to outcome, including cases where the correct outcome was to stop.

05 / DATABRICKS

A foundation for data and AI products

Databricks describes the lakehouse as an architecture combining characteristics of data lakes and data warehouses. In its implementation, Delta Lake provides a storage layer with transaction guarantees and schema controls, while Unity Catalog organizes governance for data and AI. The platform supports analytical and machine learning work. [09]

That combination may be relevant when a project needs data preparation, analysis and models within a development lifecycle. Selection still depends on the workload, available skills and existing environment. A local fit can be established without claiming universal superiority.

The business question for engineering

A maintenance product may require equipment events, service history and commercial signals. Before discussing the model, define the unit of observation. Does one row represent a machine, an occurrence or a time window? That choice affects both the prediction and its use.

Apply the same care to feature preparation. An indicator calculated after an event cannot be treated as information available before the decision. Data architecture must preserve when information existed, alongside its value.

The value of the full lifecycle

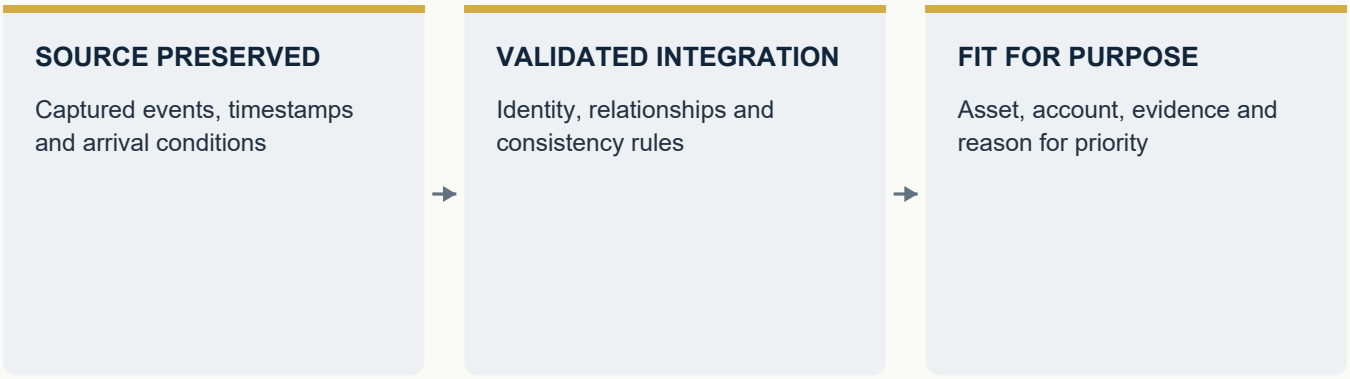
A project becomes easier to evaluate when a data version can be connected to its transformations, evaluated model and consumed output. Performance discussions become concrete: which version was tested, for which population and during which window?

That discipline must be designed and operated. A platform's ability to store artifacts does not ensure that a team uses versions consistently. Define conventions, ownership and tests that keep experimental outputs outside official workflows until accepted.

> **AUTHOR'S PERSPECTIVE**
> My documented Databricks learning includes governance, architecture, SQL/BI and commercial topics. The synthesis I propose connects those areas to a bounded decision product, with evidence and accountability at each handoff.

05 / DATABRICKS

Organize how data is refined



Refinement for the decision, with tests between layers.

Figure 6. Editorial application of layered refinement. Industrial examples are hypothetical. Layer names alone do not establish quality. **[09]**

An initial layer should retain what arrived and the conditions of its capture. An integration layer applies validity, relationship and consistency rules. A consumption layer delivers a representation suitable for a decision. The checks between these stages establish their value.

For the maintenance product, the consumption layer might present an asset, its account relationship, the latest valid evidence and a reason for priority. An event with uncertain identity can remain available for investigation without gaining the authority of the operating product.

Know the coverage of lineage

Unity Catalog can capture lineage for queries run in Databricks, including column-level relationships under documented conditions. The documentation also describes limitations and ways to represent external assets. Having the capability does not establish complete coverage of every transformation path. **[10]**

A useful test follows an important field back to its source. If an external transformation interrupts the chain, record that limitation and add the required connection. A partial graph can support investigation when its coverage is understood.

EXIT CRITERION

The consumed version can be reconstructed from recorded inputs and transformations, or its reproducibility gaps are explicit and accepted for the permitted use.

05 / DATABRICKS

Evaluate capability, integration and operations

Evaluate Databricks through an experiment representing a meaningful slice of the problem. If event preparation is the main challenge, include delayed, duplicated and incomplete events. If AI behavior is central, include ambiguous questions, missing evidence and restricted users.

A Salesforce integration test must identify its chosen path. Data 360 file sharing and query federation have separate requirements. The documentation specifies compatible configuration and privileges in Unity Catalog, among other prerequisites. Readiness for one connector does not prove readiness for the other. **[04]**

Test what the team must sustain

Observe how the team promotes versions, handles failures and explains results. A single query's execution time does not reveal the effort needed to address a schema or permission change. Include those conditions in the operating experiment.

Compare the solution with the existing process and a simple alternative. An alert rule may meet part of the need. A SQL analysis may answer a question without an agent. Additional model complexity earns its place when observed benefit and operating capacity justify it.

Expand deliberately

Once the first product works, more available data can make every adjacent use case look attractive. Before expanding, confirm the next user and the components that can actually be reused. A shared catalog does not eliminate the need for a usage contract with a new consumer.

The evaluation should produce a bounded conclusion: suitable for this workload and scope, under these conditions. That helps the company make a decision it can revisit as needs change.

The next investment depends on repeatability. The team must be able to reproduce the result, explain the cost and correct a failure without rebuilding the project for each change.

06 / SNOWFLAKE

Bring semantics closer to the question

An analytical platform can make correct tables available while users still produce conflicting answers because their definitions differ. “Customer revenue” might include invoices, orders, returns or a contract period. Before introducing a natural-language interface, the organization needs explicit business concepts.

Within Snowflake, Cortex Analyst uses Semantic Views to represent concepts, metrics and relationships that support SQL generation. This provides a concrete home for part of the semantic definition. Its quality depends on the definition and validation of the business domain. [11]

A definition needs an owner

Finance may own recognized revenue. Sales may own qualified opportunity. Operations may define resource availability. A unified interface should preserve these responsibilities and the distinctions between their measures.

Include examples of valid questions and reference answers. Test what happens when the user leaves out a time period or mixes units. Asking for clarification can be the appropriate behavior. An executable query may answer a different question from the one the user intended.

Analytics is one stage of the journey

An answer about account margin may support negotiation preparation. Changing commercial terms still requires the approval process and accountable proposal system. Analysis provides evidence; the operating design establishes authority.

This separation helps connect analytics with action through tools limited to the authorized scope. The user receives what is needed for the work, and the application retains explicit execution boundaries.

> **AUTHOR'S PERSPECTIVE**
> My Snowflake Partner Network Sales Foundation badge documents partner sales learning. Through that lens, I would ask how the platform will be used and sustained. The badge is not an architecture certification or evidence of an implementation.

06 / SNOWFLAKE

Search, AI functions and shared data

DATA AND METRICS

Semantic Views and Cortex Analyst

DOCUMENTS

Cortex Search and retrieval

CONTENT

Cortex AI Functions, by task

EVALUATED ANSWER > REVIEW OR AUTHORIZED ACTION

Shared data retains an access and meaning contract.

Figure 7. Proposed functional path for structured data and documents. Outputs need evaluation and authorization before any external action. **[05][11][12][13]**

Cortex AI Functions offer model-based operations for tasks such as classification, extraction and content processing. The documentation notes regional variation and preview status for some functions. Check the status of each resource before production use. **[12]**

Cortex Search provides hybrid search over text data and can form part of a RAG application. Relevance, freshness and response access still need evaluation within the application. **[13]**

Sharing requires interpretation

The documented Data Cloud to Snowflake path uses shared data products and catalog-linked objects. Confirm the products exposed, permissions and conditions supported by the account. Sharing does not automatically expose every object or business rule in the source. **[05]**

Test a representative query and its interpretation, including types, dates and missing values. The consumer needs to know whether the product is appropriate for the intended purpose. A working connection alone does not establish semantic equivalence between domains.

PRACTICAL DECISION

Choose the simplest path that provides consistent meaning, appropriate access and sustainable operations for the real question. Compare results using the same evidence.

06 / SNOWFLAKE

Take the experiment beyond the query

A useful Snowflake evaluation can begin with business questions and representative data. It should also cover concurrency, permissions, freshness and consumption. Separate query correctness from operating capability. A correct result under ideal conditions is one part of the decision.

If an AI interface is involved, preserve the question, semantic interpretation, executed query and presented answer. Where possible, link these records to the user and data-product version. This helps identify whether a discrepancy came from the definition, query, source or presentation.

Cost controls have different scopes

Resource monitors can track consumption and trigger warehouse suspension according to configuration. The documentation distinguishes this scope from serverless and AI service costs, for which it recommends budgets for monitoring. A warehouse monitor must not be described as a universal account spending cap. [14]

That distinction affects experiment governance. The owner must understand all consumption opened by the workload, including costs the same control cannot interrupt. A notification about projected overspend is different from preventing the next execution.

Test exit alongside entry

Try to reproduce a result, move a definition and recover a data product with its documentation. Identify what can be reused outside the current design and what would require rework. Include that observation in the cost of change.

Keep the conclusion within the evaluated scope. Success with SQL questions does not establish superiority across all AI workloads. A function suited to document classification does not determine where the entire customer journey should run.

The executive responsibility is to demand a decision grounded in actual work: what improved, which complexity was accepted and who can operate the solution when the pilot owner is unavailable?

07 / SELECTION AND COEXISTENCE

Compare workloads, skills and constraints

“Databricks or Snowflake?” can help frame an evaluation, but the question alone cannot determine a choice. The company must identify the workload, data, environment and team. Both platforms offer capabilities across analytics and AI. Fixed labels obscure that overlap. [09][12]

Begin with a hypothesis for the minimum architecture. Identify what the company already has and what the selected decision product lacks. The answer may be better use of an existing capability or a justified new component. Selection becomes meaningful once that difference is clear.

Set criteria before the demonstration

Define requirements that would eliminate an option: access restrictions, region, integration type, recovery capability or an operating limit. Then define comparative criteria: required effort, observed performance, cost, operational clarity and adaptability.

Establish how each criterion will be measured. “Simplicity” might mean faster diagnosis of an error, fewer manual release steps or broader team coverage of a routine. Without a definition, the word becomes a personal preference presented as evidence.

People are part of the architecture

Consider existing knowledge and the operating capability that must be developed. A technically suitable solution can demand a routine the team cannot maintain. The investment decision must account for that work using evidence appropriate to the actual task.

Compare the same problem across alternatives. If one demonstration uses clean data while another faces the real source, the results reflect different conditions. Preserve the population, questions and comparison constraints, recording the adaptations each path requires.

> **AUTHOR'S PERSPECTIVE**
> I would bring workload-specific evidence to a purchasing decision. The useful choice is one the company can defend for the intended use, with understood costs and responsibilities.

07 / SELECTION AND COEXISTENCE

Use a matrix without a predetermined winner

Criterion	Evidence to produce	Executive question
Correctness	Same questions and reference answers	Does the output support the right decision?
Integration	Real flows and permissions	Does data arrive with its meaning intact?
Operations	Failure, alert and recovery tests	Who sustains the service?
Economics	Observed consumption and effort	What is the cost per useful task?
Change	Versioning, export and return	What does adaptation or exit require?
Adoption	Observed use within the routine	Can the team use the output?

This is an editorial working matrix with no populated scores or results. The accountable group must set weights and approval conditions before testing. Strong averages elsewhere should not conceal failure of a mandatory requirement.

Preserve the conditions of comparison

Keep configuration, version, volume, concurrency and cost conditions with the results. The conclusion should state what was tested and what remained outside scope. A short evaluation cannot support a promise about every future workload.

Avoid counting the same benefit repeatedly. “Response time,” “productivity” and “agility” may describe the same observation. Prefer dimensions that lead to distinct decisions, and justify their weights.

Keep the unknown visible

Use “not observed” when the evidence has not been produced. An untested cell represents a dependency to resolve or explicitly accept within scope. Giving it an optimistic score or a zero distorts the meaning of the comparison.

APPROVAL CONDITION

The selected option meets mandatory requirements and has comparable evidence for the decisive criteria. Known gaps accompany the recommendation.

07 / SELECTION AND COEXISTENCE

Give coexistence an operating rationale

Running more than one platform can be a deliberate choice or an inherited condition. Either situation needs a rationale. Identify the workloads in each environment, the products crossing their boundary and the owner of the combined operation.

One design might retain a stable analytical domain while developing a new AI product. Another might keep data under a specific policy and share only an authorized representation. These are possible reasons to evaluate coexistence, rather than reasons to make it the default.

The boundary becomes a product

When platforms depend on each other, their contract needs to be operated. It includes schema, meaning, identity, freshness, permissions and error handling. Its owner needs a real service and change routine, alongside a place on the project chart.

Define where a metric is calculated and how consumers receive its definition. If each environment rebuilds the formula, the company takes on ongoing reconciliation. Include that effort in the decision, even if storage is a small part of total cost.

Test the interruption between environments

In a controlled rehearsal, suspend a representative dependency. Observe which work continues, which should stop and how the state is shown to the user. A system that silently serves old data can appear more available than it actually is for the decision.

Include the return path. If a product goes back to an earlier platform, what must be restored? How will duplicate actions be prevented? What history must be retained? Reversibility must cover operating effects as well as files and tables.

The architecture should tell a clear story: each environment has a necessary function, boundaries have accountable owners, and the additional cost is linked to an observed advantage or an explicitly unproven hypothesis.

08 / AGENTS AND TRUST

Context does not grant authority

An agent can receive evidence, select tools and prepare an action. Authorization to execute must come from an organizational policy. A retrieved document must not expand the application's privileges because it contains a persuasive instruction.

This distinction matters when the application reads messages, contracts, pages or third-party files. Their content is evidence to consult. Access rules, tool limits and approval conditions must remain under the accountable system's control.

Separate reading, proposing and executing

An initial version may retrieve information and prepare a draft. Another may create an internal task. An action changing a commitment, price or official record requires its own policy and confirmation mechanism. Each increase in authority needs tests proportionate to its possible effects.

Enforce what the agent can do through its technical identity and tools. The service executing an operation should validate arguments and authorization. A prompt asking the model to respect limits supports that control but cannot replace it.

A refusal can be useful

A useful application can explain that evidence has expired, a user lacks access or an action requires review. The handoff should identify the recipient, accompanying information and work still needed.

Evaluate more than the number of requests completed. A system can increase completion by accepting unauthorized commitments or hiding uncertainty. Value depends on completing the work it should complete and correctly recognizing cases that need another path.

> **AUTHOR'S PERSPECTIVE**
> I would treat autonomy as a bounded delegation. The company should be able to explain which action it delegated, under what evidence conditions and how that authority can be withdrawn.

08 / AGENTS AND TRUST

Make three decision states visible

EXECUTE

Sufficient evidence, valid permission and action within scope.

REVIEW

Ambiguity, conflict or a commitment requiring human judgment.

STOP

Access denied, unsuitable evidence or action outside policy.

Every output states its status and records the consequence.

Figure 8. Editorial decision policy. The three states are an operating proposal for this book, with no claim of vendor certification.

An output may be permitted, routed for review or stopped. Each state needs a comprehensible message and an operating record. An unexplained “I don't know” can abandon the user. An unverified “completed” can create false confidence.

Keep evidence with the conclusion

For an answer about commercial terms, identify the source and effective date supporting the guidance. For a proposed action, describe the intended payload and destination. The reviewer must understand the proposal before authorizing its effect.

Retain intermediate states. A submitted operation may be pending. A confirmation may arrive after a network interruption. The system must reconcile with the destination before repeating an action that could create a duplicate.

Test containment under adverse conditions

Include content that attempts to override rules, revoked access, contradictory sources and tool failures. Observe the action actually performed alongside the natural-language response. The statement “I did nothing” is not proof that the destination remained unchanged.

EXIT CRITERION

The agent respects its technically enforced scope and accurately represents execution, pending status, review and interruption, including failures of evidence or tools.

08 / AGENTS AND TRUST

Evaluate the whole task

Observe an AI application by component and as a complete workflow. Retrieval can select the wrong document. A model can misinterpret the correct document. A tool can receive unsuitable arguments. An interface can show a pending operation as complete.

An end-to-end test follows the request to its outcome and records these transitions. That prevents an improved answer from concealing a regression in authorization or execution. The aim is to locate the failure and understand its consequence for the user.

Build a reference set

Select representative tasks with expected outcomes, including cases where the correct behavior is clarification, escalation or interruption. Include the languages, access profiles, documents and data conditions that will actually occur. A set of easy questions measures a different product from the intended service.

Keep some tasks outside everyday tuning. Record versions and criteria before comparing alternatives. Changing the criteria to approve a preferred version undermines the ability to interpret progress.

Compare with simple alternatives

Use the current process, a deterministic rule or query, and a simpler search or model approach when relevant. These references reveal where additional complexity contributes to the work.

An overall average can conceal a serious error in a rare class. Examine results by task type and failure consequence. Show counts, denominators and evaluation limits. Record important behaviors that were not observed.

In monitoring, track source, policy, model and usage changes. Approval belongs to the combination tested. A material component change may require renewed evaluation before operating authority expands.

An organization gains capability when it can learn from a failure through a specific, testable hypothesis about the responsible component. “The model hallucinated” is a starting observation, not a complete diagnosis.

09 / REVENUE AND INDUSTRY

The installed-base scenario

Imagine an industrial company selling through distributors, with equipment operating at multiple sites. It wants to prepare service and aftermarket outreach with better context. This is a hypothetical scenario, not a real customer case or a result achieved by André.

Information is distributed across systems: the CRM records relationships, a transactional system records orders, a technical base identifies assets, tickets describe problems, and manuals and policies define procedures. The first decision product is a reviewable list of accounts for potential outreach.

A decision that bounds the architecture

The initial objective is to help an accountable person select situations worth investigating. The system does not diagnose faults, confirm a purchase need or send offers automatically. These limits make the initial product easier to evaluate.

Each item should show the account, related asset, reason for priority, evidence date and missing information. A user can accept the suggestion, reject it with a reason or request investigation. Recording those choices supports evaluation of the product's usefulness.

The industrial perspective

My documented professional background includes commercial leadership in industrial and agribusiness markets. The perspective I bring to this scenario looks beyond the CRM opportunity: relationships among asset, site, channel and user change the meaning of a commercial action.

The same technical signal may require different contacts under different service models. A service opportunity may fall outside the channel's coverage. Recommendation quality depends on representing those conditions as well as recognizing patterns in data.

> **SCENARIO BOUNDARY**
> No conversion rate, savings, availability or revenue is attributed to this example. The tests and indicators on the following pages are proposals for a future experiment.

09 / REVENUE AND INDUSTRY

Map the journey from signal to contact

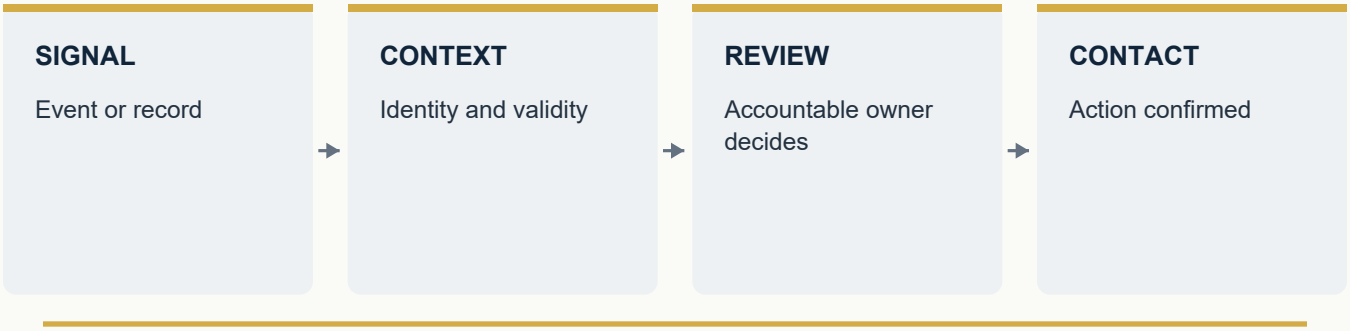


Figure 9. Hypothetical aftermarket journey. The stages describe work and accountability, without numerical results or delivery promises.

The signal enters as evidence of a situation. Preparation checks identity, validity, coverage and context. Review considers the relationship and determines whether contact is appropriate. The operating system records the authorized action.

Where the platforms may contribute

A data platform can organize assets, events and history. Data 360 may participate in composing and activating customer context. An AI service may summarize evidence and prepare an explanation. The CRM or service system records the work. These are possible functions, subject to the chosen architecture.

Databricks and Snowflake can be evaluated for the relevant data and AI workload. AWS may provide infrastructure or application services. The scenario does not require a separate component for every function or the purchase of every named platform.

Capture the return path

Record whether contact occurred, what situation was confirmed and how much work was required. Rejection because of incorrect identity is data-quality evidence. An accepted item that cannot be served may reveal a capacity constraint. Both belong in the analysis.

PRACTICAL DECISION

Expand when the list helps users work and the process can explain rejections, constraints and outcomes. The number of suggestions alone does not establish commercial value.

09 / REVENUE AND INDUSTRY

Measure usefulness before claiming revenue

Before operating the product, observe a window of the current process. Record reviewed cases, available evidence and decisions made. The window must reflect the actual work cycle and provide opportunities to observe outcomes.

The test can begin in observation mode: suggestions are generated without changing the work. A bounded population can then receive decision support. Parallel comparison, staged rollout or another evaluation design should be chosen according to the actual workflow and constraints.

Distinguish the denominators

Suggestion acceptance describes the user's assessment of the suggestions presented. Completed contact describes execution capacity. Commercial outcome describes another stage, influenced by context, offer and time. Putting these measures on the same dashboard does not make them equivalent.

Observe who was excluded as well. Strong acceptance among well-documented accounts does not establish coverage of the entire base. Missing assets or relationships in other records limit the conclusion and should appear in the report.

Investigate rejections

Ask for short operating reasons: incorrect identity, stale evidence, another accountable channel, inadequate capacity or an unsuitable opportunity. Use that classification to direct improvements. Users should be able to correct a suggestion without writing a report each time.

Preserve attribution discipline when revenue appears. A sale after contact does not establish that the system caused it. Analysis needs to consider the alternative, account selection and timing. When the comparison cannot establish incrementality, report an association and its limitation.

The product may provide value before that final proof, such as an observed reduction in manual research. Continue based on demonstrated benefit, and retain unproven benefits as hypotheses rather than adding them to realized results.

10 / COST AND VALUE

Understand the economics of useful work

Platform cost is one part of a decision product's total cost. The operation may consume storage, processing, transfer, connectors, models, search, observability and people's time. An evaluation must show which components are included and which remain outside the calculation.

A useful unit is work completed correctly within scope. In service, that may be a request resolved according to the agreed criterion. In commercial preparation, it may be a reviewed, usable proposal. For an internal agent, it may be an authorized task confirmed by the destination system.

Count the whole journey

A cheap answer can require extensive rework. A fast query can depend on expensive preparation. An inexpensive model call can generate repetitions that raise total cost. Follow the task from request to outcome, including attempts that did not succeed.

Define a window and population. Add attributable costs and divide by observed useful work. If there are no successes in the window, cost per success is undefined. Record the volume and consequences of errors as well, because a ratio alone can conceal material losses.

Separate cash savings from released capacity

A professional spending less time on research may apply that capacity to another task. That does not automatically reduce payroll expenditure. Cash savings require an observable financial change. Capacity gains need their own record and a plan for using the time released.

Apply the same distinction to revenue. Identified opportunities, issued proposals, contracted sales and cash receipts are different stages. An economic assessment should avoid adding them as separate benefits of the same work.

> **AUTHOR'S PERSPECTIVE**
> I would bring cost per useful task to the committee alongside quality, capacity and cash. Cost per call helps run the system. The investment decision needs the full consequence.

10 / COST AND VALUE

Build a cost model with visible assumptions

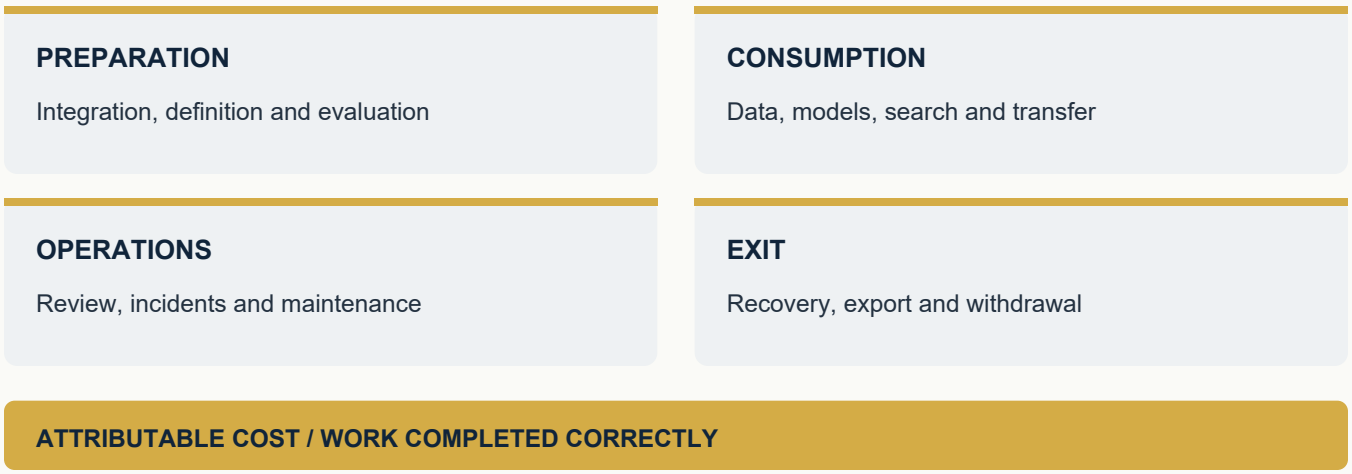


Figure 10. Editorial cost structure. No vendor prices, numerical costs or estimated returns are populated. Each component needs a measurement or an identified assumption.

The model begins with preparation and integration, continues through recurring consumption and includes operations, review and correction. Exiting the project may require work too: exporting data, rebuilding definitions, removing access and retaining necessary records.

Give every assumption a record

Identify expected quantity, unit, price source and date of the commercial terms. Record how consumption varies with volume, retrieved context, concurrency and repetition. A vendor proposal can be a valid input when its scope and conditions accompany the number.

Use scenarios defined by operating behavior, such as normal volume, concentrated demand and a degraded source. Explain which usage change produces each scenario and which limits operations will apply. Arbitrary percentage adjustments add little insight.

Include human work

A review step can be necessary and valuable. It belongs in the cost and capacity model. If every output requires inspection, observe that effort during the test and incorporate it into the volume the team can handle.

PRACTICAL DECISION

Budget approval needs a scope, consumption ceiling, alert owner and concrete containment action. Those conditions turn a spending estimate into an operating control.

10 / COST AND VALUE

Set limits the operation can enforce

Cost control starts with authorization to use resources. Define which workloads may run, at what volume and under whose responsibility. An AI application should have limits on context, tool calls, repetition and duration appropriate to the task and its risk.

Those limits need useful behavior. When the task budget is reached, the system should state its status, preserve completed work and identify the next step. An unexplained interruption can force repetition and increase cost.

An alert differs from a block

An alert communicates a condition. An execution policy may prevent new actions. A suspension may stop part of the workload. Confirm each mechanism's scope and the delay between consumption and observation. Know which costs may continue after a limit is reached.

In Snowflake, resource monitors and budget-based monitoring have different scopes. Ask the same question of every environment: which resources does this control observe, and which can it actually interrupt? Keep the claim within documented and tested behavior. [14]

Define when to stop

Set a suspension condition before scaling. It may involve cost per useful task, a serious authorization failure, inadequate freshness or human review beyond available capacity. The product's accountable owners must accept the specific condition.

An indefinite pilot also consumes resources. At the end of the evaluation window, choose an action: continue within a defined scope, correct a specific hypothesis, narrow the ambition or close the project. Attach the supporting evidence and remaining work to the decision.

An economically disciplined project can justify further learning without promising immediate returns. It must explain what was consumed, what was learned and why the next investment should be authorized.

11 / OPERATING MODEL

Bring accountability into the routine

Technical architecture needs an operating model. Name the owner of the decision, the producer of evidence, the executor of the action and the person authorized to suspend the service. One person may hold several roles, provided conflicts and available capacity remain visible.

The sponsor defines the problem and investment limit. The business owner accepts the use and monitors consequences. The data owner maintains definitions and quality. The technical team operates integrations and the application. Security and privacy functions assess relevant access and restrictions. These are proposed responsibilities, not a recommendation to hire a new team.

Reserve capacity before making a promise

Every deliverable needs a named owner and available hours. A dependency without someone able to take it remains pending. Assigning a generic job title on a plan does not establish organizational participation.

Keep a simple record: owner, activity, reserved hours, window, internal cost where known and dependency. When hours have not been estimated or accepted, record “to be defined by the owner.” The roadmap must not turn that gap into a delivery commitment.

Create a cadence for decisions

An operating review should examine exceptions, incidents, use and cost. An evolution review should decide changes in scope, version and authority. The purpose is to turn observations into bounded actions with owners and completion criteria.

Bring representative cases alongside aggregate indicators: a useful success, a rejection, a failure and a pending item. These examples help the group understand what the measures summarize and what they leave unobserved.

> **AUTHOR'S PERSPECTIVE**
> Governance learning reinforces the questions I would bring to operations: who may decide, who must be consulted and which record enables accountability afterward? Those answers must appear in daily work.

11 / OPERATING MODEL

Use ninety days as a sequence of decisions

DAYS 1 TO 30	DAYS 31 TO 60	DAYS 61 TO 90
Define and prove access	Build and evaluate	Operate within limits
Decision, data and experiment contract	Integrated slice, comparisons and failures	Real use and continuation decision

Progress depends on evidence, capacity and owner authorization.

Figure 11. Editorial ninety-day roadmap. The windows organize work; they are not a delivery forecast. Progress depends on evidence and confirmed capacity.

First window: define and prove access

Bound the decision, describe the current process and choose the initial population. Name owners and record available capacity. Confirm access and data meaning. The output is an accepted experiment contract with comparison references and stop conditions.

Second window: build and evaluate

Implement a slice of the journey with representative data. Test quality, access, failures and cost. Compare against simple references. The output is a traceable set of results, including failures and conditions that remain unobserved.

Third window: operate within limits

If evaluation supports it, introduce the product into a bounded routine. Monitor adoption, effects, incidents and review capacity. The output is a documented decision to expand, correct, narrow or close, with ownership of the next stage.

EXPLICIT DEPENDENCIES

Access, staffing capacity, budget, usage approval and process changes depend on the organization's accountable owners. Elapsed time does not replace a decision to proceed.

11 / OPERATING MODEL

Require evidence at every gate

The first gate checks the definition of the problem and its use. Is there a real decision, user, current reference and measurement approach? Have owners accepted the work and cost exposure? Until these exist, construction rests on an intention.

The second gate checks whether the implemented slice works under relevant conditions. Is the data correct for the intended use? Is access appropriate? Does the system respond acceptably to failure? Can results be linked to tested versions? A demonstration without these records leaves the evaluation incomplete.

The transition into routine use

Before operating, confirm incident handling, reversal, user communication and review capacity. The team must know which state suspends the product and who decides whether to resume. Technical functionality alone does not establish operating readiness.

The expansion gate examines actual use. Did the output help the decision? Did the review queue fit capacity? Were costs and incidents within accepted conditions? Does the additional population need new data, policies or tests?

Record decisions to stop

A suspension can protect portfolio quality. Document the reason: an unsupported hypothesis, inadequate access, cost, risk or limited capacity. Preserve the learning and state the conditions under which reconsideration would make sense.

Useful learning changes a decision, rule or next test. The record must allow someone else to understand why investment continued or stopped. Labeling a failed experiment “a learning success” is insufficient without that consequence.

An effective operating model makes these choices predictable. It reduces reliance on enthusiasm, informal availability and personal interpretations of readiness.

12 / EVALUATION AND SCALE

Approval belongs to what was observed

A pilot may provide good answers within a bounded population. The conclusion must preserve that limit. A new language, domain, access profile or document type can change the task. Even on the same platform, the company may be creating a different product.

Inventory the evaluated conditions: data, period, users, questions, actions, load and versions. Alongside them, record untested conditions that matter for the next stage. This separates confidence supported by observation from an open expectation.

Define coverage before grading

A quality assessment must state its universe. To evaluate policy answers, include current, expired, conflicting and missing versions. To evaluate execution, include success, pending status, repetition, failure and revoked permissions.

Set criteria and thresholds before reviewing results. An aggregate score must not compensate for failure of a mandatory authorization requirement. When a dimension has not been observed, record it as unknown rather than assigning a score that creates an impression of completeness.

Scale has several directions

Volume is one direction. Increasing authority, opening another domain or enabling external use may need more scrutiny than multiplying similar queries. Treat each direction as a decision with its own evidence.

Expansion is clearer when the team knows which conditions remain valid. Does the data contract still hold? Is the user profile equivalent? Does review capacity match demand? Can infrastructure support the load? The answers should justify the next bounded scope.

> **AUTHOR'S PERSPECTIVE**
> I would accept a narrower, demonstrated conclusion before a broad score without coverage. An executive decision improves when the unknown appears alongside the conditions that passed.

12 / EVALUATION AND SCALE

Observe change and consequence

Dimension	Signal to observe	Possible decision
Data	Validity, coverage and identity	Suspend part of consumption
Answer	Reference, interpretation and omissions	Reevaluate the version
Authority	Access and actual execution	Contain the tool
Operations	Pending status, repetition and recovery	Narrow scope or revert
Economics	Consumption and effort per task	Apply limits and investigate
Business	Use, outcome and safeguards	Expand, adjust or close

This is an observation map with no populated thresholds. Each product must identify critical signals, measurement methods and accountable owners. A behavior change may originate in demand or source data as well as the model.

Investigate before broadening a fix

When a failure appears, preserve the case, locate the responsible stage and test a bounded correction. A broad prompt change can obscure the original problem and affect tasks that worked. Pair the correction with affected cases and proportionate regression checks.

Maintain the decision record

For every material change, retain the reason, old and new versions, evaluation results and release decision. That record helps reconstruct the product's state when a consequence emerges days after execution.

Operations learn when incidents become retrievable evidence and specific prevention. To demonstrate improvement, show which behavior changed and under which conditions it was verified.

12 / EVALUATION AND SCALE

The executive mandate

Executives can govern investments in data and AI without taking over engineering. Their responsibility is to require a clear relationship among capability, decision and accountability. That requirement gives technical choices direction and makes their contribution assessable.

When reviewing a proposal, ask for a concrete decision, an initial population and a comparison. When reviewing an architecture, ask for data movements, owners and expected failures. When reviewing a result, ask for the observed universe and what remains unknown.

What the next conversation should establish

A useful project review should answer three questions: which work improved, what evidence supports the conclusion, and which investment decision comes next? A meeting that ends with new features but no answers has lost sight of the decision product.

Apply the same discipline to vendor selection. Companies can build useful architectures through different combinations. The executive commitment should remain with the quality of the outcome and the ability to account for it, while technical selection follows requirements and evidence.

Make the next step verifiable

Choose a decision worth improving. Complete the templates on the following pages with its owners. Identify missing evidence and the access needed for a first test. The initial delivery can demonstrate a sustainable journey without addressing the company's entire data estate.

Trusted data, evaluated models and authorized systems can expand a company's capacity to act. The executive responsibility is to keep that capability connected to worthwhile work and evidence that the decision improved.

> **TAKEAWAY**
> Leave this book with a decision contract, a bounded experiment and a way to account for outcomes. Technology selection should serve that commitment.

TOOL 01

Decision contract

Use this template before discussing vendors. Complete it with the people accountable for the decision and its operation. Fields without supporting evidence should remain explicitly pending.

Decision to improve
Which choice will be made, by whom and at what point?

Population and window
Which accounts, assets, requests or tasks are in the experiment?

Current reference
How is the choice made today, and how will it be observed?

Required evidence
Which sources are essential? What conditions prevent use?

Outcome and safeguard
What must improve, and which consequence must not worsen?

Authority
What may the system explain, prepare, record or execute?

Ownership and capacity
Who accepts the use? Who operates it? Which hours are reserved?

Investment and stop condition
What budget is authorized? What requires suspension or closure?

HOW TO COMPLETE

The business owner accepts the contract. The technical team confirms the feasibility of the bounded scope. Third-party dependencies remain pending until explicitly confirmed.

TOOL 02

Data product contract

Document the product that will be consumed. The task is to make a defined scope usable and verifiable for a purpose. It does not require consolidating every company source first.

Field	Record to complete
Product and version	Name, identifier, owner and accepted version.
Purpose and consumer	Decision served, users and permitted uses.
Entity and granularity	What each record represents and which relationships it uses.
Sources and lineage	Origin, transformation and traceability gaps.
Identity and conflict	Keys, matching rules, review and reversal.
Freshness	Event date, capture date and validity for consumption.
Blocking quality rules	Tests preventing use and behavior when they fail.
Access and retention	Authorized principal, scope, revocation and retention.
Movement and derivatives	Queries, copies, materializations, indexes and logs.
Change and recovery	Communication, compatibility, return path and owner.

Verify with the consumer

Choose a real record authorized for the test environment and walk through the contract. The consumer should explain why it belongs to the entity, when its evidence originated and which decision it may support.

Repeat the exercise with a record that fails an important condition. Observe whether the system blocks or qualifies it according to the contract. Completing the document alone does not establish that behavior.

USAGE RULE

A new purpose may require a new contract even when it uses the same table. Technical reuse does not automatically authorize another use.

TOOL 03

Evaluation and action record

Each row should represent an identifiable task that can be reconstructed. Separate the presented response from the operation actually observed at the destination.

Block	Information to record
Context	Task, population, access profile and data condition.
Versions	Source, preparation, policy, application, prompt and model.
Reference	Expected outcome, evidence and validation owner.
Alternatives	Current process and relevant simple references.
Observation	Answer, query, tool and arguments used.
Authority	Permitted, review required, stopped or pending.
Consequence	Action confirmation, failure, repetition or no effect.
Quality and cost	Applied criteria, time, consumption and human review.
Decision	Accept, correct, narrow scope or suspend, with a reason.

Cases that must remain visible

Include expired sources, users without permission, ambiguous entities, contradictory documents, improper instructions within consulted material, unavailable tools and delayed confirmations. Adapt the set to the product's risk and purpose.

Interpreting results

Show total tasks, counts by class and conditions not observed. A rate without a denominator and population cannot establish coverage. Keep critical failures visible even when most tasks pass.

USAGE RULE

Define criteria and thresholds before evaluation. An untested condition remains unknown. It does not receive a score because a nearby condition passed.

TOOL 04

Economic record and gate decision

Make scope, consumption, capacity and benefit explicit. The fields below contain no prefilled prices or returns. Populate them with measurements or identified assumptions.

Component	Evidence source
Preparation and integration	Activities, owners, accepted hours and known costs.
Recurring consumption	Resource, unit, quantity, price, terms and date.
Operations and review	Observed time, exceptions and available capacity.
Useful work	Completion criterion, volume and observation window.
Realized benefit	Observed effect and attribution method, where supported.
Open hypotheses	Unproven expected benefits and the next test.
Safeguards	Accepted limits for quality, access, margin or rework.
Exit and reversal	Data, definitions, effects and recovery effort.

Relationships to calculate

Cost per useful task relates attributable cost in the window to work completed correctly in that same window. Keep the distinction between released capacity and cash savings explicit. If calculating return, state the period, included costs and benefits with supported attribution.

Minimum gate record

Record the decision, authorized scope, owner, budget, confirmed capacity and next review. Mark open dependencies. If a gate requires correction, specify the behavior to prove before resuming, without committing an owner to an unaccepted deadline.

USAGE RULE

This record supports a management decision. When benefit or cost evidence is insufficient, record the uncertainty and bound the next experiment.

ABOUT THE AUTHOR

A perspective across revenue, data and governance

André Magrini combines an executive background in revenue and industrial markets with continuing education in data, artificial intelligence and governance. This book connects those areas to examine how companies turn information into responsible work.

His professional background includes industrial and agribusiness markets. The perspective proposed here incorporates questions those environments make concrete: who buys, who uses, which asset is involved, who serves the customer and which commitments the organization can fulfill.

What the combination adds

The revenue lens requires a clear economic consequence for architecture. Learning in analytics requires a distinction between observed results and hypotheses. Governance brings authority, accountability and stop conditions into the design.

These perspectives meet in an editorial conviction: decisions come before platforms, and responsibility follows action. The chapters show how to apply that conviction across companies with different operating contexts.

How this perspective was developed

The personal reflections connect documented areas of learning with an executive perspective on decisions and accountability. Courses and credentials retain their stated scope, and the industrial scenario remains hypothetical.

Credentials in this book do not establish product performance or endorsement by educational institutions or vendors. The following selection explains the foundations of the perspective and the scope of each record.

CONTACT
contact@andremagrini.com • andremagrini.com

SELECTED LEARNING

What the records establish

Area	Record and scope
Snowflake	Snowflake Partner Sales Foundation Accreditation. November 2025 completion record and partner sales learning badge. This is not SnowPro.
Data governance	Databricks Academy Accreditation: Data Governance Fundamentals; Partner Training: Data & AI Governance with Unity Catalog. October 2026 records.
Architecture and operations	Databricks Academy Accreditation: AWS Databricks Platform Architect and Platform Administrator. Academy accreditations recorded in October 2026.
SQL and SAP	Get Started with SQL Analytics and BI on Databricks; Databricks Selling Foundations for SAP. Local certificates dated October 14, 2026.
AI and agents	Databricks AI Agent Fundamentals, Generative AI Fundamentals and Prompt Engineering Fundamentals. Courses and academy accreditations in the consulted records.
Data and experimentation	USP/Esalq extension courses in Growth Intelligence: growth strategy with data, AI and experimentation; and Introduction to Machine Learning with Python. Certificates dated September and March 2026. Course names translated here.
Corporate governance	Wharton Executive Education: Corporate Governance. March 2023 completion certificate. Exemplary Assignment recognizes an activity within the program; it is not another course.

This selection uses local certificates and the professional credential inventory consulted for this edition. Courses, badges, accreditations and professional certifications have different scopes. None establishes an unverified academic degree, vendor authority or completed implementation.

Selected public links appear in the references. Complete personal documents and identifiers are not reproduced. The selection is documentary; a new issuer-by-issuer authenticity audit was not performed.

GLOSSARY

Terms for an executive conversation

Activation
Using information in a journey or routine to guide or execute work. It requires a purpose, access and an accountable owner.

Catalog
A layer for organizing and discovering assets and metadata. Its functions vary by platform.

Data contract
An agreed definition of a data product's meaning, quality, freshness, access, responsibility and change process.

Data 360
The current name of Salesforce's former Data Cloud product. Older sources may retain the previous name. **[02]**

Federation
A mechanism for querying or accessing another source without assuming persistent full ingestion at the destination. The implementation depends on the connector.

Grounding
Connecting a response to specific evidence. Evidence can be present while the interpretation remains incorrect.

Lakehouse
An architectural pattern combining characteristics of data lakes and data warehouses. Its value depends on implementation and use. **[09]**

Lineage
Relationships that help trace origin, transformation and consumption. Coverage must be understood.

GLOSSARY

Terms that need explicit boundaries

Decision product
The operating unit proposed in this book: it uses evidence, supports a choice, serves a user and records consequences.

RAG
Retrieval-augmented generation. An application retrieves information to support a model response. This is different from training the model on the entire source collection.

Semantics
The definitions of entities, measures, relationships and periods used to interpret information.

Useful work
A task completed correctly under defined criteria and scope, with an observable consequence where applicable.

Zero copy
A property of an access path that avoids a particular replication. Describe it per flow, without assuming the absence of every derivative.

Zero ETL
A label used for integrations managing part of data movement and preparation. The Redshift example consulted here includes replication. **[03]**

Three evidence states

MEASURED means an observation exists with its conditions, date and scope recorded. INFERRED means a conclusion depends on assumptions or an interpretation that must be explicit. UNKNOWN means the available observation or evidence does not support a conclusion.

These states preserve what an experiment actually demonstrated. Each label needs a record that lets the reader understand why the conclusion received that status.

REFERENCES / 1

Foundations and integrations

Technical sources were consulted on October 19, 2026. Reference numbers identify passages supporting descriptions of capabilities. The other models, diagrams and criteria are original editorial synthesis.

[01] Salesforce. Data Foundation for the Age of AI: Unifying AWS and Salesforce Data for a Strong Agentic AI Foundation.
A twelve-page guide from 2025, supplied by the reader. Used as thematic inspiration and identified by its internal title, without reproducing text, artwork or customer results.

[02] Salesforce Help. About Salesforce Data 360.
Naming, harmonization and unification. The documentation records the Data Cloud to Data 360 name change on October 14, 2025.

[03] AWS. Zero-ETL integrations, Amazon Redshift.
Managed integration scope, initial loading and replication. Distinct from an absence of copying.

[04] Databricks. Lakehouse Federation for Salesforce Data 360 File Sharing.
Distinctions among file sharing, query federation and ingestion. Requirements depend on the selected path.

[05] Snowflake. About Salesforce Data Cloud and Snowflake.
Salesforce to Snowflake, Zerocopy Connector and catalog-linked objects. Confirm regional and account conditions.

[06] Salesforce Trailhead. Explore Use Cases for Zero Copy Data Federation.
Federation of external sources into Data 360. The reverse sharing path has a separate configuration.

[07] Apache Iceberg. Introduction.
Open table format, evolution and snapshots. Current documentation does not establish compatibility across every implementation.

REFERENCES / 2

Platforms, AI and consumption

[08] AWS. Retrieve data and generate AI responses with Amazon Bedrock Knowledge Bases.
Retrieval for AI applications and source references. The implemented solution requires access and quality testing.

[09] Databricks. What is a data lakehouse?
Lakehouse, Delta Lake, Unity Catalog and data refinement. Architecture and features do not establish economic outcomes.

[10] Databricks. Lineage in Unity Catalog.
Lineage capture, requirements and limitations. External asset coverage needs configuration and verification.

[11] Snowflake. Cortex Analyst.
Semantic Views representing business concepts and supporting SQL generation.

[12] Snowflake. Cortex AI Functions, including LLM functions.
AI functions and availability conditions. Check the status of each resource.

[13] Snowflake. Cortex Search.
Hybrid text search and use in RAG applications.

[14] Snowflake. Working with resource monitors.
Warehouse monitoring and suspension scope, distinct from serverless and AI service costs.

REFERENCES / 3

Learning records and evidence limits

The records below support the selected learning background and development of the authorial perspective. They do not establish product performance, customer results or authorization to speak for vendors.

[15] Databricks. Data Governance Fundamentals, André Magrini's record.
Public link in the consulted local inventory. Academy Accreditation.

[16] Databricks. Data & AI Governance with Unity Catalog, André Magrini's record.
Public link in the consulted local inventory. Partner Training.

[17] Databricks. AWS Databricks Platform Architect, André Magrini's record.
Public link in the consulted local inventory. Academy Accreditation, without equivalence to a professional examination.

[18] Databricks. Get Started with SQL Analytics and BI, André Magrini's record.
Inventory link, supported by a local certificate dated October 14, 2026.

[19] Snowflake. Partner Sales Foundation Accreditation, André Magrini's record.
Inventory link, supported by a local completion record and a badge-wallet image.

[20] Wharton Executive Education. Corporate Governance, André Magrini's record.
Program certificate. The Exemplary Assignment activity badge is not counted as a second program.

Local supporting records include Databricks SAP and SQL/BI certificates, USP/Esalq extension certificates in Growth Intelligence and Introduction to Machine Learning with Python, the credential inventory and professional profile. The private editorial record retains paths and hashes. No complete personal document is embedded in the book.

ANDRÉ MAGRINI

A trusted foundation starts with a question that matters

Which decision does your company need to improve, and what evidence will show the improvement?

Use that question to choose the first scope, assign accountability and direct investment. The architecture should make the answer clearer and the action more responsible.

contact@andremagrini.com

andremagrini.com