Architecting a Data Mesh with Amazon SageMaker Unified Studio

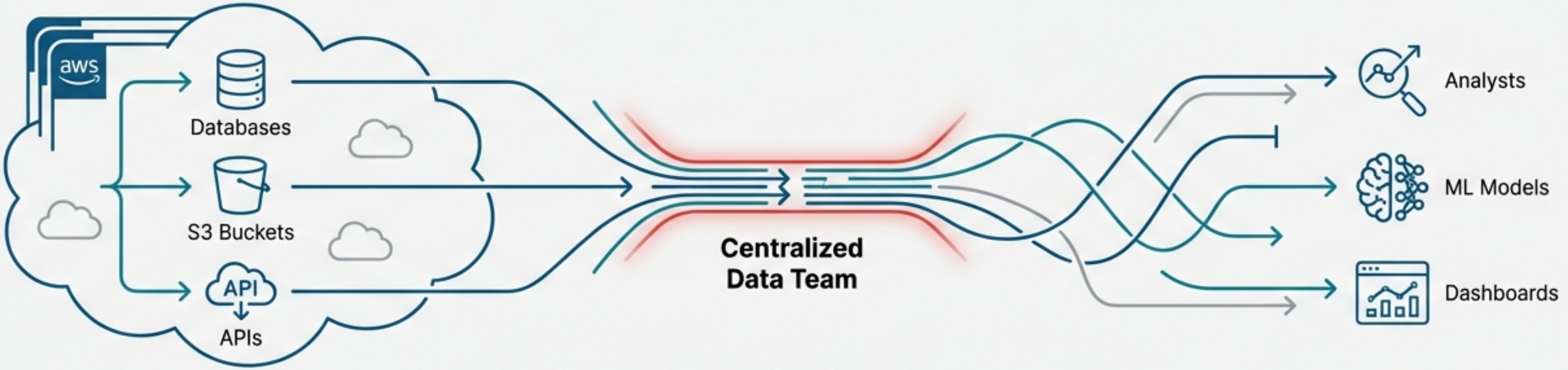
A Blueprint for Multi-Account Data Producers and Consumers

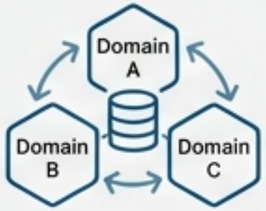
This deck outlines a practical, repeatable blueprint for building a decentralized data platform using the native capabilities of SMUS.

The Modern Data Challenge: Scaling Across Boundaries

Problem Statement: As organizations grow, data becomes siloed across dozens of AWS accounts and business domains. Centralized data teams become bottlenecks, hindering agility and innovation.

Solution Intro: The Data Mesh paradigm offers a solution by decentralizing data ownership and treating data as a product.





Decentralized Domain Ownership

Teams own their data end-to-end.

Data as a Product

Data is treated as a discoverable, trustworthy, and valuable product.

Self-Serve Data Platform

A central platform empowers domains to manage their data products independently.

Federated Computational Governance

A set of global rules enforced across a decentralized ecosystem.

The SMUS Blueprint for a Data Mesh

Amazon SageMaker Unified Studio provides the foundational components to implement all four data mesh principles natively, enabling a robust, multi-account producer-consumer model.

This architecture will be built out slide-by-slide.

The Foundation: A Central Domain Spanning Your Organization

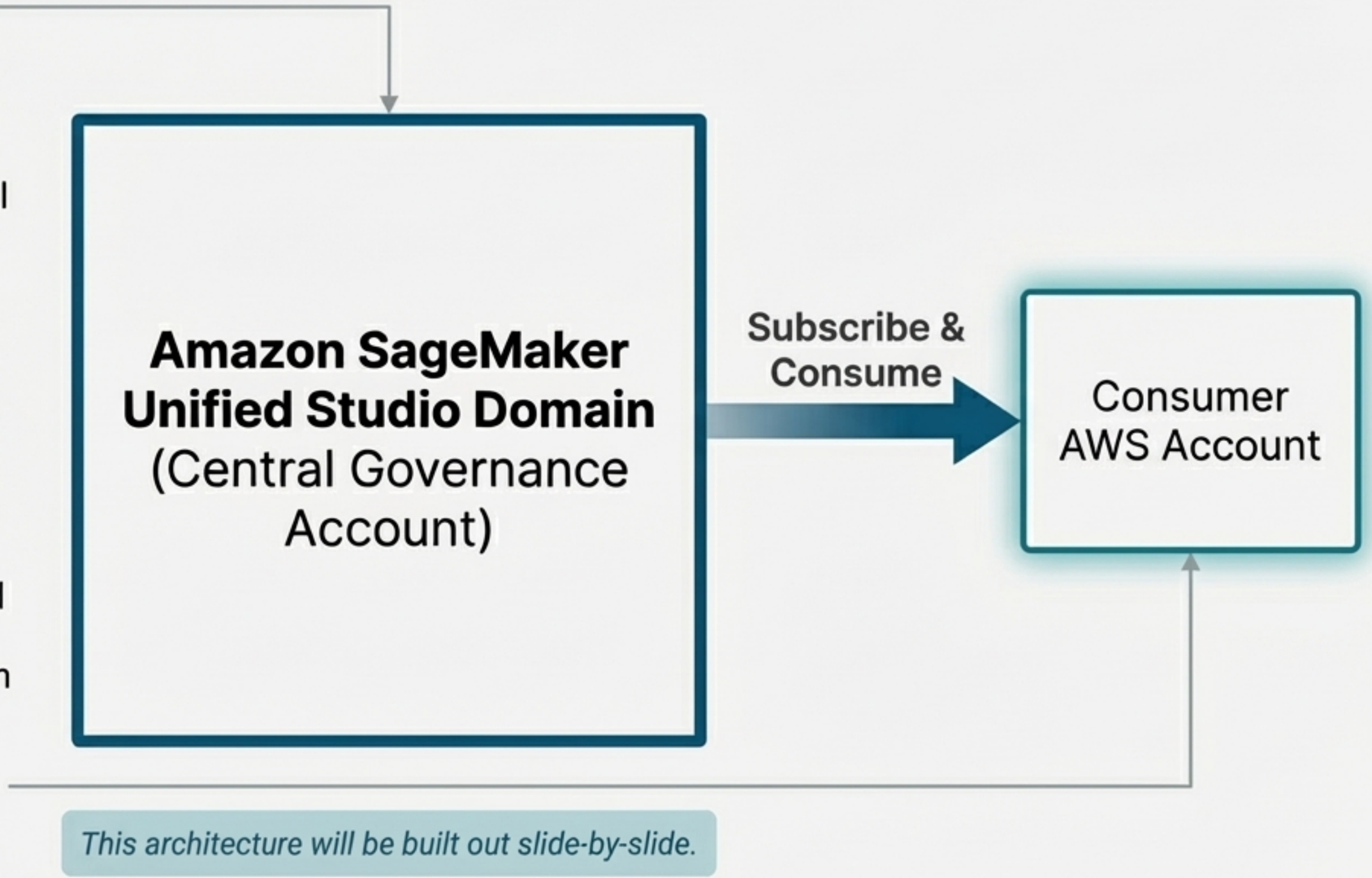
Core Concept 1: The Domain

Core Concept 1: The Domain

In SMUS, a **Domain** is the organizing entity connecting your assets, users, and their projects. It acts as the central hub for your data mesh, even when producers and consumers reside in different AWS accounts.

Core Concept 2: Associated Accounts

The key to a multi-account mesh is **Associated Accounts**. This SMUS feature allows you to link producer and consumer AWS accounts to the central domain. This enables data from other AWS accounts to be published into the central **Amazon SageMaker Catalog** and allows projects to work with data across account boundaries.



This architecture will be built out slide-by-slide.

From Principle to Practice: How SMUS Implements Data Mesh

Decentralized Domain Ownership

Teams own their data.

Projects. A project is a permission and collaboration boundary, giving a domain team its own workspace to manage code, artifacts, and data assets.

Data as a Product

Data is discoverable and addressable.

Amazon SageMaker Catalog & Assets. Producers publish **Assets** (tables, files, models) to a central, domain-wide **Catalog**, complete with business metadata.

Self-Serve Data Platform

Empower domains with tools.

Project Profiles & Blueprints. Platform admins define reusable **Blueprints** (e.g., for SQL Analytics or ML) that provision all necessary AWS resources for a new project on-demand.

Federated Computational Governance

Global rules, local execution.

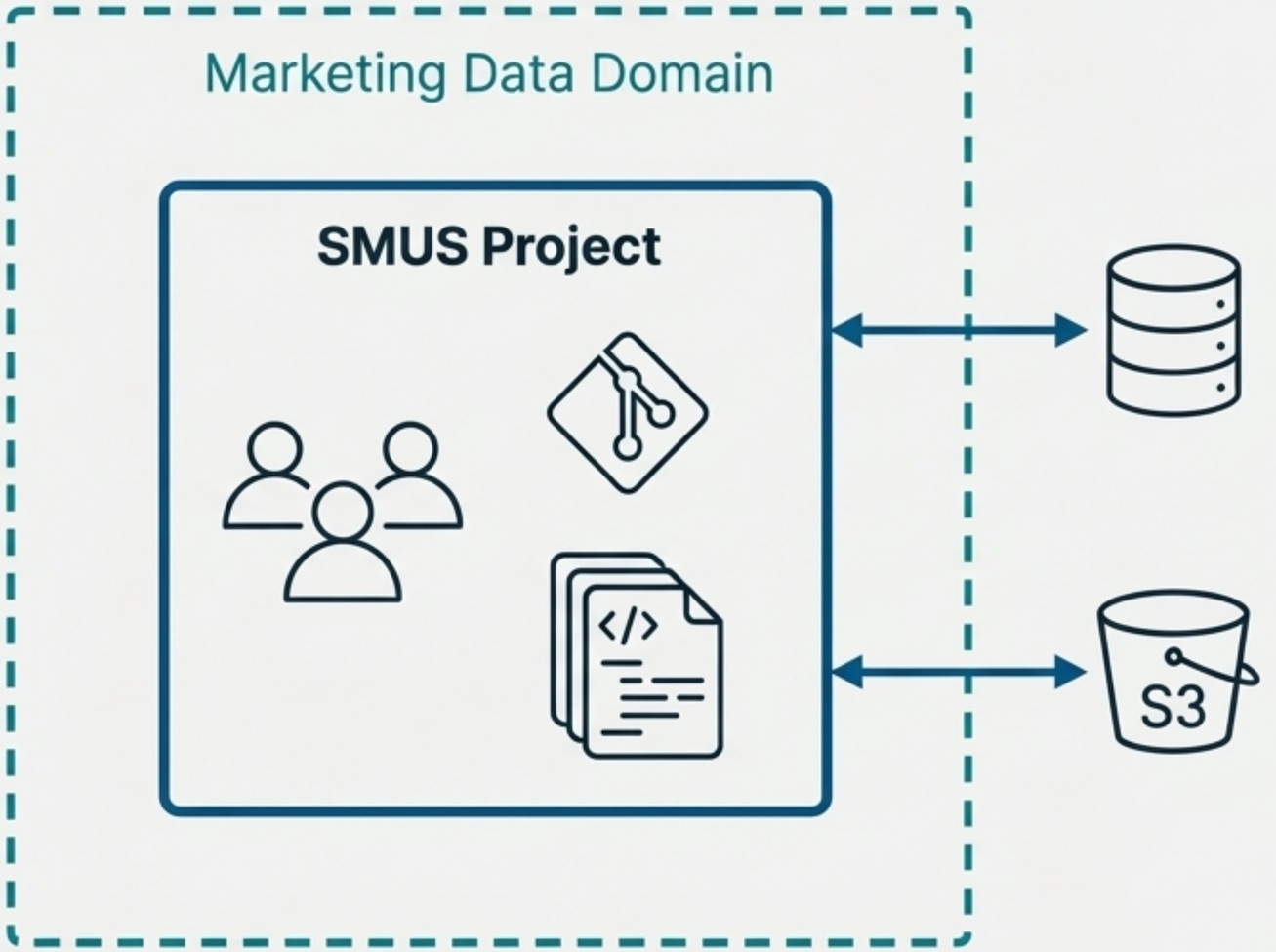
Subscriptions & Authorization Policies. Access is governed via a **Subscription** workflow. The data owner approves or denies requests, enforcing access policies at the source.

The Producer’s Journey: Building and Publishing a Data Product

Step 1: Establishing Domain Ownership with SMUS Projects

A **Project** is the mechanism by which users organize their work. For a data mesh, a project represents a single data domain’s workspace.

1. **Business Context**: Provides auditability for the work being performed.
2. **Collaboration Boundary**: A shared space with a dedicated Git repository (AWS CodeCommit by default) for version control.
3. **Permissions Boundary**: Gives project members access to all project artifacts and configured data/compute permissions.



Step 2: Creating the Data Asset

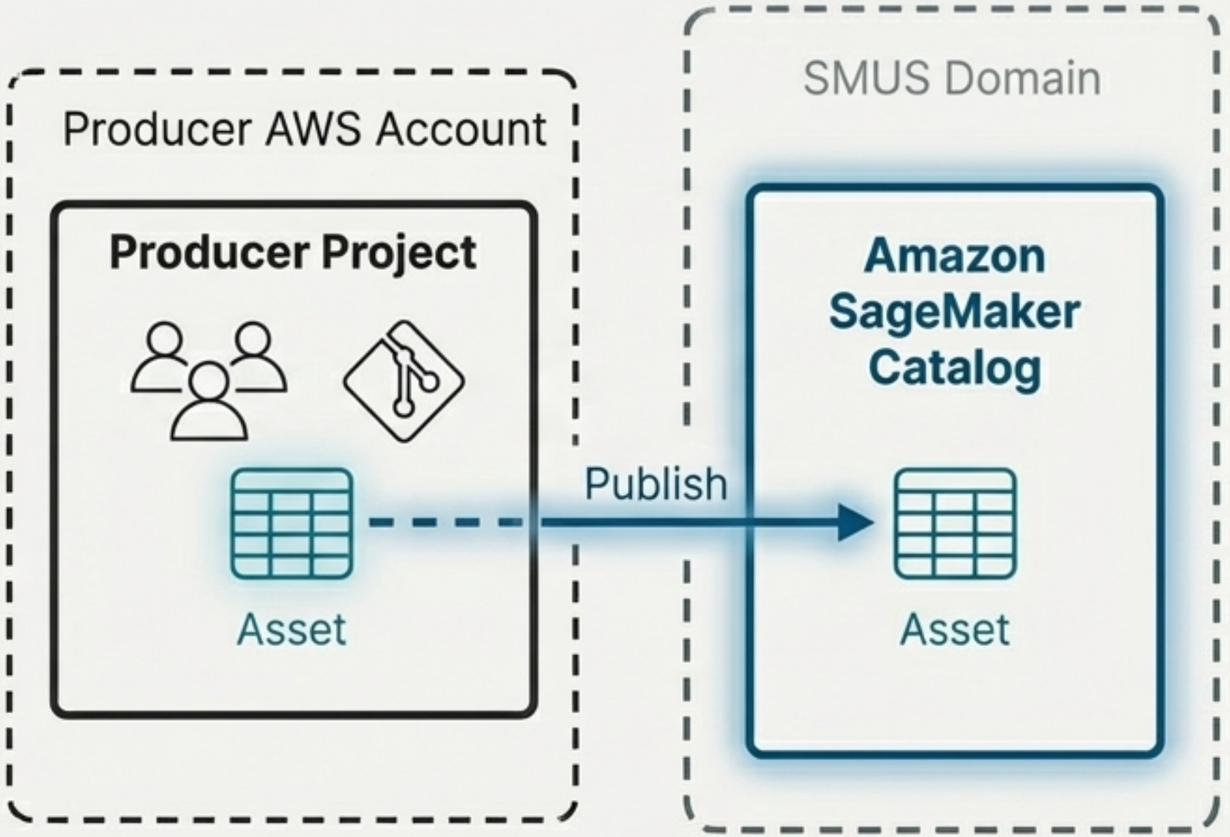
Within their project, a producer team creates a data **Asset.** An asset is a single physical data object, like an AWS Glue table or an Amazon S3 object collection.

3. Curation

- ✓ Adds business-friendly names and descriptions.
- ✓ Associates terms from a **Business Glossary**.
- ✓ Fills out required **Metadata Forms**.

Step 3: Publishing to the Amazon SageMaker Catalog

The **Amazon SageMaker Catalog** is the central, domain-wide marketplace for all published data assets. Publishing an asset from a project's inventory makes it discoverable by by all other projects in the domain, including those in associated accounts.



Key Features of the Catalog

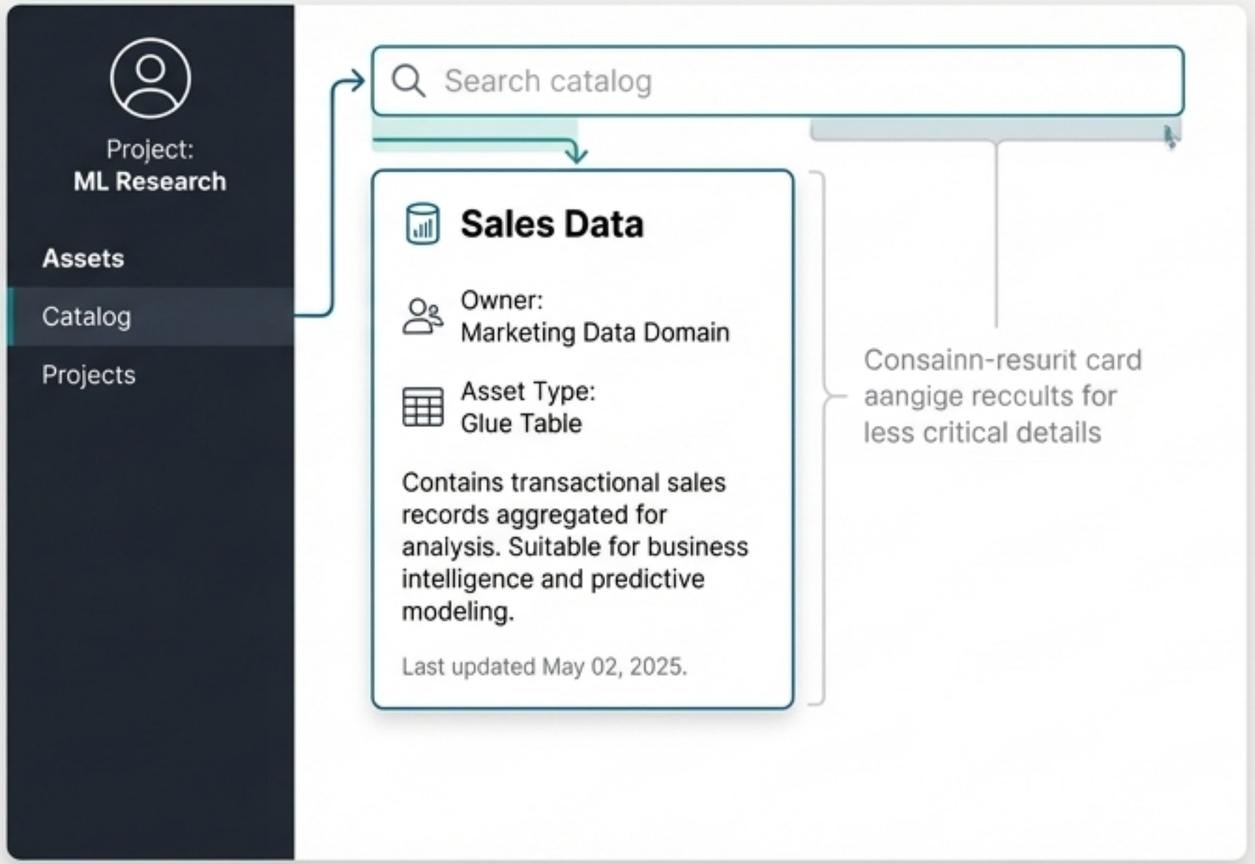
- **Cross-Account Discovery**: Enables discovery that crosses AWS account and Region boundaries.
- **Ownership**: Every asset in the catalog has an owner project (the producer) which controls access policies.
- **Versioning**: Only the latest version of an asset can be published, ensuring consumers find the most current data.

The Consumer's Journey: Discovering and Accessing Data

Step 1: Discovering Data Products in the Catalog

User Experience: A data consumer, working in their own project (e.g., 'ML Research'), can now search or browse the unified SageMaker Catalog.

How it Works: Consumers can search by asset name, metadata, or associated glossary terms. The catalog provides a rich view of the asset's business and technical metadata, its owner, and its revision history, allowing the consumer to assess its fitness for their use case before requesting access.



Step 2: Requesting Access via a Subscription

Core Mechanism:

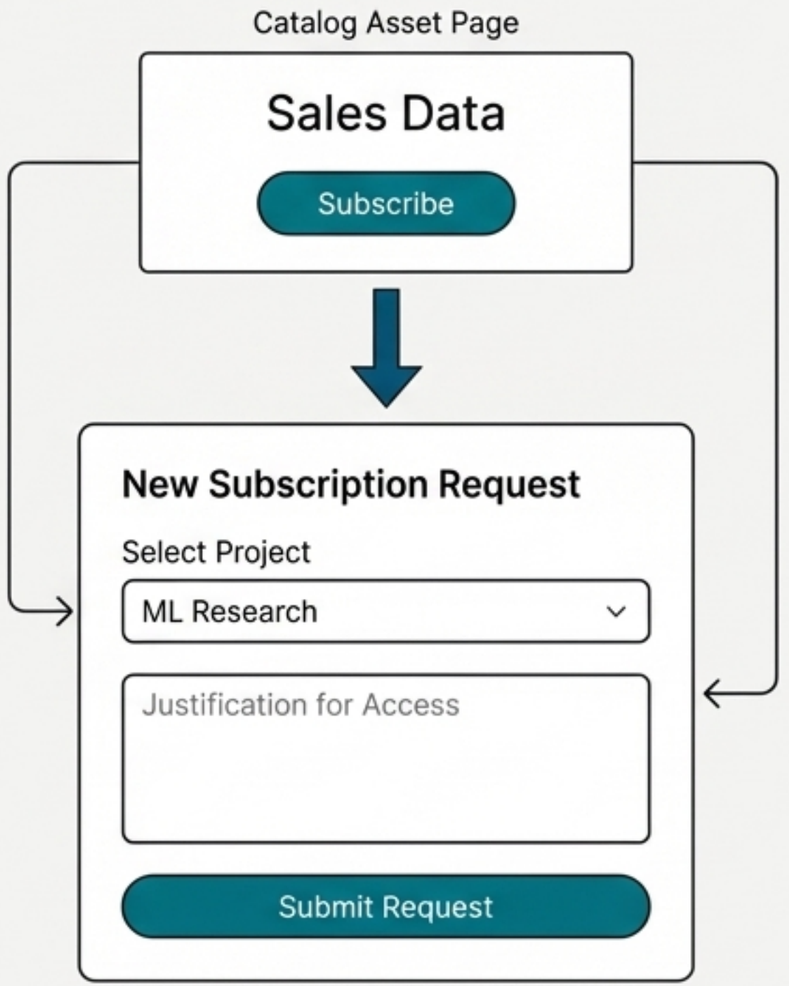
Access is not automatic. The consumer project must formally request access by creating a Subscription Request.

Process:

1. The consumer selects the desired asset in the catalog and clicks “Subscribe.”
2. They must specify which of their projects needs access.
3. Crucially, they provide a justification explaining their business need for the data.

Governance Insight:

This workflow creates a formal, auditable record of every data access request across the organization.



Federated Governance in Action: The Approval Workflow

Producer's Role

The **owner project** (the producer) receives the subscription request. Project owners have the authority to approve or reject it.

Granular Control

When approving, the producer can grant full access or apply **fine-grained access controls**:

- **Row Filters**: Restrict access to specific rows based on defined criteria (e.g., `` `region = 'Europe'` ``)
- **Column Filters**: Limit access to specific columns to exclude sensitive data (e.g., hide PII columns)

Consumer View:
ML Research Project

Producer View:
Marketing Data Domain

Data Mesh Principle: Federated Computational Governance

The domain that owns the data product computationally enforces access policies for consumers, balancing central rules with local domain context.

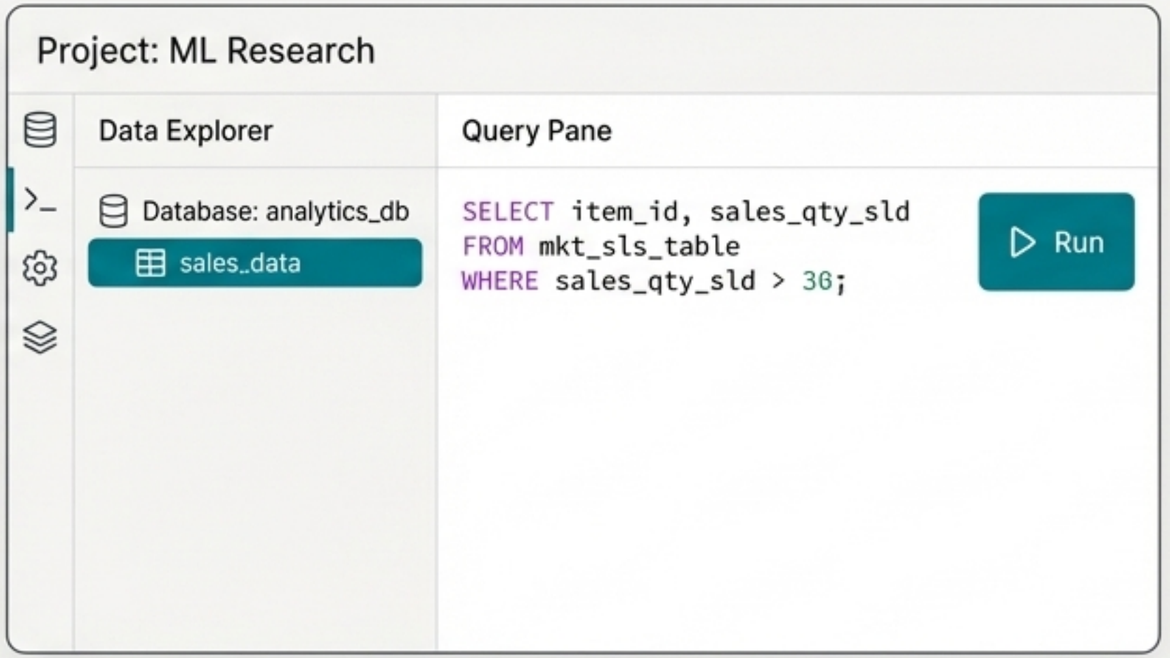
Step 3: Consuming Granted Data in a Project Workspace

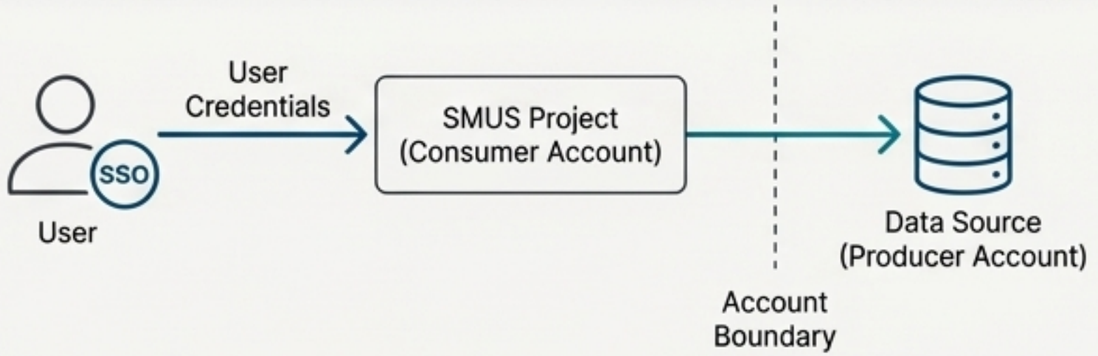
Access Granted

Once the subscription is approved, the data asset appears in the consumer's project. They can now query and analyze it using project tools like the Query Editor or JupyterLab.

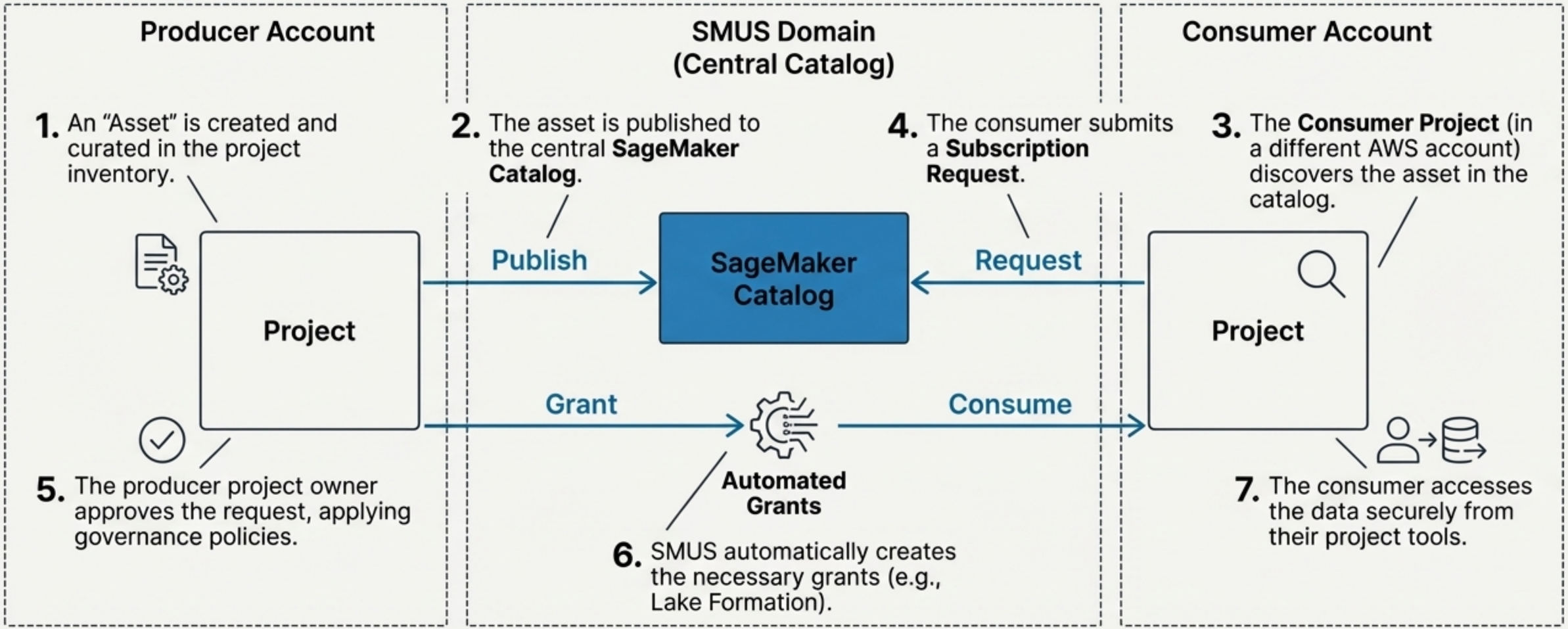
Seamless & Secure Access

- For managed sources like AWS Glue and Redshift, SMUS handles the underlying AWS Lake Formation or Redshift Datashare grants automatically.
- Trusted Identity Propagation (TIP) ensures the consumer's own SSO credentials are used when querying the data source. This allows for end-to-end user identity-based authorization, even across accounts.





The Complete Blueprint: Data Product Lifecycle in SMUS



This end-to-end workflow, orchestrated by SMUS, provides a scalable and governed framework for decentralized data sharing.

The Self-Serve Platform: Empowering Domains with Blueprints

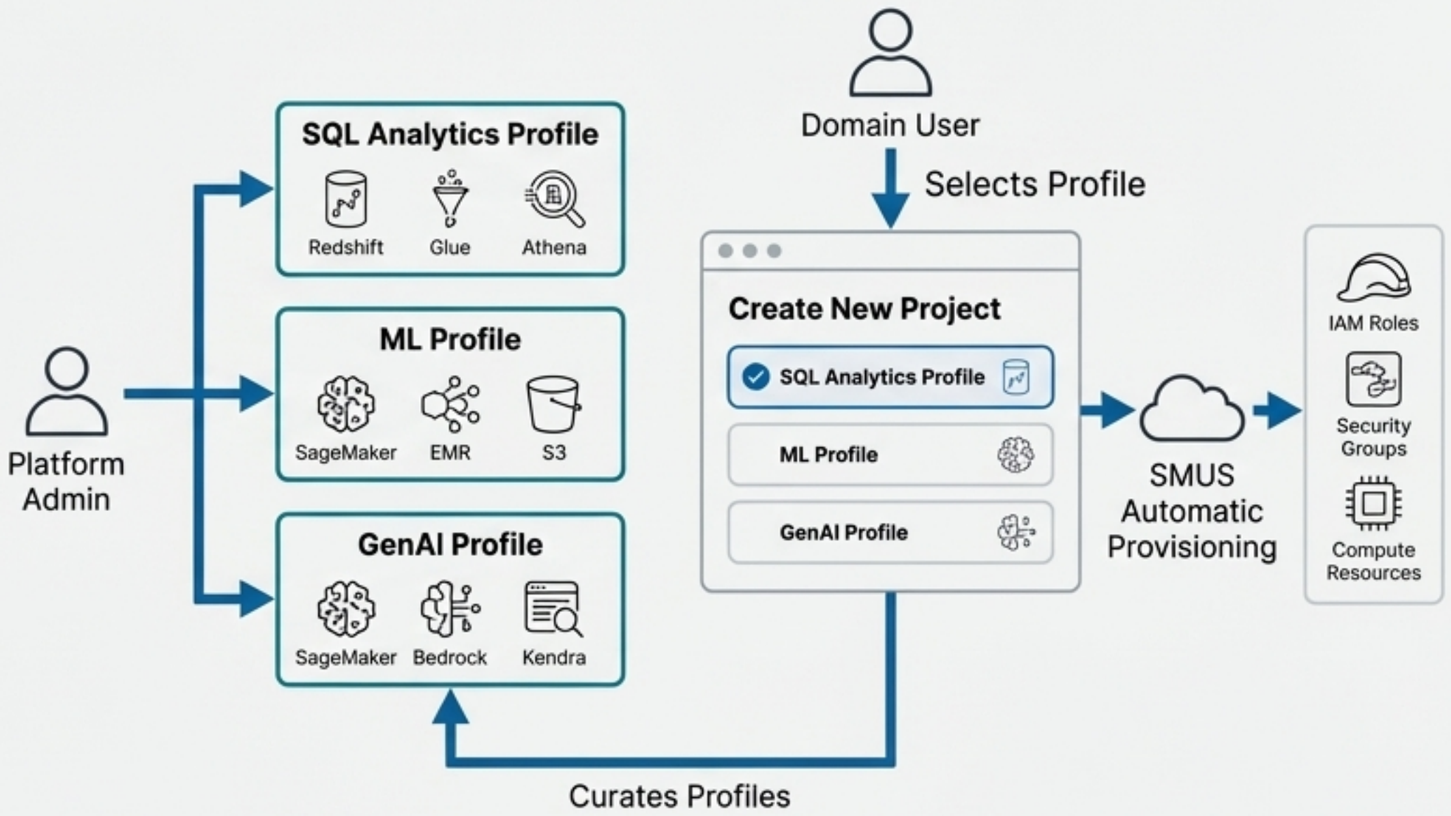
A core tenet of data mesh is enabling domains to be self-sufficient. SMUS achieves this through **Project Profiles** and **Blueprints**.

How it Works

Blueprints
A CloudFormation template that defines the AWS tools and services for a specific use case (e.g., an EMR on EC2 cluster, a Redshift Serverless warehouse).

Project Profiles
A collection of blueprints curated by the platform administrator (e.g., 'SQL Analytics Profile' or 'Generative AI Profile').

Self-Service
When a domain team creates a new project, they simply select the profile that matches their needs. SMUS automatically provisions all the necessary IAM roles, security groups, and compute resources.



Data Mesh Principle: Self-Serve Data Platform
The platform team provides the tools and infrastructure as a service, allowing domain teams to focus on creating data products, not managing infrastructure.

SMUS: Making the Data Mesh a Practical Reality

By providing native constructs for multi-account federation, data productization, and federated governance, Amazon SageMaker Unified Studio transforms the theoretical principles of a data mesh into an actionable architectural blueprint.

Unified Experience: A single interface for producers and consumers to build, discover, govern, and consume data.

Native Multi-Account Support: Securely connect data domains across your entire AWS organization with Associated Accounts.

Clear Producer/Consumer Model: The Catalog and Subscription workflow provides a formal, auditable process for data sharing.

Built-in Federated Governance: Empowers data owners to control access at the source while maintaining enterprise-wide standards.