Agent Data Quality on Azure

Solution Overview

This solution provides an intelligent, Azure-native, and scalable platform for Data Quality Governance powered by LLM-based agents. It leverages Azure Kubernetes Service (AKS), Azure Database for PostgreSQL, Azure Data Lake Storage, and Azure OpenAI Service to deliver rule validation, contextual reporting, and compliance automation.

Problem Statement

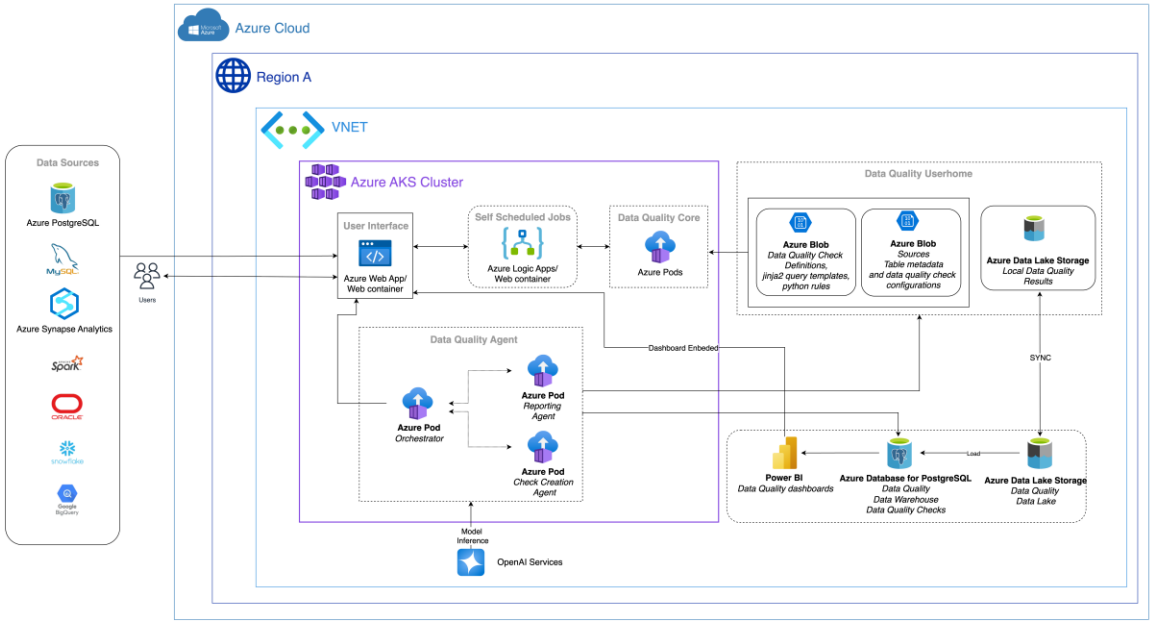
Enterprises struggle with maintaining data quality across hybrid and high-velocity data pipelines. Manual enforcement, delayed reporting, and lack of explainability introduce compliance risks and reduce trust in analytics.

Solution Detail

Our solution automates data quality validation, generates explainable insights via Azure OpenAI, and centralizes reporting with Power BI. Agents orchestrated on AKS ensure modularity, scalability, and resilience.

Technical Architecture

The architecture is based on AKS microservices with integration to Azure Database for PostgreSQL for metadata, Azure Data Lake Storage for results, and Power BI for dashboards. Event-driven orchestration is managed through Azure Logic Apps/Web containers within the AKS cluster.



Key Components

Core Services (within AKS Cluster)

1. **Azure Web App/Web Container** - User Interface for data quality management
2. **Azure Logic Apps/Web Container** - Self Scheduled Jobs for automated workflows
3. **Azure Pods** - Data Quality Core processing engine
4. **Azure Pod Orchestrator** - Coordinates and manages pod execution
5. **Azure Pod Reporting Agent** - Generates reports and insights
6. **Azure Pod Check Creation Agent** - Creates and manages data quality checks

Data Quality Userhome Components

1. **Azure Blob Storage (Data Quality Check)** - Stores data quality check definitions, templates, and Python rules
2. **Azure Blob Storage (Sources)** - Contains table metadata and data quality check configurations
3. **Azure Data Lake Storage** - Stores local data quality results and processed data

External Services Integration

1. **Azure Database for PostgreSQL** - Data Quality Data Warehouse for metadata and check results
2. **Power BI** - Data Quality dashboards and visualization
3. **OpenAI Services** - Model inference for intelligent data quality insights

Data Sources

- Azure PostgreSQL
- Azure Synapse Analytics
- Apache Spark
- Oracle
- Snowflake

Integration Points

Azure Database for PostgreSQL, Azure Data Lake Storage, Azure OpenAI, Azure Monitor, Power BI, Azure Logic Apps, Azure Blob Storage, Azure AKS.

Use Cases

1. Automated anomaly detection
2. Rule generation with LLM support
3. Compliance reporting
4. Contextual dashboards for stakeholders

Customer Pain Points Addressed

1. Manual, delayed enforcement
2. Lack of real-time alerts
3. No centralized rule management
4. Limited explainability and auditability

Industry-Specific Applications

- **Retail**: Catalog validation
- **Manufacturing**: Sensor data validation
- **Logistics**: SLA monitoring
- **Finance**: Transaction integrity

Sample Customer Journey

1. Deploy solution on AKS with all required pods and containers
2. Configure data sources (PostgreSQL, Synapse, Spark, Oracle, Snowflake)
3. Create data quality checks using the Check Creation Agent
4. Schedule automated jobs via Logic Apps/Web container
5. Execute validation tasks through Pod Orchestrator
6. Store check definitions in Azure Blob Storage
7. Store results in PostgreSQL and Data Lake Storage
8. Visualize insights in Power BI dashboards

9. Review AI-generated natural language summaries from OpenAI

Technical Requirements

1. Azure Kubernetes Service (AKS) with Kubernetes 1.27+
2. Azure Database for PostgreSQL
3. Azure Data Lake Storage Gen2
4. Azure Blob Storage (multiple containers)
5. Azure OpenAI access
6. Azure Logic Apps
7. Ingress controller with TLS

Security Architecture

Azure AD-based authentication, RBAC, Managed Identities, encryption at rest (Azure-managed keys), and in transit (TLS). All communications within the AKS cluster are secured through Azure networking and pod security policies.

Performance Considerations

- Asynchronous execution for large jobs through Pod Orchestrator
- Autoscaling with AKS HPA for all pod components
- Geo-replicated PostgreSQL Database
- Event-driven flows with Azure Logic Apps
- Distributed storage across Blob Storage and Data Lake Storage

Tools and Azure Services Used

- **Container Orchestration**: Azure Kubernetes Service (AKS)
- **Database**: Azure Database for PostgreSQL
- **Storage**: Azure Data Lake Storage Gen2, Azure Blob Storage
- **AI/ML**: Azure OpenAI Service
- **Visualization**: Power BI
- **Monitoring**: Azure Monitor
- **Workflow**: Azure Logic Apps
- **Compute**: Azure Web Apps/Containers, Azure Pods
- **Development**: Python, Jinja2

Users of Agent

- Data Engineers
- Compliance Officers
- Data Stewards

- Governance Leads
- Business Analysts

Dependencies

- Kubernetes (AKS) with pod orchestration capabilities
- Terraform/Helm/GitOps pipelines
- Azure Database for PostgreSQL
- Azure OpenAI subscription
- Azure Logic Apps for workflow automation
- Multiple Azure storage services (Blob Storage, Data Lake Storage)