Data Platform Design

Rong Wang

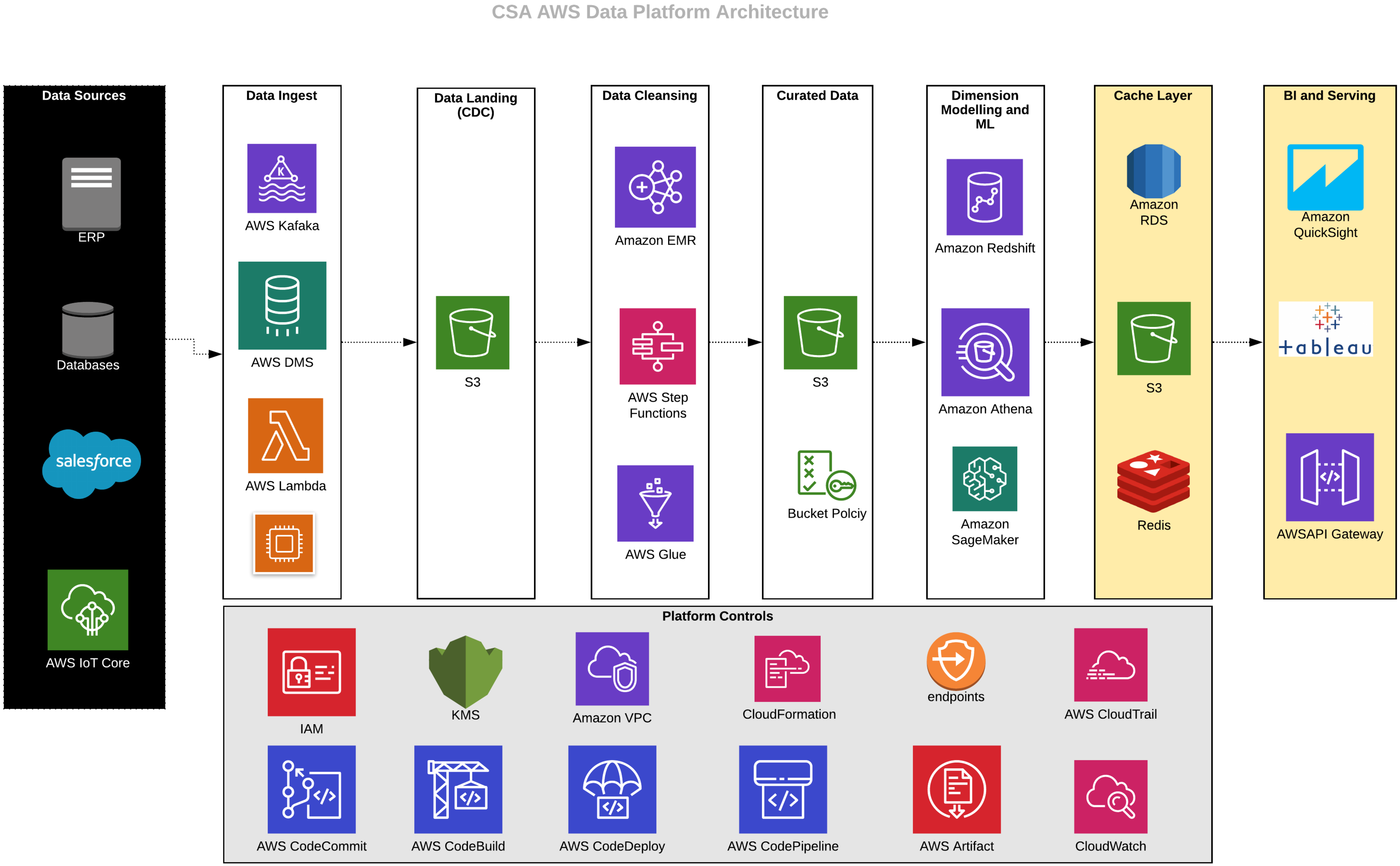
The ASK

- Complementing to existing data warehouse
- Catering for data ingestion protocol of
  - RESTful API (JSON, HTTP)
  - JDBC (Database connection, TCP)
  - SOAP (XML, HTTP)
  - IoT Stream (MQTT, TCP)
- Allow data processing
- Allow data analytics
- Allow data accessible via API

Solution Summary

- AWS Based Data Platform
  - Ingestion (Lambda, DMS, Kafka)
  - Transformation (Glue, EMR)
  - Loading (S3, Redshift, Athena)
  - Access (RDS, S3, Lambda)
- Airflow based data workflow

Platform Architecture

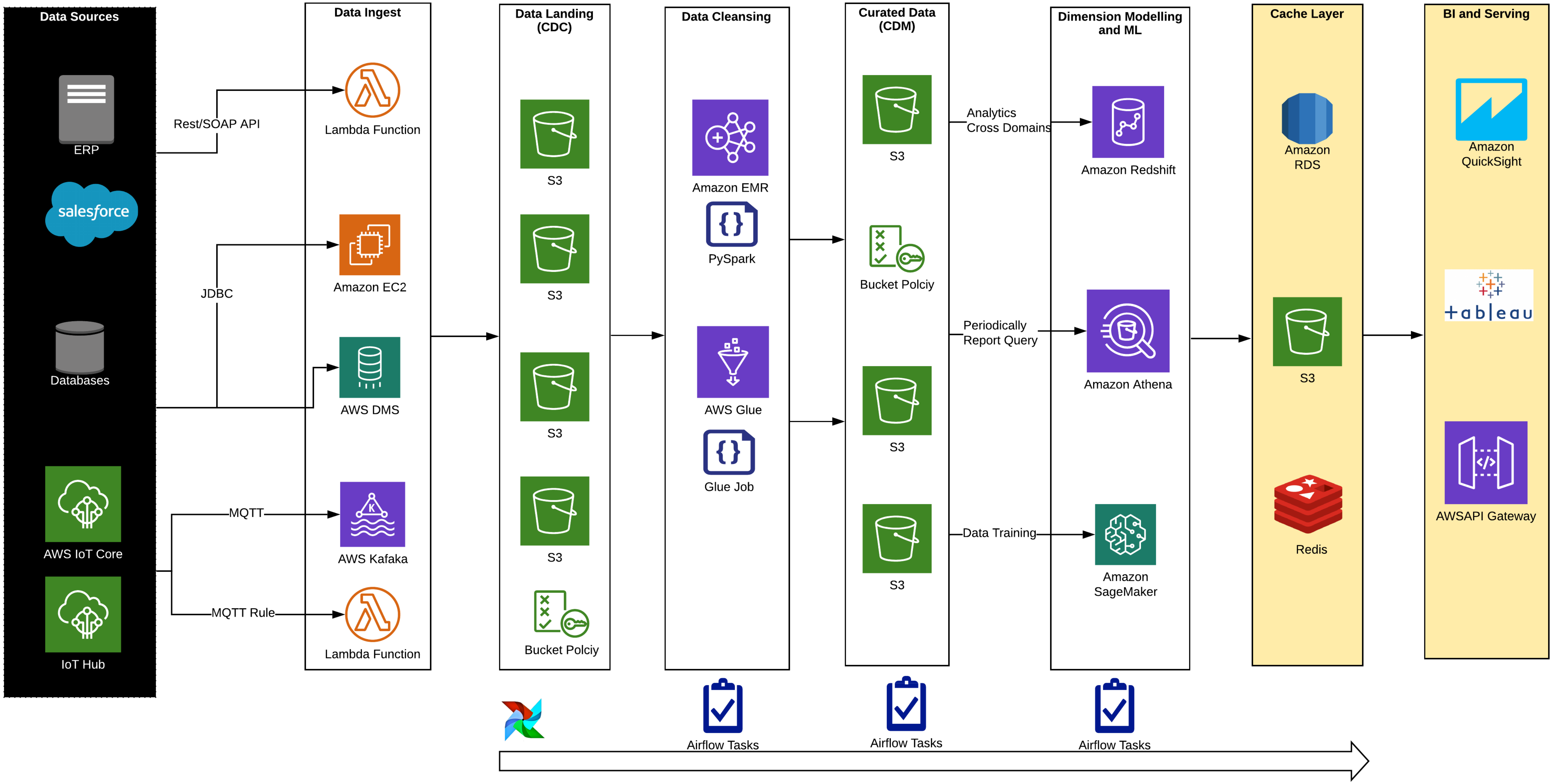


Key Considerations

- AWS LandingZone is recommend to ensure data privilege and environment segregation
- AWS Compute includes (Lambda, ECS, EC2)
- DevOps mindset and practices to ensure iteration of component release
- Heavy security controls policy and automated scanning integration (DevSecOps)
- The CDM data are tightly controlled via bucket policy

Data Workflow

CSA AWS Data Platform



Key Considerations

- IoT analytics is available for early analytics
- Lambda is heavily used across multiple stage of data processing, establish Serverless framework is important
- Chose Airflow due to its flexibility and portability (GCP has composer long ago compare to AWS)
- Promoting containerisation culture e.g. Airflow tasks, EMR processing nodes
- The data access need to be tightly controlled using IAM role with IdP integration via SSO if possible

Future Enhancement

- Apply S3 bucket lifecycle policy to reduce storage cost
- EMR Cluster and Redshift is costly, further study to optimise or using 3rd management like Snowflake
- GCP is strong in some aspect of data processing (e.g. Composer and DataProc is more matured, BigQuery is much powerful)
- Using AWS to describe and research the connectivity challenge. Then deploy the actual workload using GCP tooling for performance gain. (Personal opinion)

Q&A