Data Analytics Platform

Data Warehouse, Data Lake, and the Modern Data Cloud

Rowi Fajar Muhammad, SI 2012

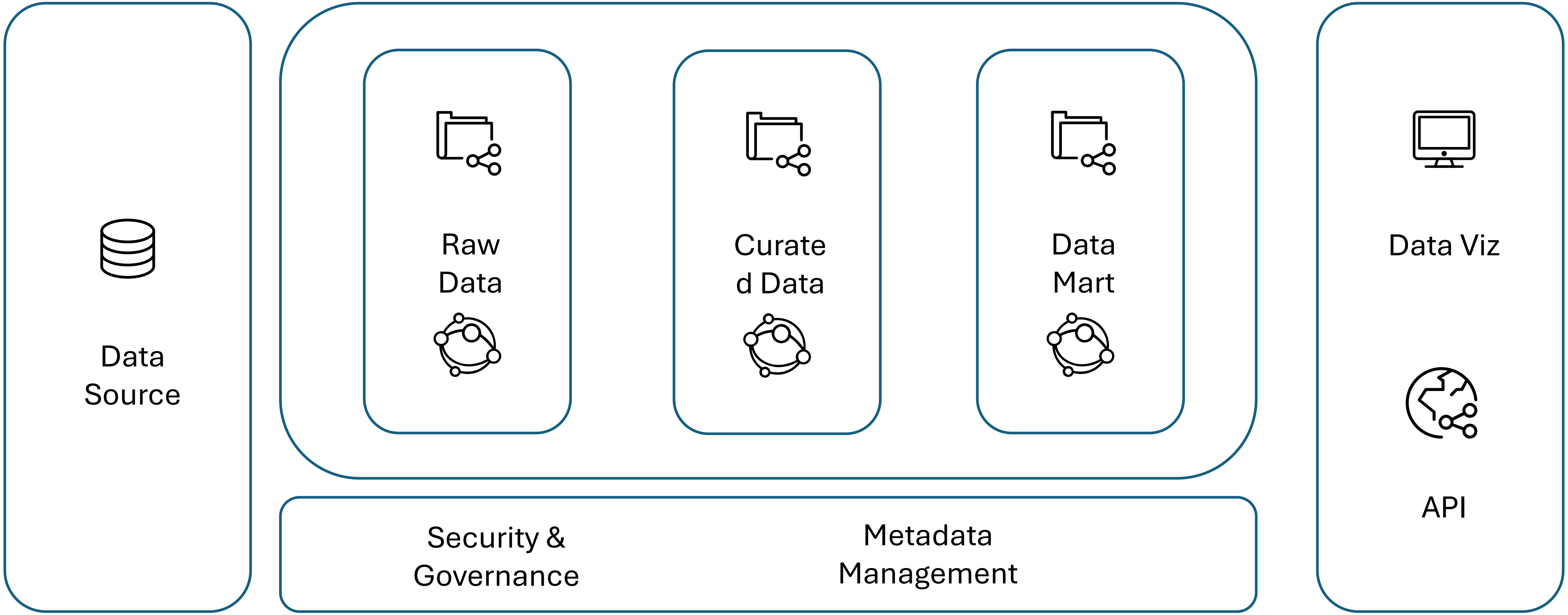
Outline

- Introduction
- The history of Data Analytics Platform
- Use Cases
- Snowflake Data Platform, Deep Dive

Traditional Data Warehouse

- Structured and data
- Optimized for complex query
- High performance, incl. indexing and partitioning
- Strong Data Governance
- Integration with BI

Typical Big Data / Data Lake Architecture



Data Source

Raw Data

Curated Data

Data Mart

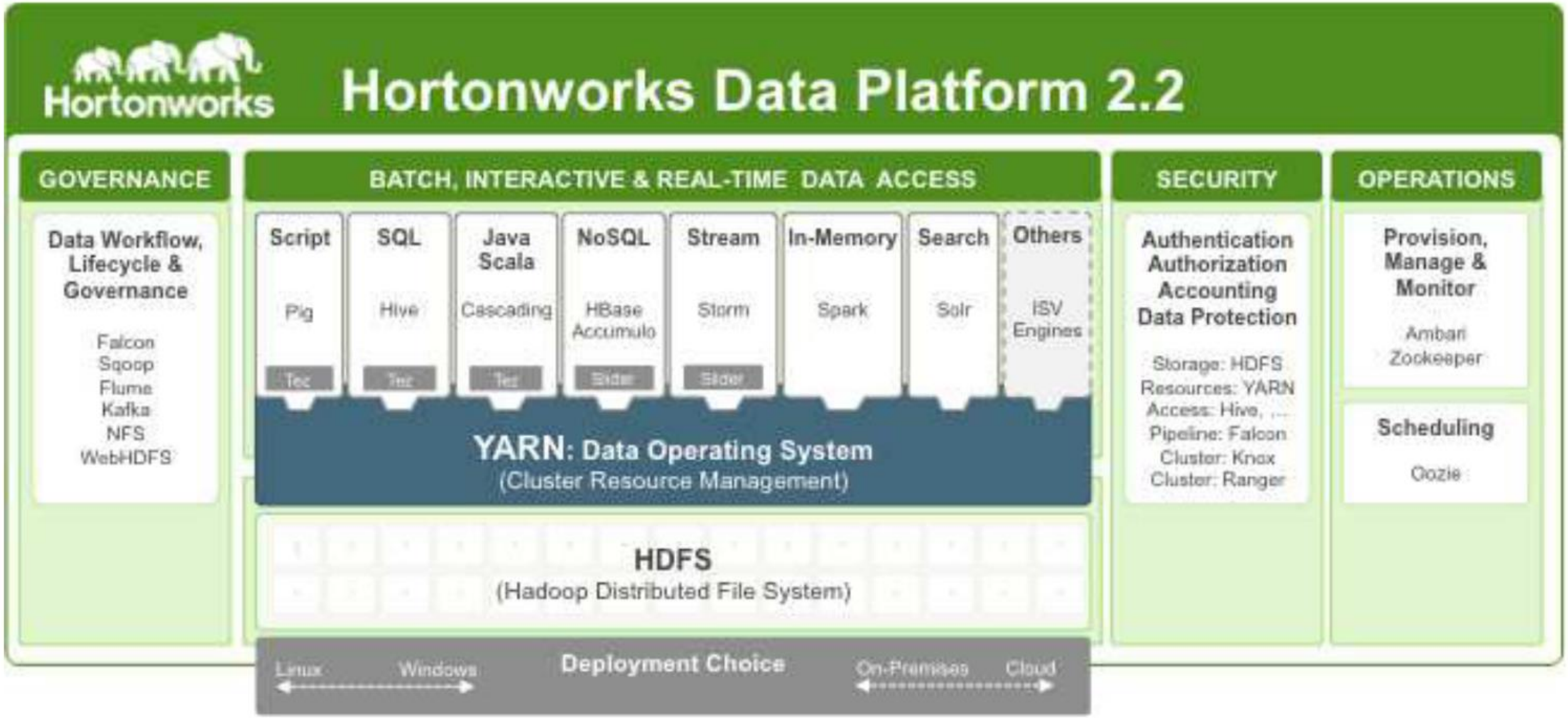
Security & Governance

Metadata Management

Data Viz

API

The Big Data Architecture



Hortonworks

Hortonworks Data Platform 2.2

GOVERNANCE

Data Workflow, Lifecycle & Governance

Falcon
Sqoop
Flume
Kafka
NFS
WebHDFS

BATCH, INTERACTIVE & REAL-TIME DATA ACCESS

Script: Pig (Tez)

SQL: Hive (Tez)

Java Scala: Cascading (Tez)

NoSQL: HBase Accumulo (Slider)

Stream: Storm (Slider)

In-Memory: Spark

Search: Solr

Others: ISV Engines

YARN: Data Operating System
(Cluster Resource Management)

HDFS
(Hadoop Distributed File System)

SECURITY

Authentication Authorization Accounting Data Protection

Storage: HDFS
Resources: YARN
Access: Hive, ...
Pipeline: Falcon
Cluster: Knox
Cluster: Ranger

OPERATIONS

Provision, Manage & Monitor

Ambari
Zookeeper

Scheduling

Oozie

Deployment Choice

Linux ↔ Windows

On-Premises ↔ Cloud

The Big Data Architecture (Pros)

- Flexibility and Scalability to support large and diverse datasets
- Support of Raw and Unprocessed Data, process the data in its original form
- Separation of compute, storage. Independent scaling between them
- Advanced Data Analytics, incl. AI&ML
- Data Governance

Hadoop Component Platform

- Storage
  - HDFS
  - AWS S3
- Processing Engine
  - YARN
  - MapReduce
- Metadata Layer
  - Hive Metastore

- Query Engine
  - Hive
  - Impala
- Framework & Processing
  - Apache Spark
  - Flink
- Streaming Platform
  - Kafka

The Big Data Architecture (Pros)

- Hard to manage, need to maintain each component separately
- For On-Prem env, need a huge investment upfront while the workload is dynamic.
- Latency and performance issues,
  - Big Data designed for large-scale batch processing
  - For real-time, it has a big latency
  - Resource pooling and management affecting this issues to

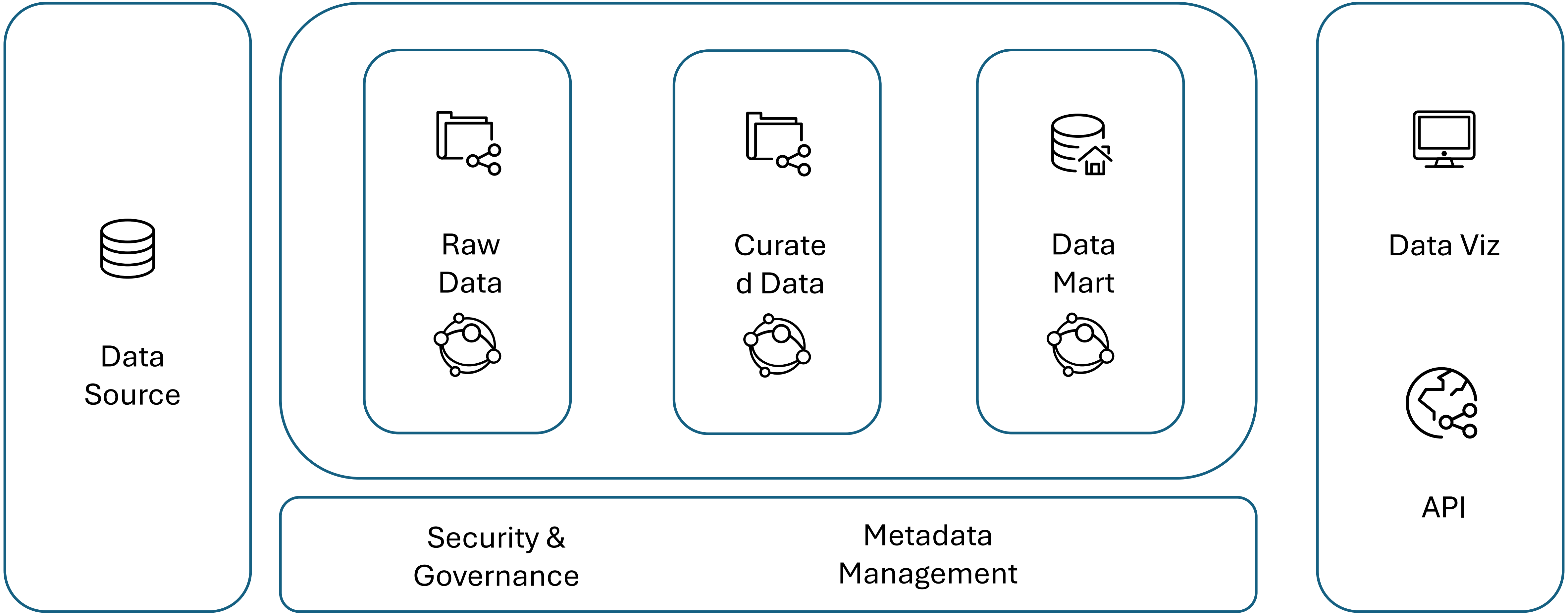
File Format

- JSON
- CSV
- Row Based Format
  - Avro
- Columnar Based
  - ORC
  - Parquet
  - Spark

Data Lakehouse

- Basically a combination of Data Lake + Data Warehouse
- Put the raw and staging data into Data Lake, then the Data Mart put into Data Warehouse

Typical Data Lake House Architecture



Data Source

Raw Data

Curated Data

Data Mart

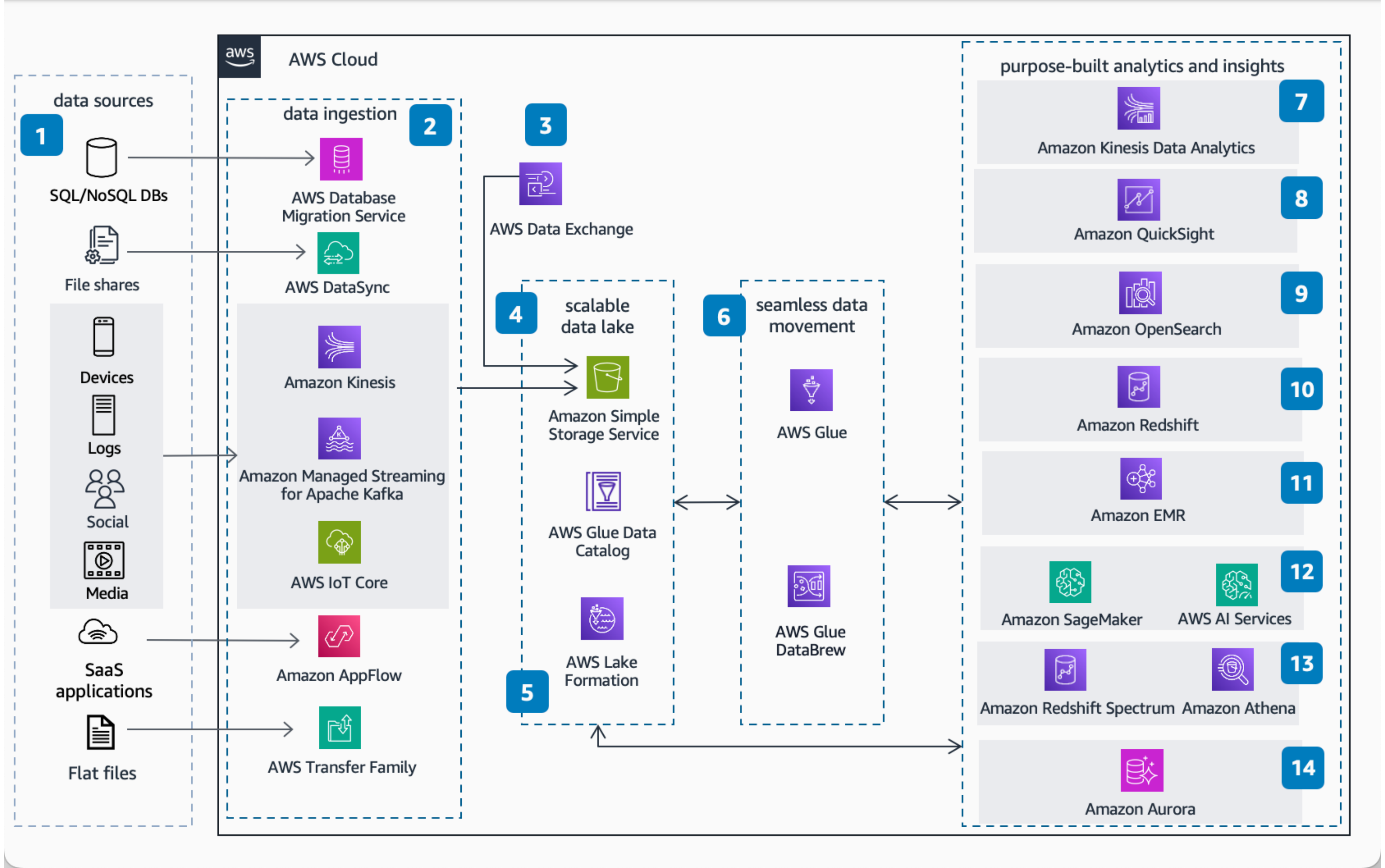
Security & Governance

Metadata Management

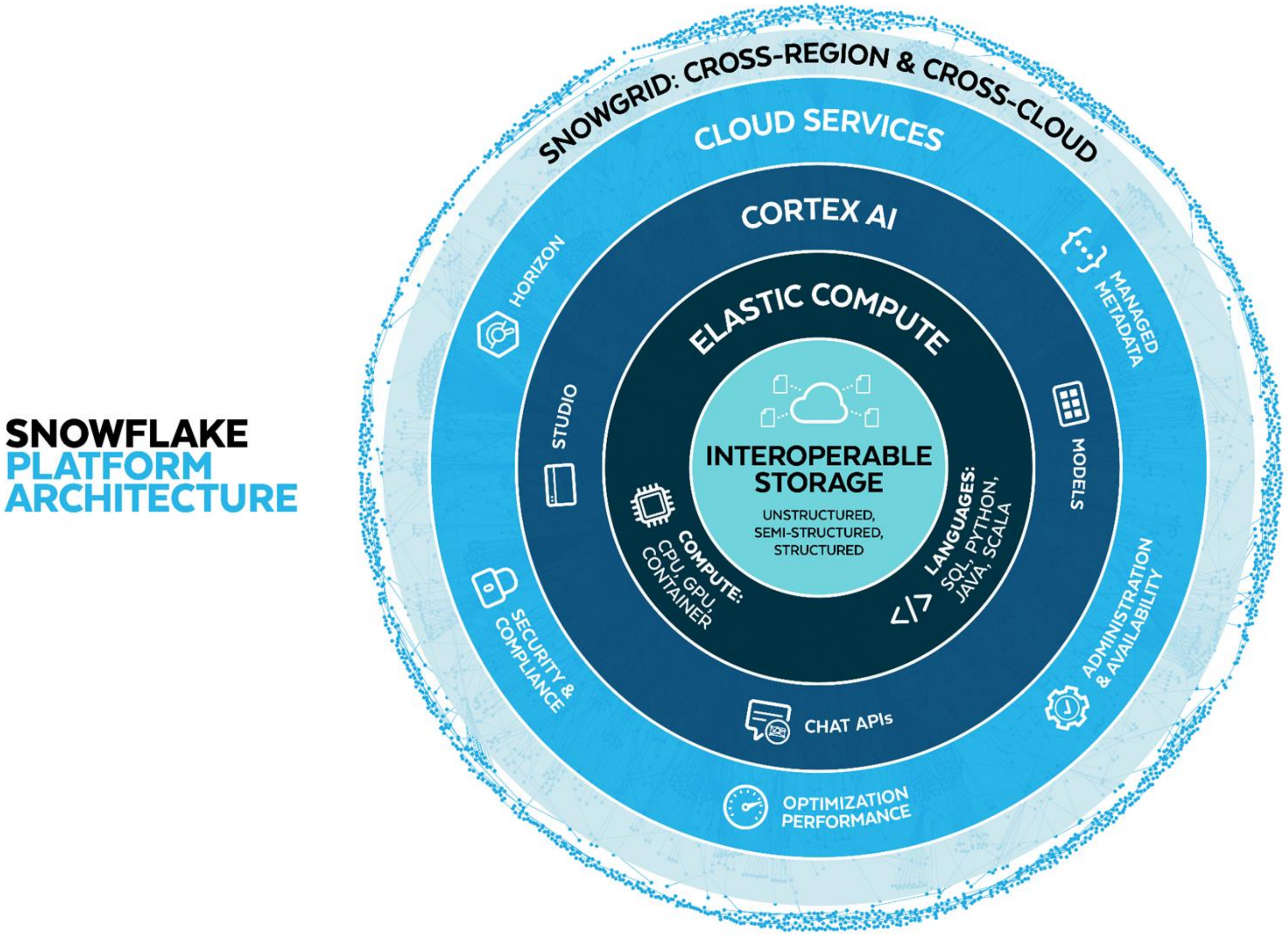
Data Viz

API

AWS Data Analytics Architecture



Snowflake Architecture



Use Cases

Rowi Fajar Muhammad, SI 2012

Precision Forestry

Image Capture by Using Drone

A drone flying on top of forestry, capturing ~20 HA. Image size ~5GB

Image & Spatial Data Analysis

Computer Vision + Geospatial is used to do image analysis, providing the insight

Data Visualization

Data is being visualized, both image, analysis result and geospatial data

Apps Integration

The Insight is pushed into several apps, incl. Mobile Apps for worker and ERP to make Work Order

Result : More accurate analysis (5% sample vs 80% accuracy), faster (a month vs 2 days)

Automotive Industry

Data Ingestion

Data Ingestion from data source to Big Data Platform

ETL & Machine Learning

Create a data pipeline. Leverage machine learning to generate use case, and create a data mart

Data Visualization

Data is being visualized, based on the analysis

Apps Integration

The insight pushed into company internal ERP

Result : Targeted marketing, more accurate segmentation

Data Warehouse Migration

Data Ingestion

Data Ingestion from data source to Big Data Platform. Could be batch or real-time

Data Quality Checking

Check the data quality ingestion, make sure it meets the business standard

ETL Refactoring

Refactor the ETL/ data pipeline based on existing pipeline

Apps Integration

Re-pointing any apps that has been connected previously by using DB Connector (JDBC,ODBC, etc.)

Result : Faster Insight Generation, More Efficient Wrokload