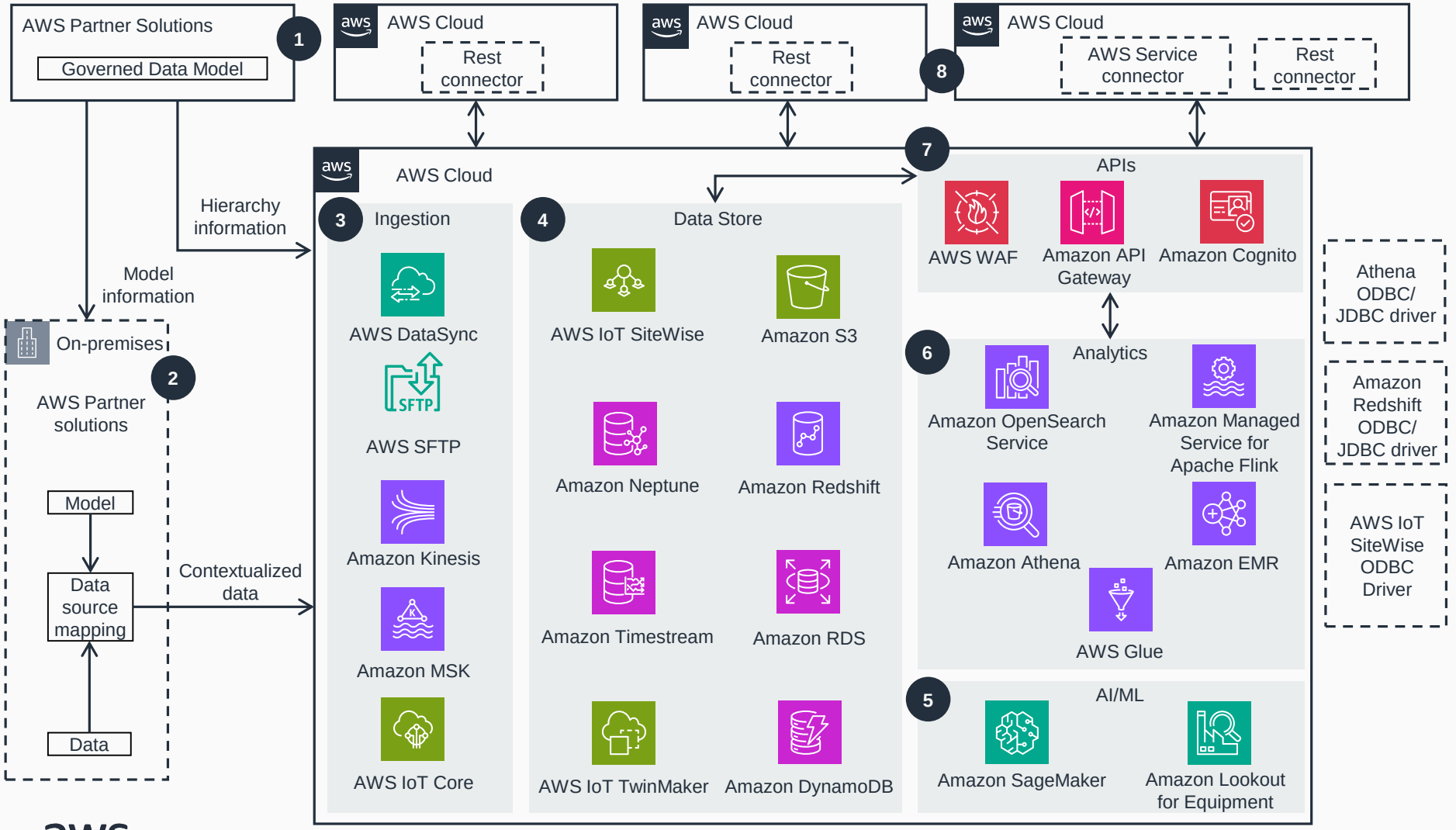
Guidance for Industrial Data Fabric with HighByte Intelligence Hub on AWS

Delivering Industrial DataOps (IDO) on Industrial Data Fabric (IDF)

This high-level architecture diagram is a reference that helps you create an enterprise governed model, ingest near real-time and historical data at scale from edge data sources into IDF on AWS, and interface with applications using REST APIs.



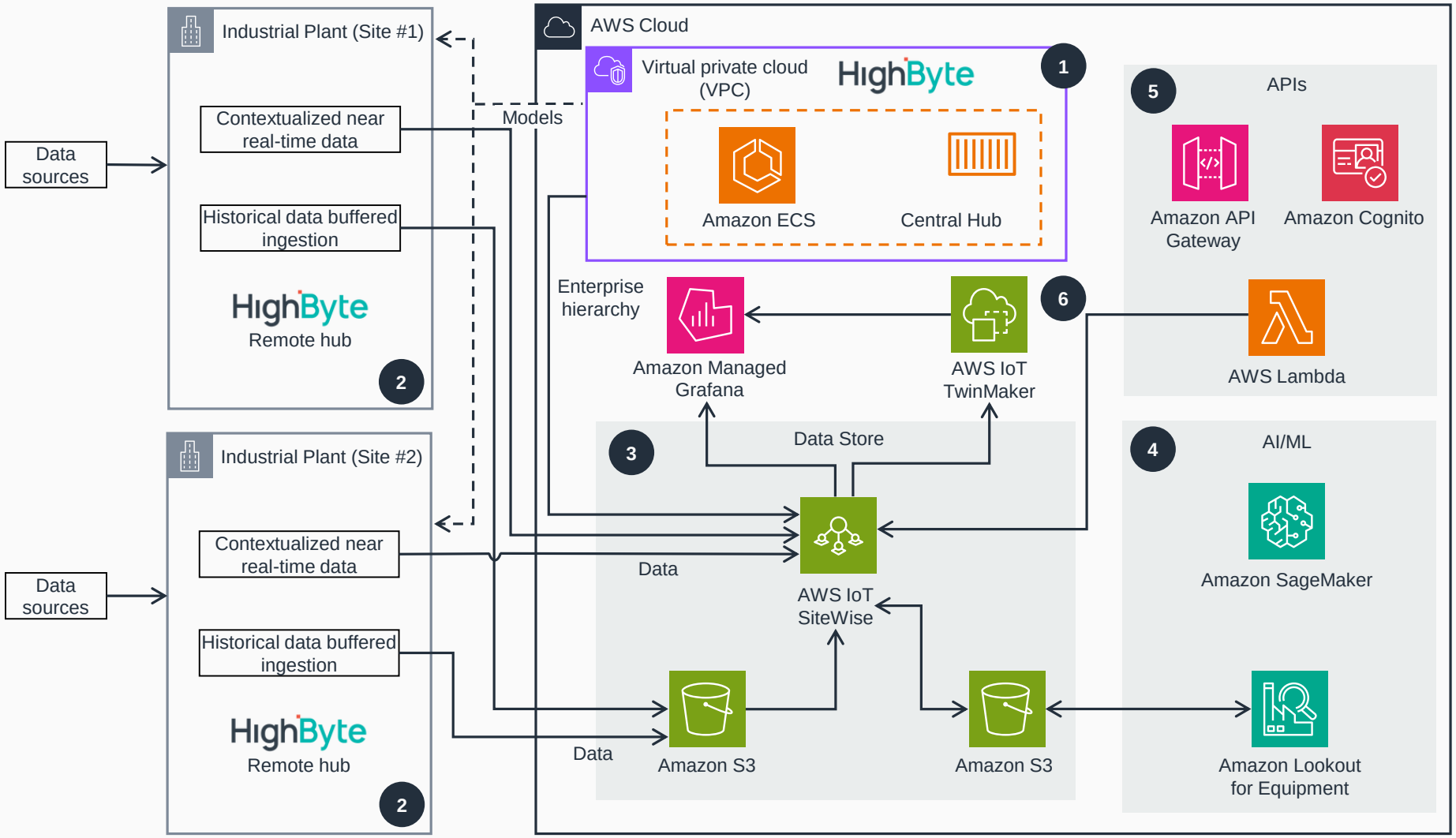
1. Define a governed data model in a central location. The centralized model can be populated with metadata from corporate systems such as enterprise resource planning (ERP), piping and instrumentation diagram (P&ID), and computerized maintenance management system (CMMS).
2. Bring the governed model into an edge application, where additional metadata from systems, such as a manufacturing execution system (MES), can be added. Data streams can then be mapped to the imported model. Asset Hierarchy can also be moved to **AWS IoT SiteWise**.
3. Ingest data depending on the source. Stream data feeds from a variety of sources using **AWS DataSync** for file share, **Amazon Kinesis, Amazon Managed Streaming for Apache Kafka (Amazon MSK)**, **AWS IoT Core**, **AWS IoT SiteWise**, or **AWS Transfer Family for SFTP**.
4. Optimize data storage for the workload, which can include **Amazon DynamoDB** key value and document data structures, **Amazon Simple Storage Service (Amazon S3)** for object storage, **Amazon Neptune** for graph use cases, **Amazon Redshift** for data warehousing, **Amazon Timestream** for time series data, and **AWS IoT SiteWise** to organize industrial equipment data. Data from **AWS IoT SiteWise** can be integrated with **AWS IoT TwinMaker**.
5. Use AWS artificial intelligence and machine learning (AI/ML) services such as **Amazon SageMaker** to build, train, and deploy ML models.
6. Use AWS analytics services such as **Amazon OpenSearch Service**, **Amazon Managed Service for Apache Flink**, **Amazon Athena**, **Amazon EMR**, and **AWS Glue** for data processing.
7. **AWS WAF** provides protection from web exploits while **Amazon API Gateway** provides REST method support. **Amazon Cognito** provides user authentication and AWS token support.
8. Egress data from the Industrial Data Fabric (IDF) with connectors directly to AWS services or through **API Gateway** to supported AWS services. For services that are not supported by **API Gateway**, an OAuth 2.0 pattern with **Amazon Cognito** is used to generate AWS temporary tokens.

Reviewed for technical accuracy April 10, 2024

AWS Reference Architecture

IDF Governed Data Model with HighByte Intelligence Hub

This architecture diagram helps you create an enterprise governed data model, ingest real-time and historical data at scale from edge data sources, and visualize this data using Amazon Managed Grafana. AWS IoT SiteWise provides a contextualized timeseries data store and AWS IoT TwinMaker allows for digital twin scenes.



1. Define a governed data model in HighByte Central Hub running as a Docker image in **Amazon Elastic Container Service (Amazon ECS)**. Publish the Enterprise Hierarchy from HighByte to **AWS IoT SiteWise**, including model information. Sync asset models from the central hub to the remote hubs.
2. Connect to data sources with the HighByte remote hubs using models from the central hub. Insert data into **AWS IoT SiteWise** in the appropriate portion of the hierarchy model. Data can be inserted in near real-time using **AWS IoT SiteWise** or delayed using buffered ingestion.
3. Use buffered ingestion through **Amazon S3** to reduce **AWS IoT SiteWise** ingestion costs. The **AWS IoT SiteWise** Open Database Connectivity (ODBC) driver provides direct integration to **AWS IoT SiteWise** for business intelligence (BI) client tools.
4. Train models in **Amazon Lookout for Equipment** for anomaly detection and prediction maintenance using data from **AWS IoT SiteWise**. Native **AWS IoT SiteWise** integration through **Amazon S3** allows data exchange between services. Use AWS AI/ML services such as **SageMaker** to build, train, and deploy custom ML models.
5. Access **AWS IoT SiteWise** data through **API Gateway** and **AWS Lambda** with **Amazon Cognito** for authenticated REST method support.
6. Visualize **AWS IoT SiteWise** data and **AWS IoT TwinMaker** scenes using **Amazon Managed Grafana**.

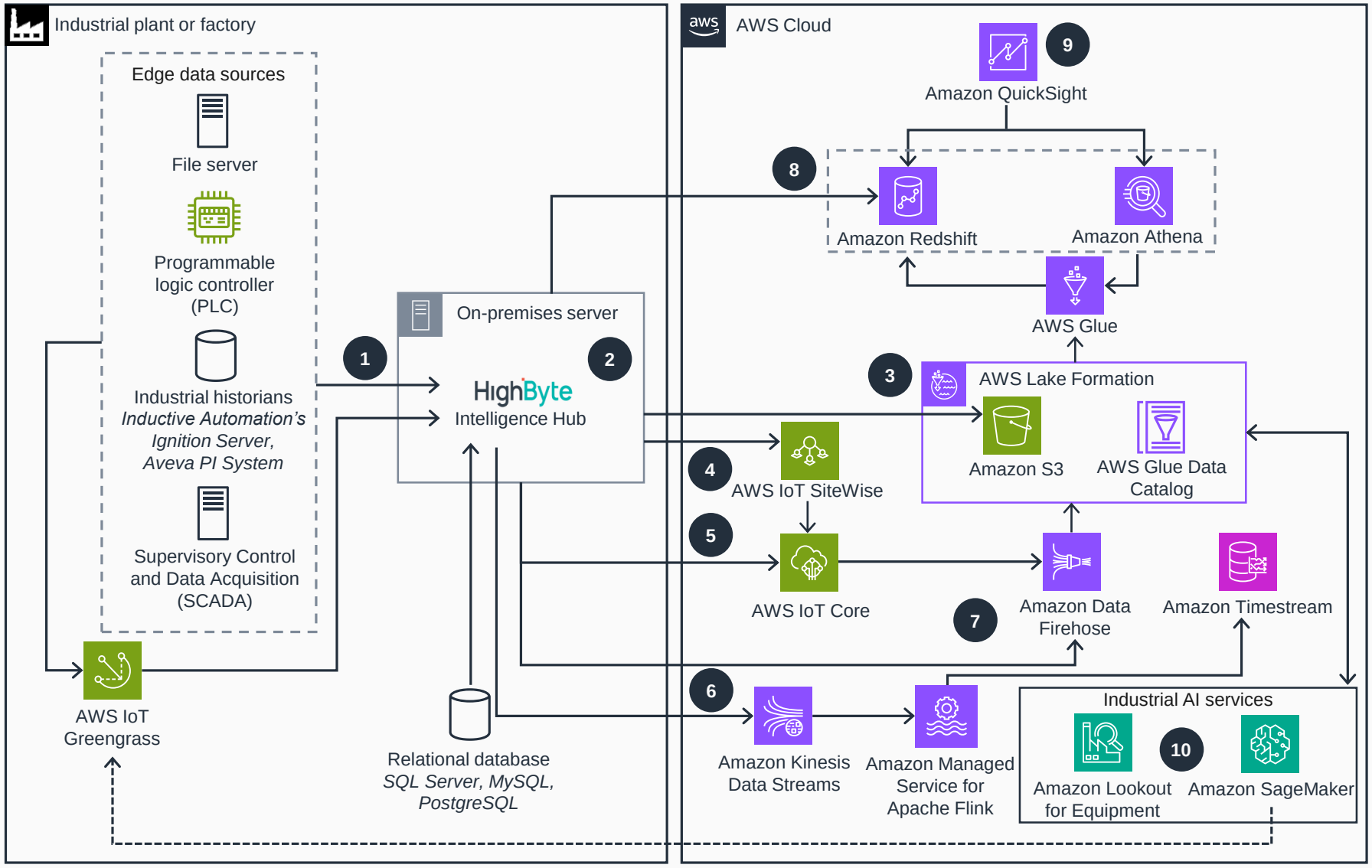
Reviewed for technical accuracy April 10, 2024

AWS Reference Architecture

Integrate Industrial Data with Enterprise Systems on AWS

HighByte Intelligence Hub Industrial DataOps on AWS

This architecture diagram demonstrates how HighByte Intelligence Hub integrates OT with IT to combine industrial information across multiple systems, enabling OT teams to model, transform, and share plant floor data with IT systems. This slide details Steps 1-6.



1. HighByte Intelligence Hub consumes both real-time and asset model data from a myriad of edge data sources, including relational databases and **AWS IoT Greengrass**, by using standard industrial protocol input connectors. This includes data ingestion from industrial historians, such as Inductive Automation's Ignition Server and Aveva's PI System.
2. Intelligence Hub enables customers to standardize, organize, and merge your industrial data into a single equipment model. Then, using *flows*, you can route the asset models to multiple output connectors, each with a different frequency. HighByte can be deployed on the edge using an AWS container option, including **IoT Greengrass, Amazon Elastic Kubernetes Service (Amazon EKS) Anywhere**, and **Amazon ECS Anywhere**. Intelligence Hub provides a variety of output connectors that support many of standard industrial protocols, including a native **AWS IoT SiteWise** connector.
3. Bring in asset models and timeseries based sensor data into **Amazon S3**. **AWS Lake Formation** helps you collect and catalog data from databases and object storage, move the data into **Amazon S3**, and clean and classify data using ML algorithms. Data is accessed through **AWS Glue Data Catalog**.
4. Build asset models within the HighByte editor and deploy the model directly to **AWS IoT SiteWise** along with the streaming data. Calculate and visualize metrics from telemetry data using **AWS IoT SiteWise Monitor**.
5. Intelligence Hub can connect directly to **AWS IoT Core** through its native Message Queuing Telemetry Transport (MQTT) service or use **IoT Greengrass** locally. Intelligence Hub also enables bi-directional communication with **AWS IoT Core** and **IoT Greengrass**.
6. Intelligence Hub can connect directly to **Amazon Kinesis Data Streams** for massively scalable and durable near real-time data streaming. Streaming data can be transformed and analyzed in near real-time using **Amazon Managed Service for Apache Flink**, and sent to **Amazon Data Firehose**. Time series data can also be sent to **Timestream** from **Amazon Managed Service for Apache Flink**.

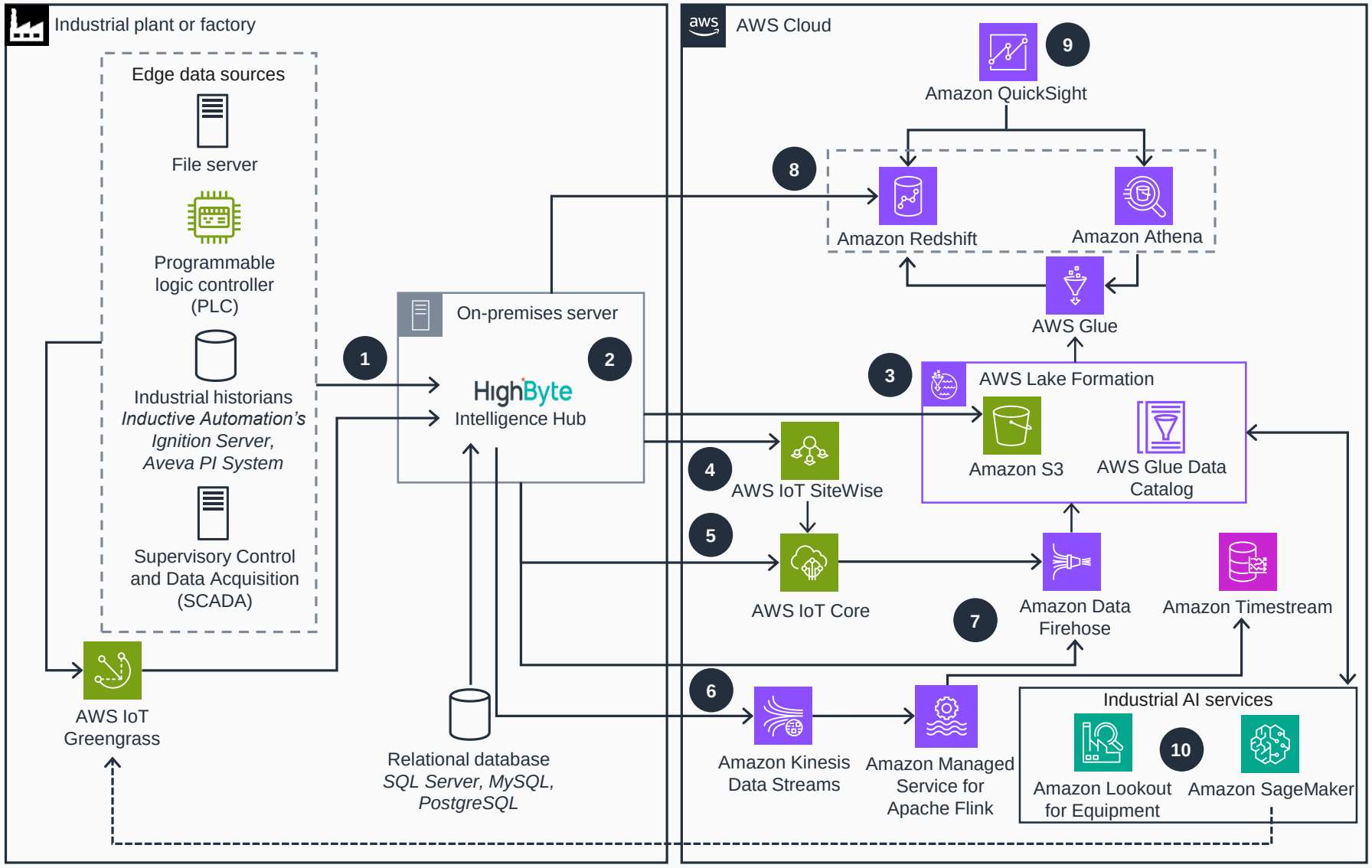
Reviewed for technical accuracy April 10, 2024

AWS Reference Architecture

Integrate Industrial Data with Enterprise Systems on AWS

HighByte Intelligence Hub Industrial DataOps on AWS

This architecture diagram demonstrates how HighByte Intelligence Hub integrates OT with IT to combine industrial information across multiple systems, enabling OT teams to model, transform, and share plant floor data with IT systems. This slide details Steps 7-10.



7. Telemetry data is published in near real-time to **Data Firehose** by either an **AWS IoT Core** rule, **Kinesis Data Streams**, or a HighByte and **Data Firehose** connector. This loads the streaming data reliably into an **Amazon S3** data lake.
8. Use **Amazon Redshift** to store structured data sets and analytics results in a data warehouse. Data into **Amazon Redshift** can be ingested either through **AWS Glue** from **Amazon S3** or directly through the Intelligence Hub Redshift connector.
9. Create BI reports and visualize data from **Amazon Redshift** and **Amazon S3** with **Amazon QuickSight** and **Athena**.
10. When real-time and historical data is available in **Amazon S3**, **Lookout for Equipment** uses the data to detect abnormal equipment behavior so that potential machine failures are detected before failures occur and unplanned downtime is avoided. Computed metrics can be written back into **Amazon S3** for storage and consumption. Custom ML models can be developed with **SageMaker**.

Reviewed for technical accuracy April 10, 2024

AWS Reference Architecture