Time series processing architecture in AWS

When data arrives as a succession of regular measurements, it is known as time series information. Processing of time series information poses systems scaling challenges that the elasticity of AWS services is uniquely positioned to address.

This elasticity is achieved by using Auto Scaling groups for ingest processing, AWS Data Pipeline for intersystem data orchestration, and Amazon Redshift for potentially massive-scale analysis. Key architectural throttle points involving Amazon SQS for sensor message buffering and less frequent AWS Data Pipeline scheduling keep the overall solution costs perdictable and controlled.

Gliffy Macro Error

You do not have permission to view this diagram.

From above diagram, regarding a Supervisory Control and Data Acquisition (SCADA), we created a flow of samples to or from Amazon DynamoDB to support additional cloud processing or other existing systems, respectively.

Using AWS Data Pipeline, create a pipeline with a regular Amazon Elastic MapReduce job that both calculates expensive sample processing and delivers samples and results.

The pipeline places results into Amazon Redshift for additional analysis, and exports historical week-oriented sample tables, from Amazon DynamoDB to Amazon Simple Storage Service (Amazon S3). The pipeline also optionally exports results in a format custom applications can accept.

Amazon Redshift optionally imports historic samples to reside with calculated results.