Moving Operational Data into AWS at Scale in less than 60 minutes

AWS has powerful IoT capabilities, including the recently released time series database Timestream. With DeepIQ's DataStudio, moving your IoT data into AWS from your diverse industrial technology and network landscape is a simple task with no code development requirement. This whitepaper will explain how you can build a production-grade data pipeline with real-time streaming data from your industrial systems into your AWS Timestream database in less than 60 minutes. Using a real-world use case, we will explain how to build these data pipelines and conclude with a brief discussion about the data apps you can build with this technology.

:kground

DeepIQ Introduction

DeepIQ is a self-service {Data + AI} Ops app built for the industrial world. DeepIQ simplifies industrial analytics by automating the following three tasks:

- Ingesting operational and geospatial data at scale into your cloud platform.
- Implementing sophisticated time series and geospatial data engineering workflows; and
- Building state of the art ML models using these datasets.

AWS IoT Offerings

AWS has rich offerings in the IoT landscape across compute, storage, and AI/ML services. This whitepaper will focus on moving industrial data from a control system into an AWS Timestream database using AWS Kinesis as the intermediate stream buffer. While we use KepwareServerEx's OPC UA server as our streaming data source in this whitepaper, DeepIQ has similar support for many other industrial time series data sources including historians like OSI PI and Honeywell PhD and industry specific protocols such as WITSML. For further information, please get in touch with info@deepiq.com.

Amazon Timestream is a fast, scalable, and serverless time series database service for IoT and operational applications that makes it easy to store and analyse trillions of events per day up to 1,000 times faster and at as little as 1/10th the cost of relational databases.

Amazon Kinesis is a serverless streaming data service that makes it easy to capture, process, and store data streams at any scale.

Use case

In this sample use case, the data is stored in an on-premises time series datastore using KEPServerEX with an OPC UA server attached to it. We are required to move this data at scale into an AWS Timestream database. The raw data is recorded time series values that can come at irregular intervals. The final goal is to create a cleaner data source for analytics, a regular spaced table with 1-hertz data.

Architecture

For this use case, we will use the architecture shown in Figure 1. We have a DeepIQ edge connector that uses an OPC UA client to connect to the on-premises streaming data source. DeepIQ edge uses an encrypted and compressed data channel to push messages to AWS cloud and land data into the storage layer of your choice. In this use case, we will persist incoming high-volume time series data into a Kinesis stream and use DeepIQ's DataStudio software to format the data into AWS Timestream data model. We then use another DataStudio workflow to create a cleaned time series table with 1-HTZ frequency data. DeepIQ's DataStudio workflow leverages Apache Spark's distributed and parallel computing framework to scale to high volumes.

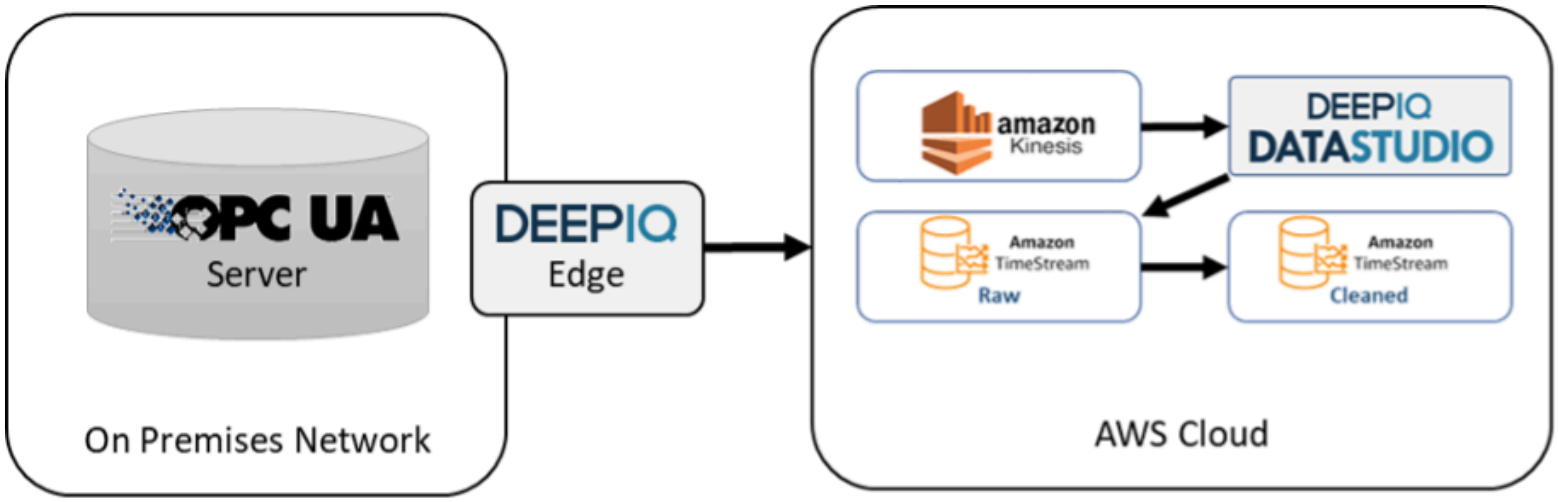

Figure 1: Architecture

DeepIQ's edge connector allows users to parallelize data read requests and maximize throughput to the extent supported by the source system. In this whitepaper, we will focus on the throughput of the cloud with different strategies.

Approach and Results

Deploying DataStudio in AWS is a simple one-step process. We use DeepIQ DataStudio preinstalled in the AWS environment for this sample use case. We also installed a DeepIQ Edge connector on our on-premises network where the timeseries data source is available. DeepIQ edge installation is also GUI based and hassle-free.

Once deployed, we are ready to build the production pipeline from the historian to AWS cloud. This task is a three-step process.

- Step 1: We configure DeepIQ Edge to publish data to an AWS Kinesis Data Stream/Topic.
- Step 2: We use a DeepIQ DataStudio workflow to read data from Kinesis and write it to an Amazon TimestreamDB table.
- Step 3: We use another DeepIQ DataStudio workflow using proprietary components to read and process data. These include time series data transformation and cleansing techniques such as interpolation.

Step 1:

First, we create a request for data using DataStudio's edge control panel. We specify the dates and tags of interest and the maximum number of retrievals for each server query in our data request, as shown in Figure 2.

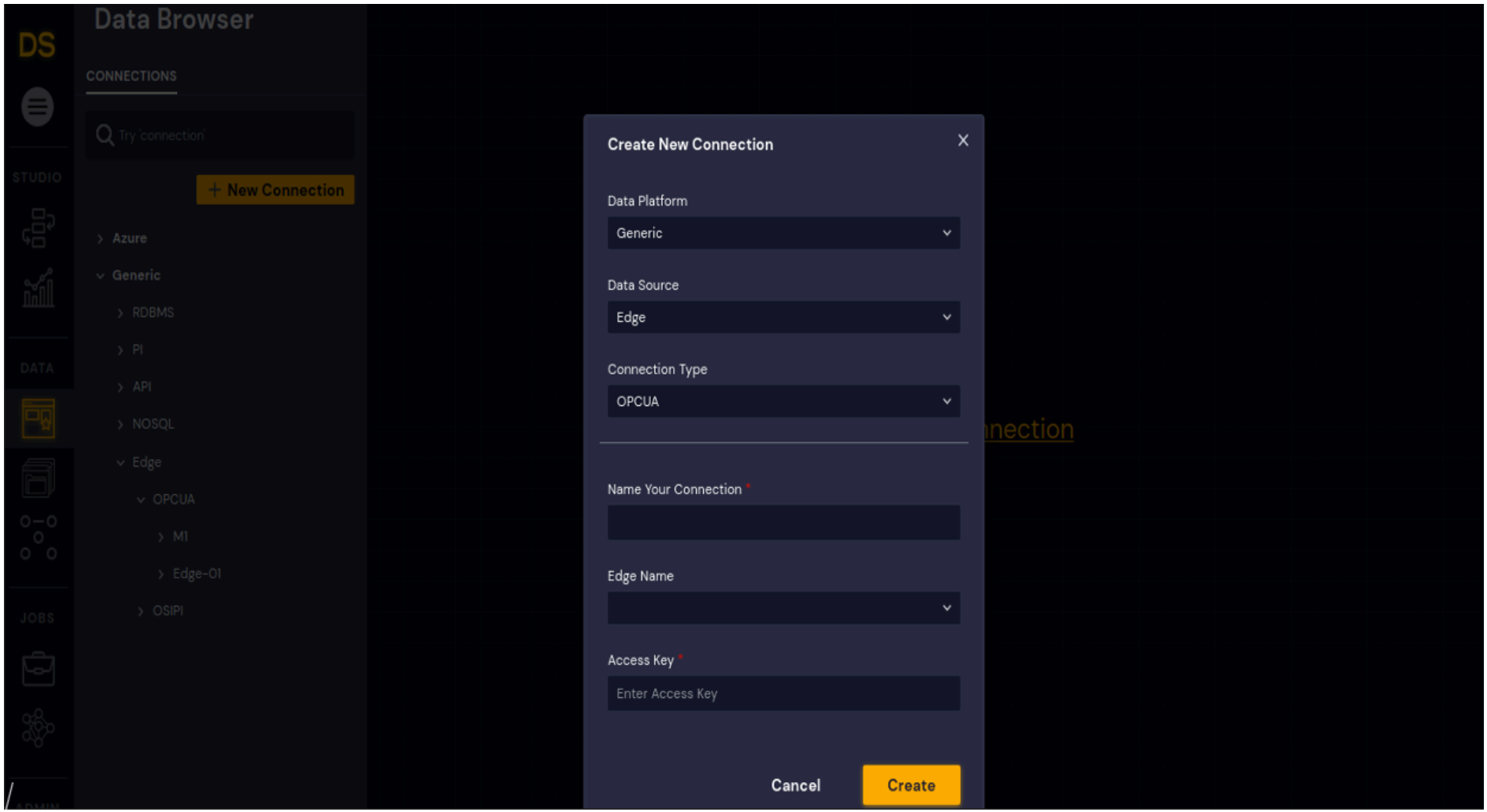

Figure 2: Creating Connection to an OPCUA Edge Connector

Once the data request is submitted, the DeepIQ Edge software processes this request by submitting it to the OPC UA server, as shown in Figure 3. The OPC UA server will publish the data into the configured Kinesis topic.

Figure 3: Requesting and Filtering for required tags from the Edge Connector.

You can monitor the process and find all the available tags in DataStudio, as shown in Figure 4.

Figure 4: Showing tags available in the connected Edge

Step2

DeepIQ's DataStudio supports data ingestion into AWS Timestream in both Spark Batch and Streaming modes.

Figure 5 is a sample workflow showing the AWS Timestream data ingestion in the streaming mode. We are reading data from AWS Kinesis, transforming the data into a TimestreamDB data model, and persisting data into an AWS TimestreamDB table.

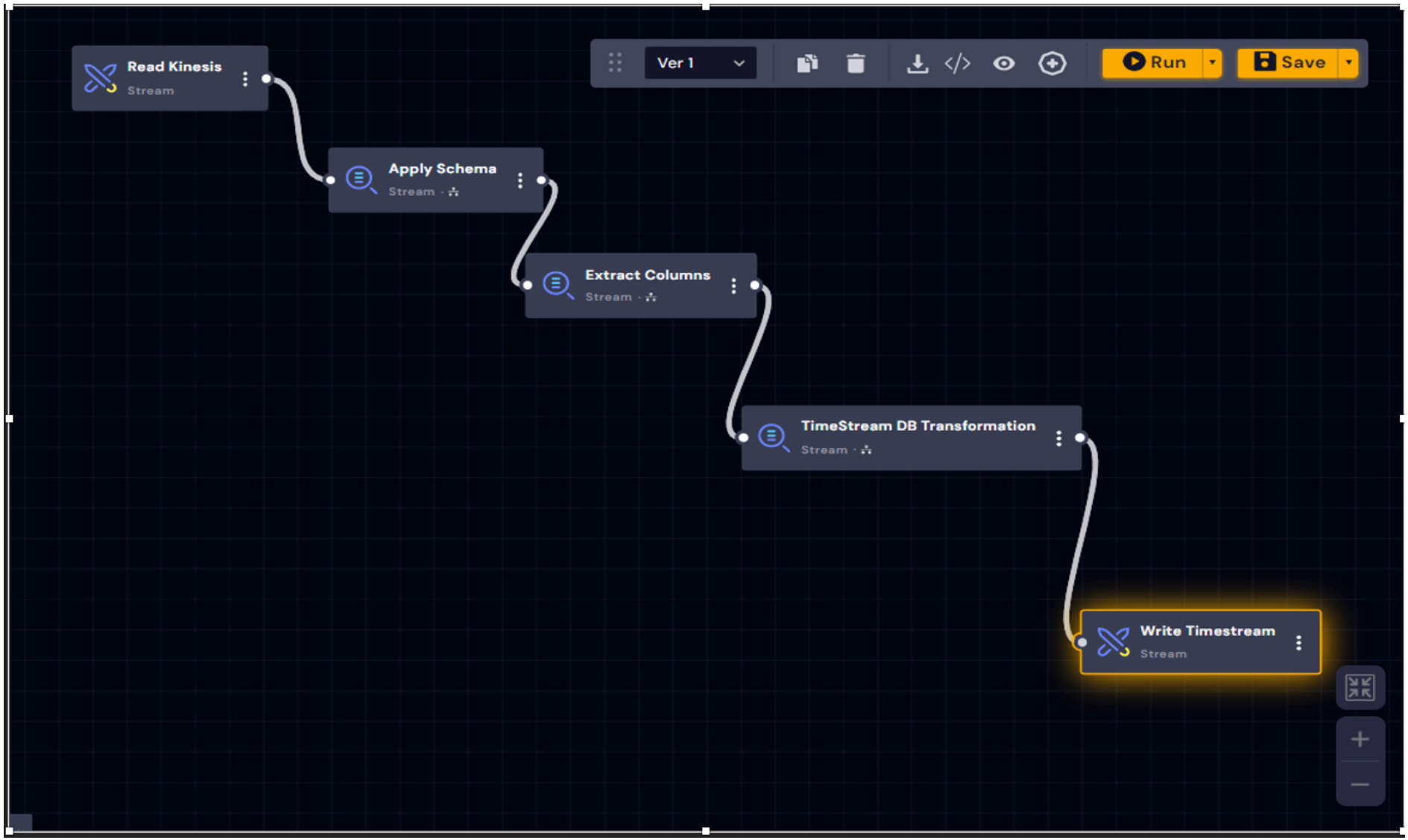

Figure 5: A simple workflow reading from an AWS Kinesis and writing to AWS TimestreamDB

Now, let us review the performance and scalability aspects of DeepIQ DataStudio components.

Figure 6 shows the performance metrics of an AWS TimestreamDB in Python mode. AWS TimestreamDB data writes in this single-threaded mode. Using this mode, the workflow took 33-seconds to write 10,000 tags into the database.

```
except Exception as err:
        print("__write_timestream_batch exception : ", str(err))
end_time=time.monotonic()
print("end time",end_time)
print("--- %s seconds ---" % timedelta(seconds=end_time - start_time))
```

```
write records :  9000 --> 9099 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9100 --> 9199 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9200 --> 9299 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9300 --> 9399 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9400 --> 9499 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9500 --> 9599 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9600 --> 9699 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9700 --> 9799 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9800 --> 9899 result : 200
writing data to timestream :  OPCUA . IOT10Ktags
write records :  9900 --> 9999 result : 200
end time 662536.832618339
--- 0:00:32.535327 seconds ---
```

Figure 6: A measure of TimestreamDB in single threaded Python mode

Figure 7 is a DataStudio workflow that writes to AWS TimestreamDB in high-performance batch mode. Using this workflow, we could ingest the identical 10,000 records in just 1 second. DataStudio uses a partitioned writer to submit records to TimestreamDB parallelizing the load to each executor, which helped us scale out and achieve a throughput of 10,000 tags per second. We can scale further by increasing the partitions and executors (cluster size) for larger datasets. We used a spark cluster with 20 cores and 70 GB to generate this performance.

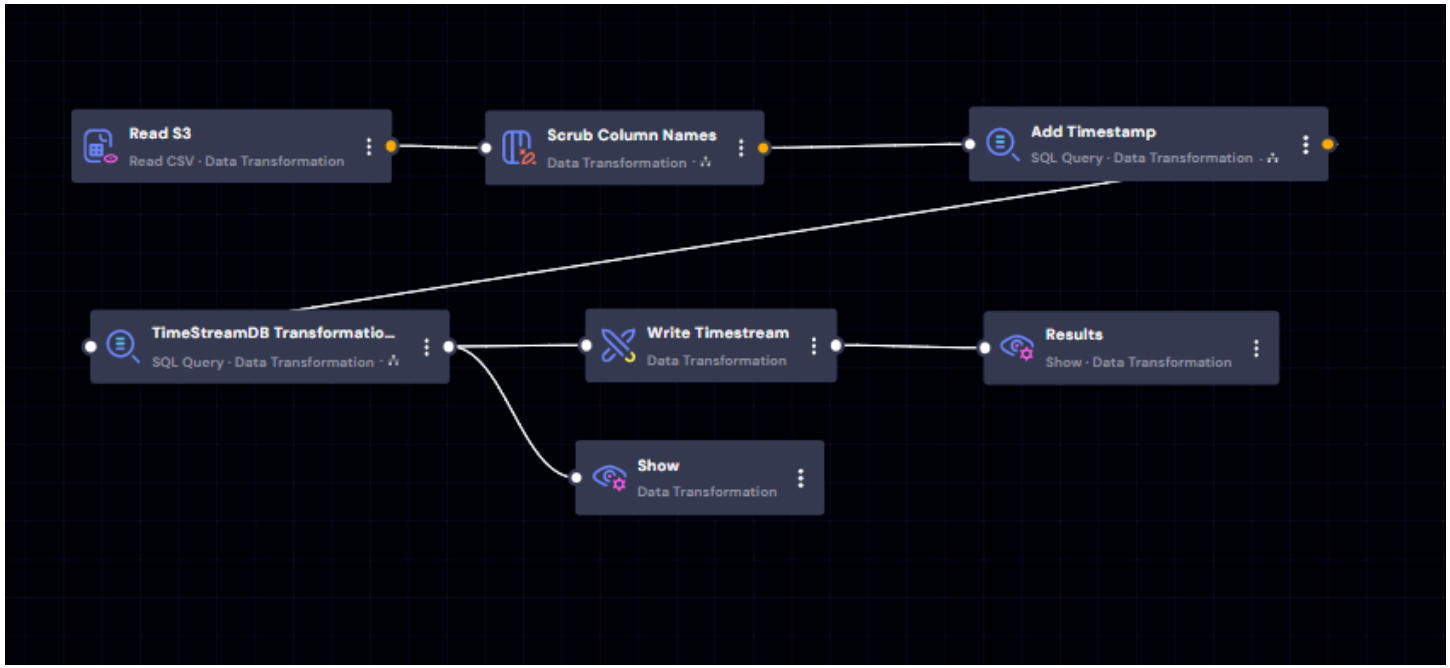

Figure 7: DataStudio Workflow writing to AWS TimestreamDB in high-performance Batch mode.

DeepIQ's DataStudio also offers components for smoothing, approximation, and interpolating time series data with various algorithms. Figure 8 shows a workflow that reads data from the raw storage table and creates a regular spaced data table after removing outliers. In this workflow, we use a cubic interpolation algorithm to interpolate data and a STD outlier removal algorithm to remove outliers.

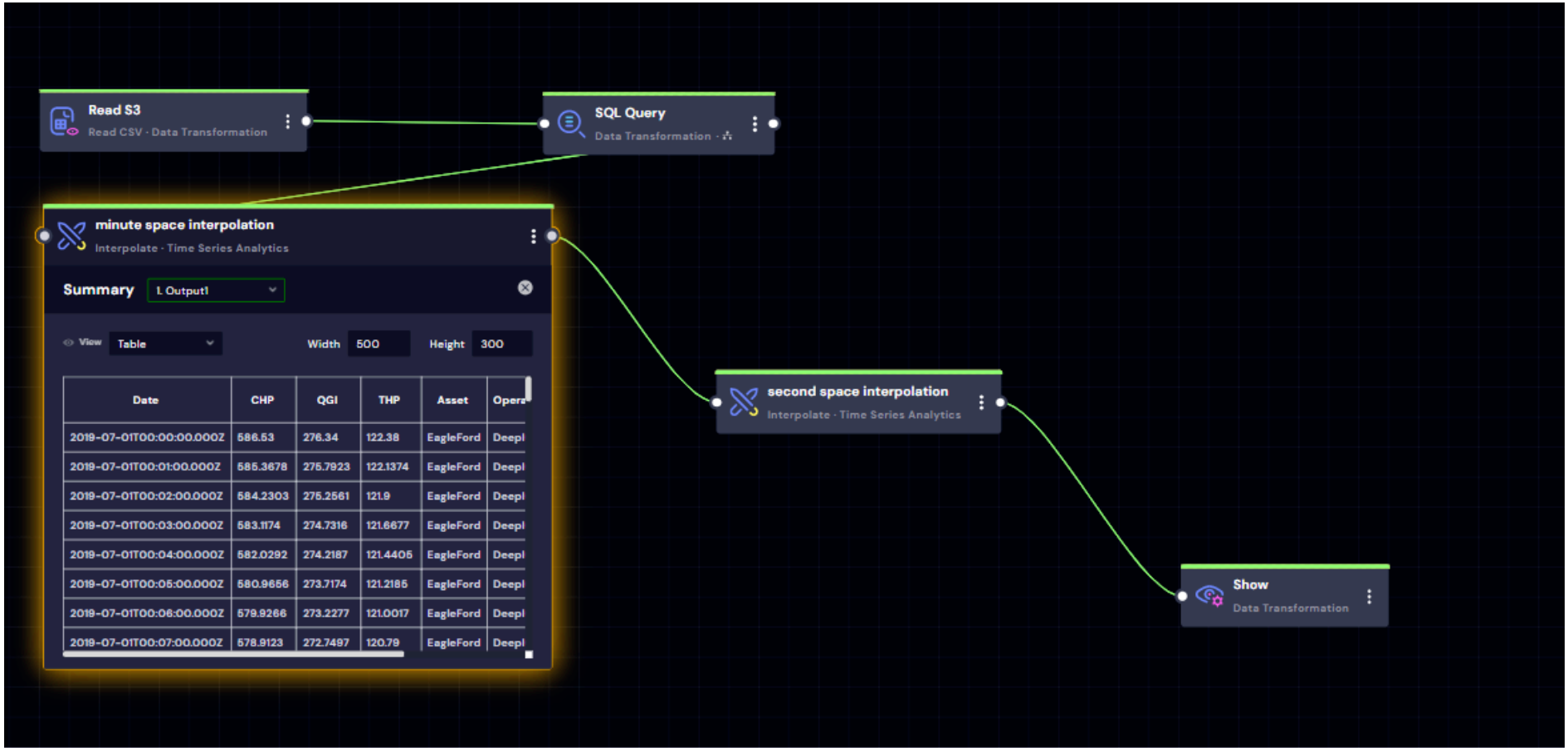

Figure 8: Workflow showing data cleansing and transformation activities.

Use Cases

Now that you have a data pipeline that moves clean time series data consistently into your time-series database, many powerful data apps can be built. In our previous whitepaper available at https://deepiq.com, we discussed how to implement predictive health models even with sparse failure data. In addition to streaming data directly into the database, DeepIQ allows you to run streaming analytics by connecting to your AWS Kinesis using Spark Structured streaming to calculate real-time KPIs. Resultant Real-Time alerts and KPIs can be visualized using AWS Quicksight.

Conclusion

'S Timestream is a powerful new addition to time series capabilities of AWS cloud. With DeepIQ's DataStudio, you can realize the full potential is new database by ingesting your operational data at scale and using it as a data source for cutting-edge analytics. For more information,

please get in touch with info@deepiq.com.

DeepIQ is on a mission to transform industrial processes by digitizing industrial expertise. Our vision is to drive end-to-end automation, enabling systems such as self-running power plants or drilling rigs using generative AI as the higher order reasoning layer operating over existing industrial automation technology stack.

Privacy Policy Terms of Services

FEATURES

- Extract
- Engineer
- Explore

INDUSTRIES

- Upstream
- Midstream
- Downstream & Chemical
- Mining

SOLUTIONS

- IT-OT Contextualization
- Well Construction Optimization
- P & ID Digitization
- Predictive Maintenance
- Production Optimization
- Route Optimization
- Sustainability

PARTNERS

- AWS
- Azure
- Cloudera
- Google
- Databricks
- OSIsoft(AVEVA)
- Snowflake

COMPANY

- About Us

+1 (832) 391-6834 info@deepiq.com