leaplogic

Automated workload transformation from PySpark to Databricks

VIDEO TRANSCRIPT

The freedom of the cloud is on the horizon, but the path to modernization is challenging.

We change that!

LeapLogic can accelerate the transformation of legacy workloads to Databricks with up to 95% automation, helping you reap the benefits of a modern Databricks-native stack.

Here's a demo of how LeapLogic simplifies and automates the transformation of PySpark scripts to Databricks Delta Live Tables (DLT) pipelines written in Python.

For this demo, we have considered a sub-set of organization management data, which consists of three entities – employee, department, and employee leave balance. We will process this data per the data lake architecture, where raw data will be used to create the bronze data layer.

To automate the transformation of PySpark scripts to Databricks DLT pipeline scripts, simply head to your LeapLogic dashboard and visit its Transformation section. Here, you can transform a single query or all workloads in bulk at one go.

Since we have to transform multiple scripts at the same time, we have selected Bulk Transformation. Simply provide a transformation name and select ETL accessories as the workloads category. Here, the source and target platforms will be PySpark and Delta Live Table.

You can now proceed to upload files. Browse and load the PySpark scripts to transform and click Execute. Under the Transformation section, you can choose to transform a single file or all files in one go. LeapLogic will notify you once your PySpark files are automatically transformed to DLT scripts.

Click on the Compare icon to quickly view the difference between the original and the transformed scripts. For instance, for the transformation of the first script, an "Import DLT" statement is added to import the functionality of the DLT package, and several new functions are added. For the transformation of the second script, we can see that the Delta Live Table created in the previous script is used in the definition of the transformed code. This showcases how LeapLogic identifies various relationships in the original scripts and maps them into corresponding DLT Scripts.

You can simply download the transformed DLT scripts from LeapLogic for offline access.

Now, let's run the pipeline in a Databricks environment to validate the transformed code. Go to your Databricks dashboard and create a notebook. Provide a name to your notebook and simply paste the transformed DLT scripts to the Databricks interface.

After submitting all scripts, go to your workflows. Visit the Delta Lake Tables section and create a new pipeline. Enter the name of your pipeline and select the notebook that you recently created under the Select Source field. Click Create to let Databricks automatically create your pipeline. Now, click Start and wait for a while as Databricks initializes your pipeline, sets up tables, and completes the rendering process.

You can see that Databricks has successfully identified the topological order of jobs and rendered it in the form of a graph. Here, you can observe the flow of data from raw to bronze, silver, and finally gold layer that can help you gain valuable insights from your data.

LeapLogic also automates rigorous validation tests and handles orchestration so that you can modernize your legacy PySpark workloads to Databricks without any business disruption.

And when you do, LeapLogic assists with Databricks-specific optimization and capacity planning, ensuring the performance of transformed workloads in your new environment.

Choose LeapLogic to migrate workloads from PySpark to Databricks – faster, at a lower cost and with lower risk.

It's more than the next step. It's a leap into the future of your business.