

# Spark Execution Properties Settings

When you specify a job in the Run Job page, you may pass to the Spark running environment a set of Spark property values to apply to the execution of the job. These property values override the global Spark settings for your deployment.

**NOTE:** A workspace administrator must enable the Custom Spark Options feature in the Workspace Settings page. For more information, see *Workspace Settings Page*.

Spark overrides are applied to individual output objects.

- You can specify overrides for ad-hoc jobs through the Run Job page.
- You can specify overrides when you configure a scheduled job execution.

In the Run Job page, click the Advanced Execution Settings caret.

The screenshot shows the 'Run Job' page with the 'Running Environment' section. Two options are listed: 'Photon' (Run job on Trifacta Photon) and 'Spark' (Run job on Spark), with 'Spark' selected. Below this is the 'Advanced environment options' section, which is expanded to show 'Spark Execution Properties'. The 'Transformer Dataframe Checkpoint Threshold' is set to 200 (default 200). The 'Enable whole-stage code generation for Spark' checkbox is checked. The 'Maximum number of fields that whole-stage code generation supports' is set to 100 (default 100).

Run Job

Running Environment

- Photon  
Run job on Trifacta Photon (best for small and medium-sized jobs, up to approximately 1 GB of data)
- Spark  
Run job on Spark

Advanced environment options ^

Spark Execution Properties ?

Transformer Dataframe Checkpoint Threshold Default value: 200

200

Enable whole-stage code generation for Spark

Maximum number of fields that whole-stage code generation supports Default value: 100

100

**Figure: Spark Execution Properties**

Spark parameter	Description
Transformer Dataframe Checkpoint Threshold	<p>When checkpointing is enabled, the Spark DAG is checkpointed when the approximate number of expressions in this parameter has been added to the DAG. Checkpointing assists in managing the volume of work that is processed through Spark at one time; by checkpointing after a set of steps, the Trifacta platform can reduce the chances of execution errors for your jobs.</p> <p>By raising this number:</p> <ul style="list-style-type: none"> <li>• You increase the upper limit of steps between checkpoints.</li> <li>• You may reduce processing time.</li> <li>• It may result in a higher number of job failures.</li> </ul> <p>Default value:200</p>
Enable whole-stage code generation for Spark	<p>When enabled, whole-stage code generation optimizes Spark SQL queries for execution performance on the cluster.</p>
Maximum number of fields that whole-stage code generation supports	<p>This defines the number of fields (columns) that are permitted in a whole-stage code generation query. If the number of fields in the query exceeds this value, then the Trifacta platform disables whole-stage code generation to prevent performance issues and memory exceptions.</p> <div data-bbox="354 701 1455 783" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p><b>NOTE:</b> Avoid modifying this value unless you have a clear understanding of the implications.</p> </div> <p>Default value: 100</p>