Configure Batch Job Runner

Contents:

The Trifacta® platform utilizes the batch job runner service to orchestrate jobs that are executed on the selected backend running environment. This service passes jobs to the backend and tracks their progress until success or failure. This service is enabled by default.

Configure Timeout

Setting	Default Value	Description
`batchserver.spark.requestTimeoutMillis`	`120000` (2 minutes)	Maximum number of milliseconds that the Batch Job Runner service should wait for a response from the Spark Job service during job execution. Default is 2 minutes.

Configure polling intervals

The following parameters can be modified to change the Batch Job Runner polling intervals for various types of jobs.

Setting	Default Value	Description
`jobMonitoring.ingestPollFrequencySeconds`	`3`	Polling interval in seconds for Batch Job Runner to check for status of ingest jobs.
`jobMonitoring.publishPollFrequencySeconds`	`3`	Polling interval in seconds for Batch Job Runner to check for status of publishing jobs.
`jobMonitoring.wranglePollFrequencySeconds`	`3`	Polling interval in seconds for Batch Job Runner to check for status of wrangling jobs.
`jobMonitoring.maxHeartbeatDelayMilliseconds`	`7200000` (2 hours)	Duration in milliseconds for the service to wait for a job heartbeat before failing.

Configure Job Threads

As needed, you can configure the number of worker threads assigned to each process that is managed by the batch job runner. Depending of the volume and complexity of jobs that you run of each type, you may choose to modify these settings to improve performance for key job types.

Copyright © 2022 Trifacta Inc.

Tip: These settings can be configured through the Admin Settings page in the Trifacta application. See *Admin Settings Page*.

By running environment:

Setting	Default Value	Description
`batchserver.workers.photon.max`	2	Number of worker threads for running Trifacta Photon jobs. This value corresponds to the maximum number of photon jobs that can be queued at the same time. For more information, see *Configure Photon Running Environment*.
`batchserver.workers.spark.max`	16	Number of worker threads for running Spark jobs. For more information, see *Configure Spark Running Environment*.
`batchserver.workers.wrangle.max`	16	Number of worker threads for running transformation jobs.

By job type:

Setting	Default Value	Description
`batchserver.workers.ingest.max`	16	Maximum number of worker threads for running ingest jobs, which are used for loading relational data into the platform. After this maximum number has been reached, subsequent requests are queued.
`batchserver.workers.profile.max`	16	Maximum number of worker threads for running profile jobs, which provide summary and detail statistics on job results.
`batchserver.workers.publish.max`	16	Maximum number of worker threads for running publish jobs, which deliver pre-generated job results to other datastores.
`batchserver.workers.fileconverter.max`	16	Maximum number of worker threads for running fileconverter jobs, which are used to convert source formats into output formats.
`batchserver.workers.filewriter.max`	16	Maximum number of worker threads for running filewriter jobs, which are used for writing file-based outputs to a specified storage location.

Depending on your running environment, there may be additional parameters that you can configure to affect Batch Job Runner for that specific environment:

- *Configure for Spark*
- *Configure Photon Running Environment*

Configure BJR for EMR

Multiple BJR instances

If the Trifacta platform is connected to an EMR cluster, multiple instances of the batch job runner are deployed to manage jobs across the cluster so that if one fails, YARN jobs are still tracked. No configuration is required.

YARN logs from EMR

The following properties below can be modified for batch job runner:

Setting	Default Value	Description
`aws.emr.getLogsOnFailure`	`false`	When set to `true`, YARN logs from all nodes in the EMR cluster are collected from S3 and stored on the Trifacta node in the following location: `/opt/trifacta/logs/jobs/<jobId>/container` where: `<jobId>` is the Trifacta platform internal identifier for the job that failed.
`aws.emr.getLogsForAllJobs`	`false`	When set to `true`, YARN logs from nodes in the EMR cluster are collected and stored in the above location for all jobs, whether they succeed or fail. **NOTE:** This parameter is intended for debugging purposes only.

Configure Database

Configure database cleanup

By default, the Jobs database, which is used by the batch job runner, does not remove information about jobs after they have been executed.

Logging:

- Batch Job Runner activities are surfaced in `batch-job-runner.log`. For more information, see *Configure Logging for Services*.
- Logging information for individual jobs is available in the `job.log` file written in the job directory. For more information, see *Diagnose Failed Jobs*.

As needed, you can enable the Trifacta platform to perform cleanup operations on the Jobs database.

> **NOTE:** If cleanup is not enabled, the Jobs database continues to grow. You should perform periodic cleanups in conjunction with your enterprise database policies.

Steps:

To enable cleanup of the Jobs database, please complete the following steps.

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following settings and set them accordingly:

Settings	Description
batch-job-runner.cleanup.enabled	Set this value to `true` to enable this feature.
batch-job-runner.cleanup.interval	The interval in ISO-8601 repeating intervals format at which batch-job-runner should clean outdated information about jobs. Default value is `R/PT1H`, which means that the job is executed once per hour.
batch-job-runner.cleanup.maxAge	The retention time for deployments in ISO-8601 interval format after which information about jobs is considered outdated. Default value is `P14D`, which means that the jobs information is cleaned out after 14 days.
batch-job-runner.cleanup.maxDelete	Maximum number of jobs whose information can be deleted per cleanup pass. Default value is `1000`.

For more information on ISO-8601 interval format, see *https://en.wikipedia.org/wiki/ISO_8601#Repeating_intervals*.

3. Save your changes and restart the platform.

Configure Jobs database

The Batch Job Runner utilizes its own Jobs database. For more information, see *Configure the Databases*.

Logging

For more information on logging for the service, see *Configure Logging for Services*.

Copyright © 2022 Trifacta Inc.