**📅 QuantumStream Scheduling Tasks – Configuration & Execution Guide**

**📦 Feature:** Task Scheduling via Configuration\ **🛠️ Available Since:** v2.5.0\ **📅 Last Updated:** July 2025\ **🎯 Objective:** *"Automate recurring operations using cron-based task definitions within your QuantumStream configuration."*

**🧠 Overview**

QuantumStream supports **cron-based task scheduling** directly within the project configuration file. This allows users to define and automate recurring operations such as data ingestion, stream restarts, model retraining, and plugin execution without relying on external schedulers.

The **Scheduler module** reads from the tasks section of the qs.config.yaml file and manages execution based on defined cron expressions.

**🧩 Key Features**

* **Cron Expression Support**\ Define execution intervals using standard cron syntax (e.g., 0 0 \* \* \* for midnight daily).
* **Task Types Supported**
  + Start/stop/restart streams
  + Run ingestion jobs
  + Trigger model retraining
  + Execute custom plugins
  + Generate reports
* **Integrated Execution Engine**\ The built-in Scheduler module handles timing, execution, and logging of all scheduled tasks.

**⚙️ Defining Scheduled Tasks**

Add a tasks section to your qs.config.yaml file:

tasks:

- name: "daily\_ingestion"

cron: "0 2 \* \* \*"

action: "ingest"

params:

source: "s3://data/daily/"

stream: "user\_activity\_stream"

- name: "weekly\_model\_retrain"

cron: "0 4 \* \* 1"

action: "retrain\_model"

params:

model\_id: "churn\_predictor\_v3"

dataset: "s3://models/training\_data.csv"

- name: "stream\_restart"

cron: "0 \*/6 \* \* \*"

action: "restart\_stream"

params:

stream: "realtime\_orders"

Each task must include:

* name: Unique identifier
* cron: Cron expression for scheduling
* action: Task type to execute
* params: Task-specific parameters

**🛠️ Supported Actions**

| **Action** | **Description** |
| --- | --- |
| start\_stream | Starts a specified stream |
| stop\_stream | Stops a running stream |
| restart\_stream | Restarts a stream instance |
| ingest | Triggers a data ingestion job |
| retrain\_model | Launches a model retraining pipeline |
| run\_plugin | Executes a registered plugin |
| generate\_report | Runs a report generation task |

**📊 Monitoring & Logs**

* Task execution logs are stored in logs/scheduler.log
* Use qs monitor to view upcoming and recently executed tasks
* Failed tasks are flagged with error codes and stack traces

**🔐 Security & Access Control**

* Only users with scheduler\_admin or config\_editor roles can modify scheduled tasks
* Secrets used in task parameters should be referenced via environment variables or secret managers
* Task execution is sandboxed and audited

**🧭 FAQs**

**Q: Can I disable a task without deleting it?**\ A: Yes. Add enabled: false to the task definition.

**Q: Does the Scheduler support time zones?**\ A: Yes. You can add a timezone field (e.g., timezone: "UTC").

**Q: Can I trigger a scheduled task manually?**\ A: Yes. Use qs scheduler run <task\_name> to execute it on demand.

**Q: How do I validate my task configuration?**\ A: Run qs validate to check for syntax errors and unsupported actions.

QuantumStream’s task scheduling system gives you the power to automate complex workflows with simple, declarative configuration—keeping your data operations running smoothly and on time.