



Queues and cron

Workflow

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-08-25

SOURCE
[Open this article online](#)

Queues and cron

Live workflows rely on two Magento queue consumers and one cron job.

Component	Purpose
<code>workflow.event.trigger.consumer</code>	Processes registered event messages and matches them to published definitions.
<code>workflow.run.start.consumer</code>	Starts or resumes workflow runs and advances their nodes.
<code>workflow_resume_due_runs</code>	Runs each minute and queues waiting runs whose resume time has arrived.

Run the consumers through the same supervised process used for the store's other Magento consumers. For a manual diagnostic, Magento can start an individual consumer with `queue:consumers:start`, but a long-running production setup needs process supervision and restart handling.

```
php bin/magento queue:consumers:start  
workflow.event.trigger.consumer  
php bin/magento queue:consumers:start  
workflow.run.start.consumer
```

Health check

1. Confirm that Magento cron runs successfully.
2. Confirm that both consumers are configured and remain alive.
3. Publish a small controlled workflow.
4. Trigger its Magento event.
5. Verify that the definition grid receives a latest-run status.
6. For a Human Task, decide it and verify that the run continues.
7. For a short Wait in staging, verify that cron queues it after the due time and the run consumer completes it.

Queue backlog explains several misleading symptoms: a real order exists but no task appears; a decision is saved but the run stays waiting; or a wait passes without continuation. Check the correct stage before changing the definition.

The runtime retry and batch limits are global settings. Increasing them is not a substitute for fixing a repeatedly failing node or undersized worker pool.