expqueue: the job queue

Source file: doc/eng-us/features/6.0/extensions/expqueue.md Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/features/6.0/extensions/expqueue.md

Read this page if an extension you use queues work (Exponential Intelligence does) or if you want to run your own background jobs. The extension is the Composer package `se7enxweb/expqueue`, required by `composer.json` from 6.0.15 on (`~0.1.0`), so it is in `extension/expqueue` on every installation and stays inactive until it is activated. Its full documentation lives with it, in the expqueue repository.

In short

What it does	A generic, database-backed job queue. Producers queue a job with one call; workers run it on the command line, from cron or as a systemd service, never inside a web request. Extensions register job types and their handlers in `expqueue.ini`.
Scheduling	Seven priority classes with weights, fair scheduling with aging and a minimum share per class, quotas per user and site, concurrency limits per resource that adapt to rate-limit hints
Reliability	Leases with a reaper, retries with backoff, a dead letter, idempotency keys, dependencies, not-before times, job groups with progress, cancel, pause and resume, a result checkpoint for handlers
Admin	Setup > Job queue (dashboard, jobs, groups, controls); policies `expqueue/view` and `expqueue/manage`
Engines	MySQL/MariaDB, PostgreSQL, SQLite, Oracle and MongoDB
Release	0.1.0

expqueue is separate from the kernel's content jobs, which run large content operations (removes, copies, moves, package installs, audit exports) as jobs of their own.

Switch it on

1. Activate it globally, so producers on every siteaccess and the workers see it, in `settings/override/site.ini.append.php`: `[ExtensionSettings] ActiveExtensions[]=expqueue`.
2. Create the tables (idempotent; also the upgrade):

   Shell

   ```
   ./console ext:expqueue:install-schema
   ```

3. `php bin/php/ezpgenerateautoloads.php -e` and `./console exp:velocity deploy` (or clear the INI, template and translation caches and reload PHP-FPM by hand).
4. A worker, as the site user, every minute:

   Cron

   ```
   * * * * * cd /path/to/installation && php runcronjobs.php -q expqueue >> var/log/cronjob-expqueue.log 2>&1
   ```

   When the job types of an extension are active for some siteaccesses only (expai activated

per admin siteaccess), add `-s <that siteaccess>`, or the worker does not know them. A systemd service (`./console ext:expqueue:service install`) can run next to cron for shorter waits.

5. Check: `./console ext:expqueue:enqueue expqueue.noop`, `./console ext:expqueue:work --once`, `./console ext:expqueue:status`, or Setup > Job queue > "Send a test job".

The complete steps: the installation guide; the API for producers, scheduling, workers and settings: the documentation.

Related pages

- expai, Exponential Intelligence, which runs its queued work here
- Content jobs
- Installing Exponential, section 10
- Extensions index

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.