Monitoring the database

What to monitor on the database of an Exponential site and where to read it - the Database card of Setup > System information, exp:system:info for scripts and alerts, the log files and the patterns that matter, the…

Source file: `doc/eng-us/help/database-administrators/monitoring-the-database.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/database-administrators/monitoring-the-database.md

This page is for database administrators who set up monitoring and alerting for the database of an Exponential site. It answers: what the application itself reports about its database, how to read it from a script, which log lines deserve an alert, and which checks belong to the engine. The application's view and the server's view complement each other: the first tells you whether the site can use the database, the second how the server is doing.

Contents

1. Setup > System information
2. From a script: exp:system:info
3. Log files and patterns
4. Queues
5. Per engine
6. Checklist

Setup > System information

The page needs the `setup/system_info` policy and only reads (System information):

- **Database card**: engine and server version, driver class, the file (SQLite) or server and database name, character set, number of tables, size (SQLite always; MySQL and PostgreSQL with **Measure sizes**).
- **Database connection** part: type, server, socket, database name, character set, read replica.
- **Database queries** box: the SQL query cache mode, hit rate and the SQL profiler switch.
- **Health checks** first, failures and warnings before notes; a warning when the connection character set is not UTF-8.
- **Download report** (text or JSON) for a support request; passwords and the database user are never in it.

Open it through each server that serves the site (Apache with PHP-FPM, Velocity's port): each has its own PHP and may report differently.

From a script: exp:system:info

Shell

```
./console exp:system:info --allow-root-user            # the text report
./console exp:system:info --json --allow-root-user     # JSON for your monitoring system
./console exp:system:info --sizes --allow-root-user    # with directory and database sizes
```

It reports the command-line PHP and **exits 1 when a check fails**, so it can feed an alert or guard a deploy. `exp:expfetch` (the system card) shows version, engine and database on one screen with hints.

Log files and patterns

The logs are in `var/log/` (`error.log`; `warning.log` only with `[DebugSettings] AlwaysLog[]=warning`). `exp:logs` lists, follows and searches them (Log viewer and rotation).

Pattern	Meaning	Alert
`Database implementation not supported`	mistyped `DatabaseImplementation`; every page fails	immediately
`eZDBNoConnectionException`, "database cannot be reached"	wrong host, credentials, or the server is down	immediately
`database is busy: the transaction could not start`	SQLite write queue timed out (Write queue)	when repeated
`attempt to write a readonly database`	SQLite file ownership (Permissions)	immediately
`could not set the connection charset`	MySQL connection charset fell back	at setup
`MONGO TODO`	SQL the MongoDB adapter could not translate	review daily
`ORA-`	an Oracle error	by number

Shell

```
grep -c "database is busy" var/log/error.log
grep -rn "MONGO TODO" var/log/ | tail
```

Queues

Check	Healthy
`SELECT action, COUNT(*), MIN(created) FROM ezpending_actions GROUP BY action;`	small counts, recent `MIN(created)`
expired rows in `ezsession` (database sessions)	few
Setup > Cronjobs, recent runs	no issues, last run recent

See Housekeeping: cronjobs and queue tables.

Per engine

Engine	From Exponential	From the engine
SQLite	`PRAGMA quick_check`, `PRAGMA journal_mode` (`wal`), size of `-wal`, disk space, file ownership (SQLite maintenance)	
PostgreSQL	System information	connections, locks, replication lag, autovacuum (`pg_stat_*` views)
MySQL, MariaDB	System information; replica in use	connections, InnoDB status, replica lag
MongoDB	`MONGO TODO` count	server status, replica set health
Oracle	`ext:ezoracle:health`, the hourly `ezoraclehealth` cronjob, which mails `HealthMailReceivers[]` on warnings (Oracle maintenance commands)	AWR, OEM

Also monitor the backups themselves: their age, size and the last restore test (Backups per engine).

Checklist

- ☐ `exp:system:info --json` polled; exit code 1 alerts.
- ☐ Alerts on the immediate log patterns above.
- ☐ Queue tables checked daily.
- ☐ Engine-side monitoring for connections, locks, replication and disk.
- ☐ Backup age monitored.

Related pages

- Monitoring overview, Health checks and uptime
- Troubleshooting database errors
- System information
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **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.