Capture only slow ticks with a bounded spark profile

An average profile can hide a short pause among many healthy ticks. Spark can focus a profile on ticks exceeding a chosen duration. This helps investigate a repeatable spike, but its percentages describe the captured slow ticks rather than the entire playing session.

Record the starting point

Record the server version, spark version and the approximate spike frequency. Choose a representative period and a consenting tester for the triggering action. Keep a normal health report or unfiltered profile as a separate baseline so the filtered result has context.

Work through the change

1. Check the installed command help and start a bounded recording, for example `/spark profiler start --timeout 120 --only-ticks-over 100`. The example selects ticks over 100 milliseconds; choose a threshold relevant to the observed pause.
2. Repeat the suspected action once or wait for the expected spike. Note its timestamp and player count, without adding unrelated stress to the server.
3. Open the resulting viewer and confirm that slow ticks were actually captured. Compare the busiest branches with the action time and the server's tick measurements.
4. Make one justified plugin or workload adjustment on a test copy, then record the same filtered test again. Retain the original capture for comparison.

Check the result

A useful capture contains the reported pause and enough context to identify candidate work. A branch dominating filtered samples may still be rare overall. Compare its duration and frequency with normal gameplay before describing it as the server's constant bottleneck.

If the result differs

If the profile is empty, the threshold may exceed the observed ticks or the event may not have occurred. Do not lower every performance setting merely to obtain a busier chart. Repeat during a representative window with a documented threshold.

Cancel an unnecessary recording using the installed profiler controls and revert only the trial workload change if it harms gameplay. Profiling does not require deleting worlds or resetting plugin data. Share the report only after reviewing its metadata.

References and related help

- spark maintainer reference 1
- spark maintainer reference 2
- Diagnose Minecraft lag with TPS, MSPT and spark
- Find Minecraft logs and send useful troubleshooting details

---

Revision #1
Created 2026-11-09 16:36:26 UTC by CloudExa Hosting Admin
Updated 2026-11-09 16:36:26 UTC by CloudExa Hosting Admin