Tune Minecraft view distance and simulation distance

Players seeing distant terrain and the server processing nearby activity are different workloads. This guide helps you adjust Minecraft Java distance settings without treating every performance problem as a need for more RAM. Make a small measured change, then compare the result under similar conditions.

Choose the problem you are testing

`view-distance` controls the world data sent around a player. `simulation-distance` controls the range of entity simulation. Both are chunk distances around the player, rather than the full width. Client rendering settings can also limit what an individual player sees.

Describe the symptom before changing anything: terrain arrives slowly, players see a short horizon, or active areas produce slow ticks. Ask whether everyone is affected and whether the problem happens while exploring or while standing still. A client with low frame rate needs a different investigation from a server whose tick time rises.

Save a baseline

1. Record the current distances, Minecraft version and server software.
2. Record player count and what those players are doing. Exploring new terrain is not the same test as standing at spawn.
3. Measure TPS and MSPT using the available profiling tools. Keep a short report from the affected period.
4. Stop the server normally and download the configuration files you will edit.

Choose a quiet window for the restart and explain the test to players. Keep the original values in your notes so reversing the setting is straightforward. Avoid combining the distance change with a plugin update, map generation job or Java change: several simultaneous changes make the comparison less useful.

Change the effective configuration

In `server.properties`, locate the existing distance lines. For example, `view-distance=8` with `simulation-distance=6` is a test configuration, not a universal recommendation. Keep values within the supported range for your installed version.

On Paper, check `spigot.yml` for world-specific distance overrides. A value of `default` or `-1` inherits the corresponding server property. Preserve other worlds' settings unless you intend to change them too. Use the reference matching your installed build rather than copying an old Paper configuration file.

Save and start the server. Read startup warnings before joining. If a value is rejected or another configuration layer overrides it, resolve that first; repeating a test with a value that never applied proves little.

Compare the result fairly

Repeat the same route or activity with a similar number of players. Check tick time, terrain delivery and player experience. Record the new values and the observation period. Lowering distances can affect how players use farms and travel, so ask the affected players to test their normal activity before keeping the change.

Do not increase both values to their maximum because the server appears idle with nobody online. Separated players can create a substantially different workload. A pre-generated map also does not remove the cost of processing active gameplay.

If there is little improvement, restore your baseline and investigate the profiling evidence. See TPS, MSPT and spark. Further reading: Paper's per-world overrides and distance properties.

Revision #2
Created 2026-11-08 21:45:13 UTC by CloudExa Hosting Admin
Updated 2026-11-08 21:45:13 UTC by CloudExa Hosting Admin