

WebbMotell - ISPConfig

Hur använder jag Redis för WordPress

[Redis cache](#) ingår i våra Premium planer.

Aktivera Redis Cache för WordPress

Innan Redis installeras, installera och aktivera Query Monitor-plugin.



Query Monitor

Query Monitor is the developer tools panel for WordPress.

By John Blackbourn

[Install Now](#)

[More Details](#)

★★★★★ (377)

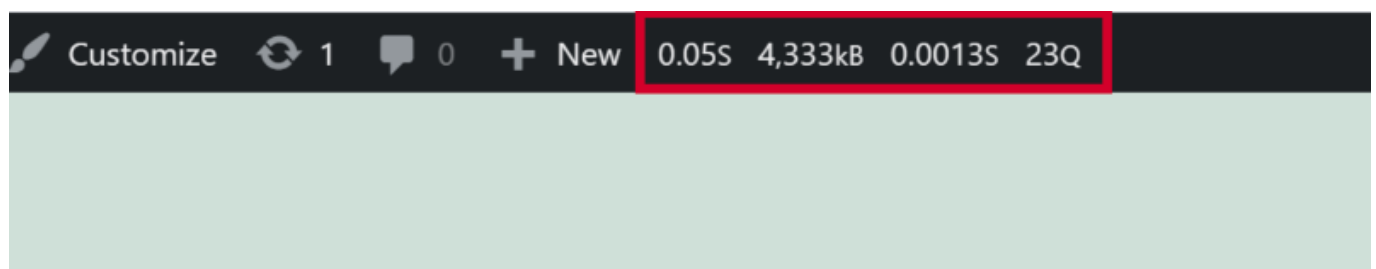
100,000+ Active Installations

Last Updated: 2 months ago

✓ Compatible with your version of WordPress

Genom att lägga till [Query Monitor](#) kan databasfrågor, PHP etc felsökas. Det hjälper också till att visa totala antalet databasfrågor grupperade efter akutell plugin som gjort förfrågan.

Inloggad som admin, gå till förstasidan på er WordPress hemsida, alltså inte admin sidan utan den vanliga hemsidan. Högst upp på sidan visas nedanstående rad (om ni är inloggad som admin). Klicka på Query Monitor.



Detta startar en Query Monitor panelen.

WebbMotell - ISPConfig

Query Monitor				
GET https://18.216.117.75/ → 200				
Overview				
Queries				
Object Cache				
Logs				
Request				
Template				
Scripts (9)				
Page Generation Time	Peak Memory Usage	Database Queries	HTTP API Calls	
0.0540	4,333 kB	0.0013	None	
0.2% of 30s limit	0.8% of 524,288 kB server limit 10.6% of 40,960 kB WordPress limit	Total: 23		

Just detta exempel är en ganska enkel sida, och här kan man se totalt 23 databasanrop som tar 0,0013 millisekunder att köra.

Lägg nu till ett Redis Object Cache plugin. Sök på Redis och välj det här specifika plugin-programmet **Redis Object Cache**.



Redis Object Cache

A persistent object cache backend powered by Redis. Supports Predis, PhpRedis, Credis, HHVM, replication, clustering...

By Till Krüss

[Install Now](#)

[More Details](#)

 (99)

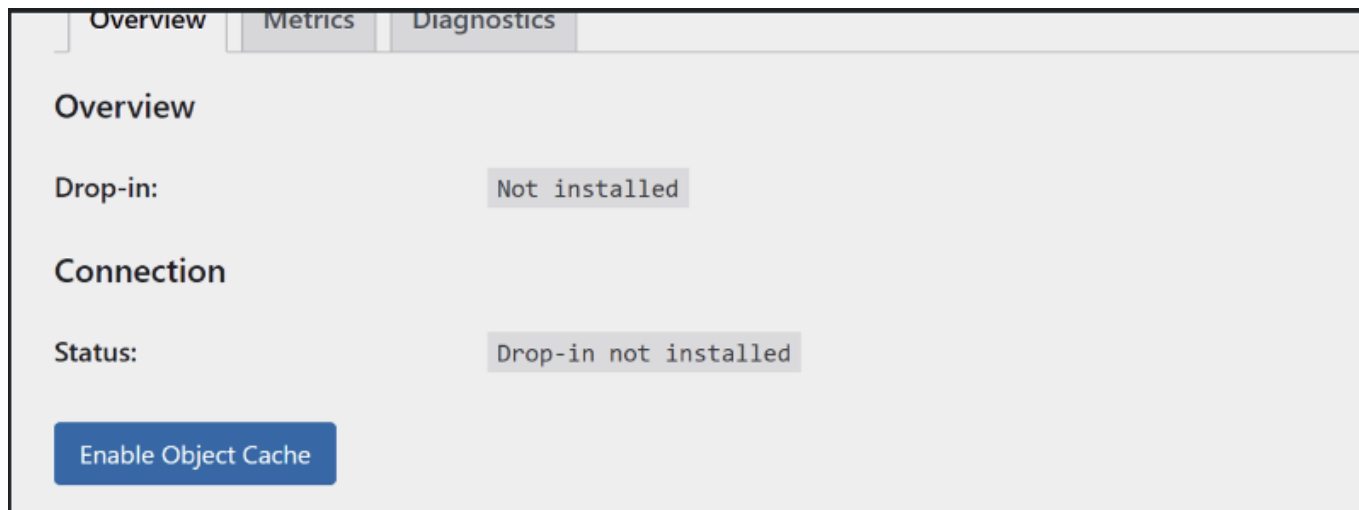
100,000+ Active Installations

Last Updated: 3 months ago

 **Compatible** with your version of WordPress

Installera och aktivera det här plugin-programmet. Gå sedan över till Redis-inställningar (på instrumentpanelsmenyn). Nu bör du aktivera Objektcache.

WebbMotell - ISPConfig



Detta ställer in Redis med standardinställningarna. Det går att ändra dessa inställningar, se [Configuring the plugin](#). Vi rekommendera starkt att ni sätter **WP_REDIS_PREFIX** till något eget.

WebbMotell - ISPConfig

The screenshot shows the WordPress dashboard with the 'Settings' menu item highlighted. The 'Redis Object Cache' settings page is open, displaying the 'Overview' tab. The page shows the client as 'Predis (v1.1.6)', the drop-in as 'Valid', and the connection status as 'Connected'. The host is '127.0.0.1' and the port is '6379'. The database is '0', the connection timeout is '1s', the read timeout is '1s', and the Redis version is '5.0.3'. At the bottom, there are two buttons: 'Flush Cache' and 'Disable Object Cache'.

Redis Object Cache

Overview Metrics Diagnostics

Overview

Client: Predis (v1.1.6)

Drop-in: Valid

Connection

Status: Connected

Host: 127.0.0.1

Port: 6379

Database: 0

Connection Timeout: 1s

Read Timeout: 1s

Redis Version: 5.0.3

Flush Cache Disable Object Cache

Kontrollera om Redis fungerar

Om du går till din webbplats och öppnar en blogg, låt oss säga standard hello world WordPress-bloggen, kommer resultaten av denna begäran att sparas på Redis.

Uppdatera nu startsidan som du laddade Query Monitor.

The screenshot shows the Query Monitor plugin interface. The 'Overview' tab is selected, displaying performance metrics for the request 'GET https://18.216.117.75/ → 200'. The metrics are as follows:

Page Generation Time	Peak Memory Usage	Database Queries	HTTP API Calls	Object Cache
0.0601 0.2% of 30s limit	4.227 kB 0.8% of 524,288 kB server limit 10.3% of 40,960 kB WordPress limit	0.0003 Total: 3	None	99.2% hit rate (1,181 hits, 10 misses) External object cache in use Opcode cache in use: Zend OPcache

WebbMotell - ISPConfig

Och du kan se att sidan nu laddas med endast 3 databasfrågor som tar 0,0003 millisekunder. Vår totala databasfråгатid är nu betydligt lättare än där den var. Alla dessa frågor sparas nu i minnet, vilket minskar de samtal du måste göra direkt till MySQL-databasen. Detta gör backend snabbare för att returnera förfrågningar.

Slutsats

Om du har en omfattande webbplats som woo-commerce kommer detta att spela en stor roll för att göra din webbplats snabbare. Detta cachelagrar också databasintensiva uppgifter som gör webbplatsen mer motståndskraftig.

Även när du har många användare som gör beställningar drar de information från cachen. Du kommer att använda färre resurser och få en snabbare backend-upplevelse.

Redis cachelagrar bara databasfrågor och objekt. Detta innebär att du fortfarande kan fortsätta använda andra cache-plugins som Cache Enabler och WP Cloudflare för att cachelagra dina sidor och göra din webbplats ännu snabbare.

Unikt lösnings-ID: #1012

Av: : Admin

Senast uppdaterad: 2025-01-11 08:48