YOORshop SAS
RCS 817 466 147 — 254 Rue Vendôme, 69003 Lyon, France — yoorhosting.com
`[dati aziendali da completare]`

Rif. rapporto: YH-DEMO-2026-02
Data di analisi: 30 ottobre 2026

DIAGNOSI E SOLUZIONI CHIAVE

RAPPORTO D'INGEGNERIA APPLICATIVA E DATABASE • LIVRABILE N. 2

Ottimizzazione delle prestazioni di un sito web

Analisi dei picchi CPU al 70 %, contesa della cache del menu e riscrittura SQL di una scansione da 787 504 righe.

CLIENTE DESTINATARIO

Società: SARL Moto Pièces Express (Fittizio)
Negozio online: https://boutique-moto.example (PrestaShop 8.2.3)
Server dedicato: srv-demo.yoorhosting.com (Xeon 16 core, 64 GB RAM)
Ambiente: PHP-FPM pool motodemo_prod (48 worker) • MariaDB

CARICO CPU OSSERVATO
70 % a 1 270 %
45 picchi registrati durante gli import

QUERY LENTE PHP
540 / 976
Bloccate in preg_grep del menu

ACCELERAZIONE SQL RILEVATA
x 4 000
Da 1,60 s a 0,0004 s per chiamata

TEMPO FINALE NUOVI PRODOTTI
0,0004 s
Tramite confronto diretto indicizzato

RICHIESTA DEL CLIENTE

«Riscontrando forti rallentamenti durante l'aggiornamento periodico di listini e scorte tramite moduli d'importazione automatizzati, con saturazioni ricorrenti del processore (picchi fino al 1 270 % di CPU cumulativa su 16 core), il cliente ha incaricato YOORshop di **identificare l'origine esatta dei blocchi** e definire un **piano d'azione di ottimizzazione senza impatto negativo sul catalogo**.»

MECCANISMO COMUNE DI SOVRACCARICO IDENTIFICATO

L'analisi incrociata del registro delle query lente PHP (*PHP slowlog*) e dei timestamp di modifica del database (`date_upd`) ha evidenziato un collo di bottiglia strutturale: **ogni salvataggio di un prodotto svuota simultaneamente due cache applicative pesanti**, immediatamente ricalcolate in ciclo continuo durante i flussi di sincronizzazione:

Correlazione temporale accertata: La modifica di 22 prodotti in un'ora coincide al secondo con l'esplosione delle query lente (20:44, 20:47, 20:49, 20:50 e da 20:52 a 20:56). In totale sono stati registrati 45 picchi di carico, innescati dalle richieste POST di amministrazione eseguite dall'IP operatore `198.51.100.22` .

1 Problema N. 1: Rigenerazione abusiva del menu (Modulo `ps_mainmenu`)

PUNTO D'ANALISI	RISCONTRO TECNICO E SOLUZIONE VALIDATA
Aggancio eccessivo	Il modulo è inutilmente agganciato ad `actionObjectProductAddAfter` , `actionObjectProductUpdateAfter` e `actionObjectProductDeleteAfter` (tabella `psdemo_hook_module`). Questi hook forzano la chiamata a `clearMenuCache()` (ps_mainmenu.php:1007), distruggendo l'intera cache JSON.
Costo di ricostruzione	**540 delle 976 richieste lente** PHP si bloccano in `preg_grep` (ps_mainmenu.php:796). La ricostruzione confronta **9 788 categorie con 26 968 immagini**. Due processi concorrenti ricalcolano lo stesso file in parallelo (conflitto di scrittura alle 21:02:01 e 21:02:05).
Rimedio immediato	**Sganciare `ps_mainmenu` dai tre hook prodotto** dal back-office PrestaShop (*Design* > *Posizioni*). Modifica 100 % priva di rischi: il menu non contiene singoli prodotti (configurazione `MOD_BLOCKTOPMENU_ITEMS = CAT77,CAT71,LNK1,LNK3`). Gli hook di categorie e pagine CMS restano attivi.

2 Problema N. 2: Query non indicizzata del blocco «Nuovi prodotti» (`ps_newproducts`)

Il modulo `ps_newproducts` esegue dopo ogni aggiornamento il metodo `Product::getNewProducts()`. La scrittura originale applica un calcolo diretto sulla colonna indicizzata, rendendo l'indice inutilizzabile da MariaDB:

Query originale (Non sargable) — **1,60 secondi**

```
WHERE DATEDIFF(date_add, NOW()) < 20
```

Conseguenza MariaDB: L'indice `date_add` è disattivato. *Full Table Scan* di **787 504 righe** in `psdemo_category_product`, creazione di tabella temporanea su disco e ordinamento pesante.

✓ Query ottimizzata (Override) — **0,0004 secondi**

```
WHERE date_add > '2026-10-07'
```

Conseguenza MariaDB: Utilizzo immediato dell'indice B-Tree. Scansione limitata agli 8 record richiesti. **Guadagno di velocità misurato: fattore 4 000**.

3 Dimensionamento della capacità del pool PHP-FPM

Per assorbire i flussi simultanei senza rifiuti HTTP 503, il pool `motodemo_prod` è stato dimensionato su misura dai nostri tecnici:

PROCESSI MAX (CHILDREN)
48 worker
Standard condiviso: 15

MEMORIA ALLOCATA
1 024 MB / worker
Assorbe gli import di catalogo

CARICO CRON MEDIO
57 % di 1 core
Regime stabilizzato al minuto

RIPRISTINO DELLE CACHE
Immediato
Fine dei conflitti tra processi

4 Sintesi delle azioni e guadagni ottenuti

#	AZIONE RACCOMANDATA	MODALITÀ OPERATIVA	IMPATTO MISURATO
1	Sganciamento del modulo `ps_mainmenu`	Back-office (Design > Posizioni)	**-540 query lente** Eliminazione della contesa `preg_grep`.
2	Deploy override SQL `getNewProducts`	File in `override/classes/Product.php`	**Tempo diviso per 4 000** Fine della scansione di 787 504 righe.
3	Pulizia delle sonde di monitoraggio	Rimozione di `/root/cpuwatch/`	**Completato** Sonde fermate il 28/10 alle 12:44.

Validazione tecnica di competenza:
Polo Ingegneria e Prestazioni Web — YOORshop SAS

✓ Piano di Ottimizzazione Validato