

Scaling Django to 100K RPM - Lessons from Production

How we took a Django monolith from 500 to 100,000 requests per minute without a rewrite.

Phase 1 - Database (500 -> 5,000 RPM)

Indexes, select_related/prefetch_related, PgBouncer connection pooling.

Phase 2 - Caching (5,000 -> 25,000 RPM)

Redis for hot reads, signal-based invalidation, HTTP caching headers on idempotent endpoints.

Phase 3 - Async + edge (25,000 -> 100,000 RPM)

Celery for CPU-bound work, Django Channels for realtime, Nginx upstream pool with health checks.

P95 dropped from 2,300ms to 45ms - zero downtime during the entire programme.