

PERFORMANCE ENGINEERING

Schauen Sie beim nächsten Performance-Einbruch nicht hilflos zu! Lernen Sie jetzt die wichtigsten Begriffe und Methoden kennen und setzen Sie die richtigen Hebel an, um Ihre Performance stabil zu halten.

TYPEN VON PERFORMANCE TESTS



Browser- / Frontend-Tests

messen die Frontend-Leistung aus der Sicht des Nutzers. Konzentrieren Sie sich hierfür auf Metriken wie Time to First Byte, Largest Contentful Paint und Speed Index. Gängige Tools sind beispielsweise Lighthouse, WebPageTest oder Sitespeed.io.



Last-/ Backend-Tests

dienen dazu, die Leistung Ihrer Anwendung in verschiedenen Belastungssituationen zu testen. Der Schwerpunkt liegt auf Metriken wie Anfragen pro Sekunde, Antwortzeiten und Fehlerquoten. Beliebte Tools sind k6, JMeter, Artillery oder XLT.



Hybride Tests

kombinieren Browser- und Lasttests, um die Performance unter realen Bedingungen bewerten zu können. Sie messen sowohl Frontend- als auch Backend-Metriken. Verbreitete Tools hierfür sind k6, Artillery oder XLT.

ARTEN VON LASTTESTS



Klassische Lasttests

werden genutzt, um zu überprüfen, ob Ihre Anwendung Ihren spezifischen Leistungsanforderungen gerecht wird. Dabei wird die Performance unter erwarteter Last über einen kurzen Zeitraum (eine halbe bis mehrere Stunden) getestet.



Spike-Tests

kommen zum Einsatz, wenn die Anwendungsleistung bei plötzlichen, kurzen Traffic-Spitzen (wie z.B. infolge einer größeren Marketingkampagne) überprüft werden soll. Hierbei wird das Zwei- bis Dreifache der durchschnittlichen Belastung simuliert.



Stresstests

dienen zur Überprüfung der Stabilität unter einer länger anhaltenden, extremen Belastung. Die Testzeit beläuft sich auf mehrere Stunden. Währenddessen wird die Belastung schrittweise bis um das Zehnfache des durchschnittlichen Werts erhöht, bis die Belastungsgrenze erreicht ist.



Skalierbarkeitstests

helfen dabei, die Kapazitätsgrenzen Ihrer Anwendung zu finden. Dafür wird die Last schrittweise erhöht, um den Sättigungspunkt zu finden, ab dem die Leistung abnimmt.



Konfigurationstests

dienen dazu, die optimale Konfiguration für Ihre Infrastruktur zu ermitteln. Getestet wird unter der zu erwartenden Last.



Soak-/Endurance-Tests

werden verwendet, um die Anwendungsstabilität über einen längeren Zeitraum zu testen. Getestet wird unter reell erwarteter Belastung über einen Zeitraum von einem halben Tag bis hin zu einer Woche.

HÄUFIGE PERFORMANCE-PROBLEME & DAS RICHTIGE TESTVERFAHREN



Langsame Seitenladezeiten

Verwenden Sie Browser-/Frontend-Tests, um Metriken wie TTFB, Speed Index, LCP, FID usw. zu messen.



Hohe Antwortzeiten unter Last

Verwenden Sie Lasttests, um Sättigungspunkte und Engpässe zu ermitteln. Messen Sie die Antwortzeiten bei unterschiedlichen Anforderungsvolumen.



Häufige Fehler unter Last

Führen Sie Lasttests durch und überwachen Sie die Fehlerraten bei verschiedenen Belastungen. Prüfen Sie, ob die Anwendung richtig skaliert.



Spitzen im Datenverkehr

Nutzen Sie Lasttests, um zu überprüfen, ob die Anwendung Traffic-Spitzen ohne Absturz bewältigen kann. Testen Sie auf das Zwei- bis Dreifache der durchschnittlichen Last.



Layout-Verschiebungen beim Laden der Seite

Verwenden Sie Browser-Tests zur Messung von Cumulative Layout Shift (CLS) und zur Visualisierung von Layout-Änderungen. Optimieren Sie das Seitenlayout.



Langsame Datenbankabfragen

Beobachten Sie ihre Datenbanken unter Last. Ermitteln Sie auf diese Weise, welche Anfragen es konkret sind, die die Performanceminderung verursachen.

DIE WICHTIGSTEN FRONTEND-METRIKEN

Speed Index

Gewichtete Metrik, die das Gesamtladeerlebnis bewertet

Largest Contentful Paint (LCP)

Zeit bis zum Rendern des größten Inhalts im Viewport

First Input Delay (FID)

Zeit, bis der Browser auf eine Eingabe reagiert

Cumulative Layout Shift (CLS)

Stabilität des Layouts

Time to First Byte (TTFB)

Zeit, bis der Server antwortet

TIPPS ZUR VERBESSERUNG DER FRONTEND-LEISTUNG

- Verwenden Sie Code-Splitting zum verzögerten Laden nicht kritischer Assets.
- Minimieren Sie Ressourcen, die das Rendern blockieren.
- Komprimieren Sie Bilder und aktivieren Sie Caching.
- Verwenden Sie ein Content Delivery Network (CDN) zur Verteilung statischer Assets.

DIE WICHTIGSTEN BACKEND-METRIKEN

Anfragen pro Sekunde

Anzahl der gleichzeitig möglichen Anfragen pro Sekunde, bevor es zu Performance-Einbußen oder Crashes kommt

Antwortzeit

Zeit für die Verarbeitung einzelner Anfragen vom Server

Fehlerrate

Häufigkeit der fehlgeschlagenen Anfragen

TIPPS ZUR VERBESSERUNG DER BACKEND-LEISTUNG

- Identifizieren und optimieren Sie langsame Datenbankabfragen.
- Implementieren Sie Caching, um doppelte Arbeit zu vermeiden.
- Verwenden Sie Load Balancer, um Anfragen zu verteilen.
- Sorgen Sie für horizontale Skalierung durch das Hinzufügen von Servern.
- Analysieren Sie den Code zur Identifizierung von Engpässen.



Ein Einbruch Ihrer Performance ist für Sie ein No-Go?

Unsere Performance-Engineering-Profis steht Ihnen gern zur Verfügung. Sichern Sie sich jetzt ein kostenloses 30-minütiges Beratungsgespräch.

Wir freuen uns auf Ihre Anfrage per Mail an: performance@dotsource.de



HIER UNSERE BEST PRACTICES »CLOUD-SERVICES«
KOSTENFREI HERUNTERLADEN

WEB www.dotsource.de | E-MAIL info@dotsource.de

[in](https://www.linkedin.com/company/dotsource) @dotsource SE [f](https://www.facebook.com/dotsource) @dotsource [yt](https://www.youtube.com/channel/UC...) @dotsource SE