

Iteration 2 - App-Server

Durch den Wechsel auf MySQL wurde das Hauptziel (50% kürzere Antwortzeiten, keine Fehler mit nicht angemeldetem Benutzer) bereits erreicht. Dadurch dass SQLite nicht produktiv eingesetzt werden soll, war jedoch zu erwarten dass diese Datenbank um einiges langsamer als MySQL ist.

Damit auch das Nebenziel, nämlich die Fehlerrate, weiter verbessert werden kann bzw. die App noch weiter beschleunigt werden kann wird versucht einen anderen App-Server zu verwenden. Nach ersten Tests während der Übung in der FH kamen für mich Thin bzw. Unicorn in Frage. Ich habe mich schließlich für Unicorn entschieden, obwohl ich im Unterricht kaum Geschwindigkeitsunterschiede zwischen den beiden feststellen konnte.

Test-Setup

Die Analyse der bestehenden Applikation wurde mit nachfolgendem Setup durchgeführt.

Ruby (System): 1.8.7-p352

App-Server: Unicorn 4.6.3 (vorher WEBrick 1.3.1)

Datenbank: MySQL 2

Benchmarking-Tool: Autobench 2.1.1

Analyse

Bei den Messungen ohne angemeldeten Benutzer konnte die **Geschwindigkeit** durch die Verwendung von unicorn als App-Server gegenüber der Iteration 1 erneut um **mindestens 50%** auf allen Seiten verbessert werden.

Bei den Tests mit angemeldetem Benutzer sind die Ergebnisse unterschiedlich. So konnte die **Fehlerrate** auf manchen Seiten (users/45, recipes/XXXX) **verbessert** werden und es traten zumindest keine Totalausfälle mehr auf. Auf den anderen Seiten blieb die Fehlerrate jedoch gleich. Ein **Geschwindigkeitszuwachs** von mindestens **25%** auf allen Seiten konnte jedoch auch hier festgestellt werden. Vor allem bei erhöhten Zugriffszahlen schneidet unicorn hier deutlich besser ab.

Verbesserungen zur Iteration 1 zusammengefasst:

- Erneuter Geschwindigkeitszuwachs sowohl ohne als auch mit angemeldetem Benutzer
- Geringe Verbesserung der Fehlerrate