EagerDB:
A Predictive Cache Warming Tool
6.885 Final Paper

John J. Wang
wangjohn@mit.edu

December 10, 2013

Abstract

Many applications increase database performance using some form of caching. The majority of cache implementations, however, only increase performance for queries that have been already seen. EagerDB takes this a step farther and provides a predictive cache. It predicts queries that will occur in the future with high probability and automatically loads them into the cache. EagerDB allows the user to manually specify query preloads or automatically parse historical database logs, estimate a Markov model, and preload queries which have high transition probabilities.

1 Introduction

Most web applications improve database performance by caching – storing data has has previously been used. Using a cache can dramatically improve database performance when temporal locality is present (i.e. when previously requested data is requested multiple times). Caching has become one of the most fundamental concepts in computer science, and in turn, almost all large technology companies use caches in some form.

However, caches assume that historical data repeats. This assumption does not capture all of the potential performance gains that are possible. In fact, there is quite a large amount of data that exists which can be used to improve performance even further. Since there workflows of users tend to be methodical, the queries that get sent to a database can be predicted. One can then estimate the probability of some query X happening given the current state of the database, and "preload" X when it has a high probability of happening.

EagerDB provides a framework for preloading such queries. Developers can manually specify queries to preload or they can use a prediction engine for discovering queries to preload. EagerDB is built as a Ruby gem and can be easily included in a Ruby on Rails web application for easy improvements in database performance. This paper shall outline the design of EagerDB, the reasons for making particular design decisions, and the results of benchmark tests.

2 Background

There have been a number of papers which have had similar ideas.