Individual Research Paper

CS 480 – Senior Design
Group 3 – Project LightSpeed
Bryan Turek

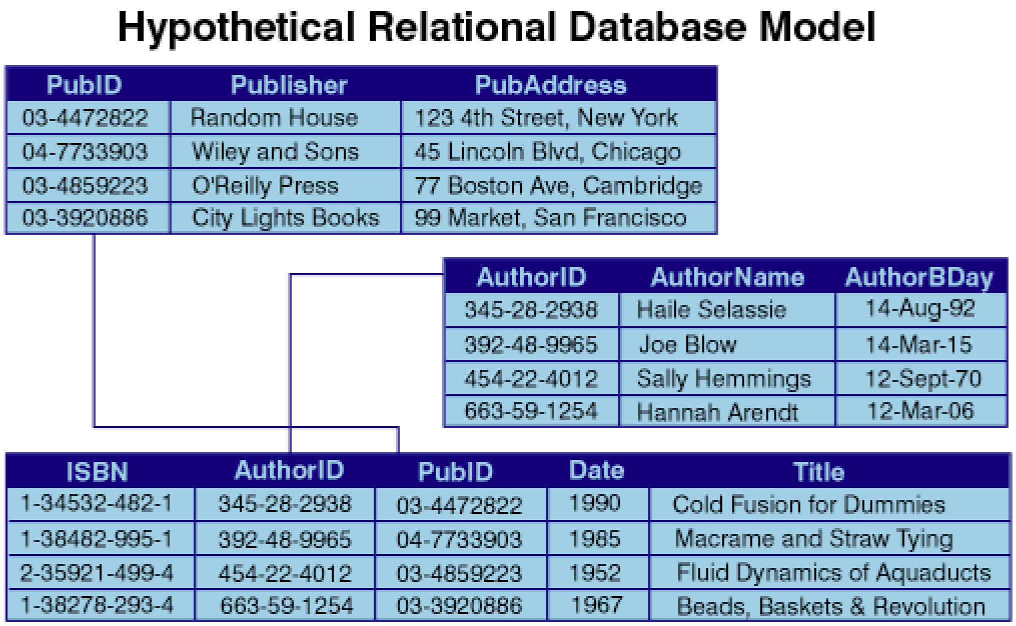
Academic Advisor: Ray Morehead
Professor & Special Assistant to the Chair, Lane Department of Computer Science and Electrical Engineering
Instructor: Yenumula V. Reddy

Table of Contents

Background

Before the 1950's the process of storing data for a company was extremely inefficient. Because data was stored on paper records; a lot of man power and time was wasted. The first efficient database-like systems came out around 1960. These systems utilized the hard disks for direct access, the data was stored in files, and the files were stored in a File Processing System. These systems were called File based Systems. But because these systems encouraged separation and isolation of data; specific data needed to be embedded in these applications and they had no control over access and manipulation of data.

The result of these drawbacks was to create a Database Management System. The first of these DBMS's were hierarchal and network. In the 1970's, relational DBMS's arose, introduced by E. F. Codd (History of Database Systems). These relational databases provide conceptually simple model with all data visible. The first database that incorporated relational models was the DB2 by IBM. The relational model uses "tables" to organize the information together. These tables are structured in rows and columns where rows are entries into the database and columns are attributes of the entry. Codd's successful idea revolutionized the way data was stored and standardized the database industry by providing a way to store and retrieve data based off a simple concept.



Hypothetical Relational Database Model

PubID	Publisher	PubAddress
03-4472822	Random House	123 4th Street, New York
04-7733903	Wiley and Sons	45 Lincoln Blvd, Chicago
03-4859223	O'Reilly Press	77 Boston Ave, Cambridge
03-3920886	City Lights Books	99 Market, San Francisco

AuthorID	AuthorName	AuthorBDay
345-28-2938	Haile Selassie	14-Aug-92
392-48-9965	Joe Blow	14-Mar-15
454-22-4012	Sally Hemmings	12-Sept-70
663-59-1254	Hannah Arendt	12-Mar-06

ISBN	AuthorID	PubID	Date	Title
1-34532-482-1	345-28-2938	03-4472822	1990	Cold Fusion for Dummies
1-38482-995-1	392-48-9965	04-7733903	1985	Macrame and Straw Tying
2-35921-499-4	454-22-4012	03-4859223	1952	Fluid Dynamics of Aquaducts
1-38278-293-4	663-59-1254	03-3920886	1967	Beads, Baskets & Revolution

Figure: Relational Database Model

Key-value databases are a derivation of relational model databases, but behave almost nothing alike. This is because key-value is based off a schema-less model and don't actually have “tables”. This allows data to be free-form and not maintain any sort of structure. A simple way to understand these types of databases is that a value is stored in the database; and to retrieve this value you must give the database the “key”, or the unique name where that value is stored. There are many successful database systems that incorporate this type of structure they are: MongoDB, Cassandra, and Spark just to name a few.

Overview

To solve a persistent problem in big data; Project LightSpeed is an alternative database system that aims to compete with similar database systems but compiling data much faster than any commercial system out there. LightSpeed is a reliable big database system that is based off a key-value database system called Lightning Memory-mapped Database. LMDB, which serves as a base system for LightSpeed, will be wrapped with a concurrent actor system with Scala and have an intuitive user interface that is both user-friendly and powerful. Amazon EC2 instances will allow the database system to use concurrency over multiple CPU's allowing for even faster compiling speeds. With the use of concurrency over multiple instances the LightSpeed database should compile data of up to 100x that of industry standard databases.

LMDB

Lightning Memory-mapped Database is a new database library. It utilizes a highly optimized binary tree system that is much faster than any other binary tree software implementation already out. LMDB is a Key/Value stored database that is crash proof and tiny (SYMAS). The entire system can fit into modern CPU L1 caches (The Lightning Memory-mapped Database). Because of these attributes and the fact that it is highly stable and secure is the reason for why LightSpeed will use this database structure as a basis.

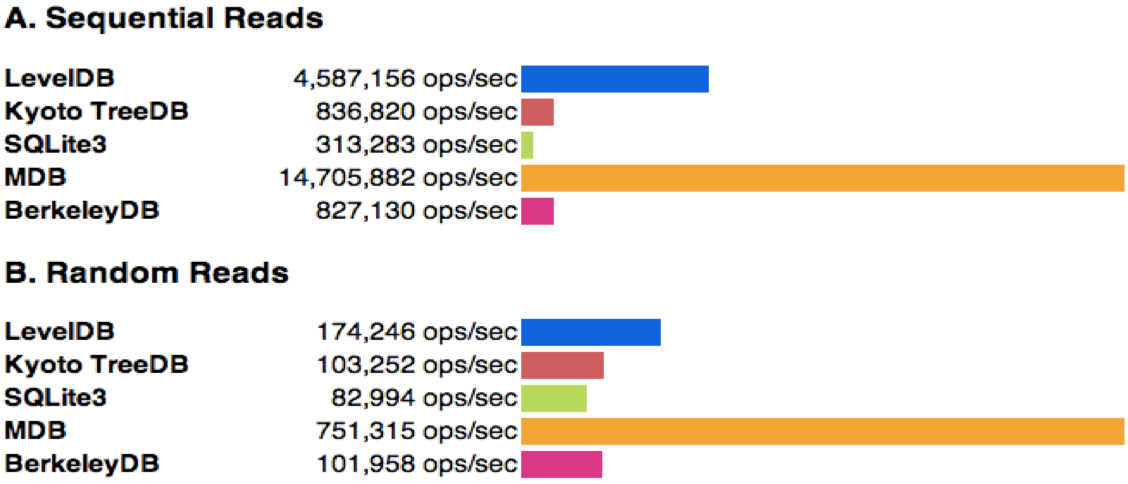


Figure: LMDB Benchmarks

Scala

Scala, or “Scalable Language”, is a new functional and object oriented programming language (SCALA). Because Scala handles concurrency and the fact that it runs on the Java Virtual Machine is why Project LightSpeed will write a Scala “wrapper” for the LMDB. A wrapper is just code that “wraps” another software enabling additional features for the software. We will use Scala to bring concurrency to the LMDB, as well as a Web Application to allow for user-friendly interaction. Scala also brings a scalability part to the project as well. Because our database will run on many CPU's Scala will allow for scalability of software on multiple CPU's allowing for better process management.

Amazon EC2 & S3

The last component of Project LightSpeed will be the actual hardware behind the code. Because we have the latest and greatest software solutions, it only makes sense to have the latest and greatest hardware to compliment it. Amazon EC2 instances are some of the most reliable well functioning server instances you can buy. This will allow us to manage CPU instances better and more efficient, which in turn allows us to test and develop code more effectively. Another part of Amazon Web Services are their S3 solutions. S3, which stands for Simple Storage Service, will allow us to host a large collection of data sets that we can test and compare our product to other databse solutions already out.

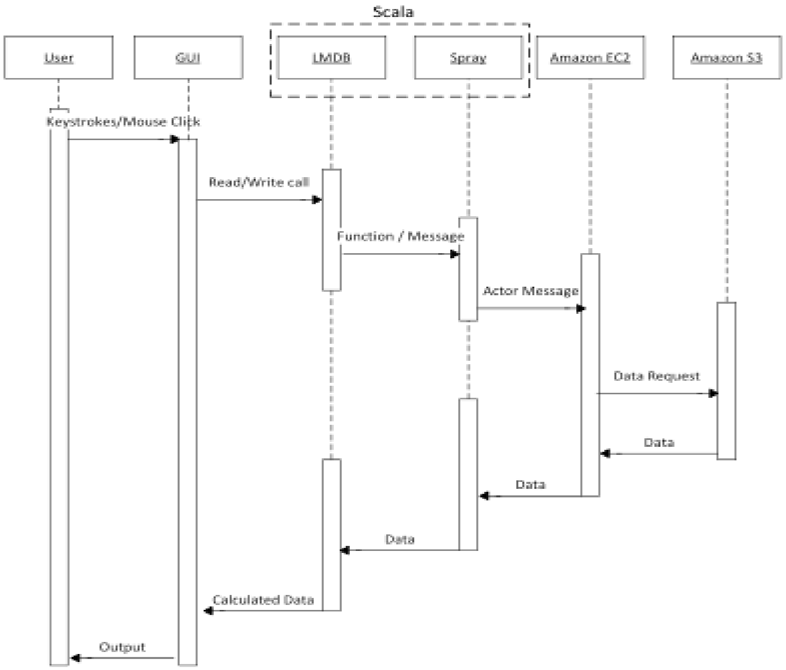


Figure: Sequence Diagram

Objectives

There are three main objectives Project LightSpeed aims to accomplish. First off we will aim to have a reliable database solution that it always secure and maintains data no matter what happens. Next we want a user-friendly user interface that isn't complicated for database management. And last, and most importantly, we are looking to create a database that queries big data faster than anything already out. These objectives will be our “end goal” for the project. We chose these objectives over other goals because we felt that most of these goals are capable of being completed with the team that we have now. Lightning Memory-mapped Database is reliable and fast naturally, to make it even faster we will use concurrency to process data of speeds up to 100x faster than some of the industry standard software solutions. And because we will be using a Web application as a front-end for the database; it will help ensure that our database is both quick and user-friendly.

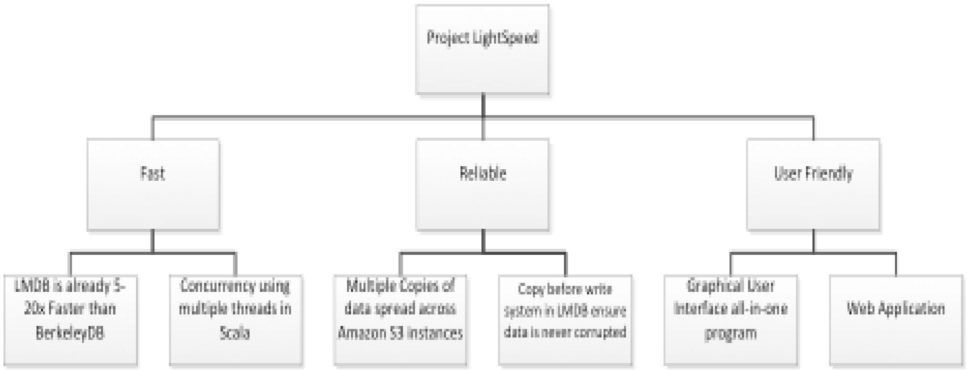

Project LightSpeed
Fast
Reliable
User Friendly
LMDB is already 5-20x Faster than BerkeleyDB
Concurrency using multiple threads in Scala
Multiple Copies of data spread across Amazon S3 instances
Copy before write system in LMDB ensure data is never corrupted
Graphical User Interface all-in-one program
Web Application

Figure: Objective Tree

Ranking of Objectives

1. Reliable

First and foremost databases need to be reliable. If we built a database that was not able to store and maintain data, there was no point in building that database. LMDB is naturally reliable by the nature of the optimized binary tree it is based off of. Because of this there will be multiple copies of data spread across many Amazon S3 instances. We are also naturally protected from over-writing data that was just written from another source, it is protected from concurrent timing issues, therefore data is never corrupted.

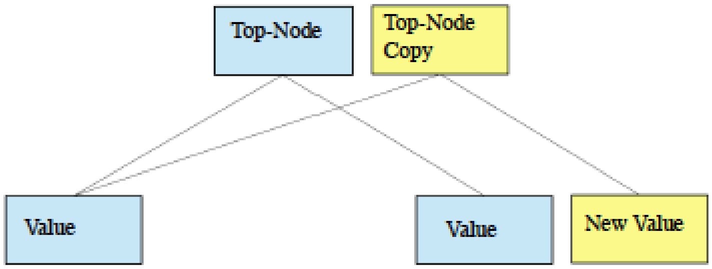


Figure 1.5 - A Lightening MMDB Data Write

2. Fast

Next, because we have large amounts of data everywhere, databases need to process and compile data very quickly. Again, with LMDB, its optimized binary trees accomplish this by providing a lightweight solution that can easily be scaled with Scala and Scala's concurrent nature. LMDB without concurrency has speeds of up to 5-20x that of its predecessor: Berkely DB. With Scala and concurrency speeds are estimated to be increased to nearly 100x that of the industry leading databases.

3. User-Friendly

Many databases today are very complicated and require extensive knowledge about the database to be able to manage it. With LightSpeed we aim to get rid of this need by providing a web interface that easy manages queries, connections, and statistics. By providing this interface it allows developers and database managers to easily view data and provide ways to access the data more effectively.

LightSpeed removes the need to understand complex query languages. With the interface you can highlight, drag and drop, and simply click to manage the data and ways to access the data. Simple interfaces to manage data have been constructed before, however due to the complex nature of existing database solutions, features and other aspects made the interfaces difficult to understand. LightSpeed solves this by taking a solve by need approach, and minimalistic design.

Stakeholders

Today, data is key towards success. Because of this many companies have large data sets that need to be maintained. The problem with large data sets or “big data” is that to query this data takes a lot of time and sometimes data can be lost. LightSpeed has a huge selection of stakeholders as most of the world's companies require large databases for their company. Companies like Facebook, Google, Amazon, and Yahoo would be key stakeholders for Project LightSpeed. These companies have large databases already setup that require complex and difficult systems. With LightSpeed these companies can save money and time by using a system faster than what they are already using.

Existing Solutions

There are already many big data Database Management System solutions. Project LightSpeed, however, has certain attributes that provide a second thought to most people in the DBMS world. Solutions like Oracle, Cassandra, Hadoop, and systems of the like; provide outdated but stable answers for most companies that want to handle large data sets. They handle both relational and key-value type data, and have large communities behind each and every successful one. So why provide another database solution?

LightSpeed uses key-value stored data in conjunction with a concurrent versatile and scalable programming language. The two closest database systems that are similar to LightSpeed are Apache Spark and Meso. Both provide API's to manage data over Amazon Clusters, but don't provide database management solutions meant for companies. Furthermore, LightSpeed will provide an easy to use interface that allows for the average user to understand what is happening and how to manage and retrieve data. The success of these existing products can already be seen. They have been integrated anywhere from small businesses to large corporations including Facebook, Google, and Yahoo. If done correctly LightSpeed can set a new standard for Concurrent Cloud Computing database solutions.

Conclusion

Project LightSpeed offers a realistic goal for a senior design project. However because of the complex nature of databases and software, it provides an intellectually challenging problem. This project really incorporates a lot of the studies from computer science and computer engineering to allow for a stimulating and fun exercise that can be researched and studied to greatly improve later. But most importantly this provides a product that can be used and potentially sold. Along with it having monetary potential the standards and services that can be provided through centering this product within a company allows for greater investments and possible jobs for employees.

References

Chu, H. (2013, March 29). The Lightning Memory-Mapped Database. Retrieved October 12, 2013, from Parleys: http://cdn.parleys.com/p/517f58f9e4b0c6dcd95464ae/517b8a4ecd54a_1367050795078.pdf

Kaushalya Dharmarathna. History of Database Systems http://courses.dbnet.ntua.gr/fsr/5706/history_dbms_long-3.pdf

"Symas Lightning Memory-Mapped Database (LMDB)." *Symas Lightning MDB (aka Lightning Database, LMDB)*. N.p., n.d. Web. 01 Dec. 2013. http://symas.com/mdb/

Odersky, Martin. "What Is Scala?" *The Scala Programming Language*. N.p., n.d. Web. 01 Dec. 2013. http://www.scala-lang.org/what-is-scala.html