Understanding the Impact of Databases on the Energy Efficiency of Cloud Applications

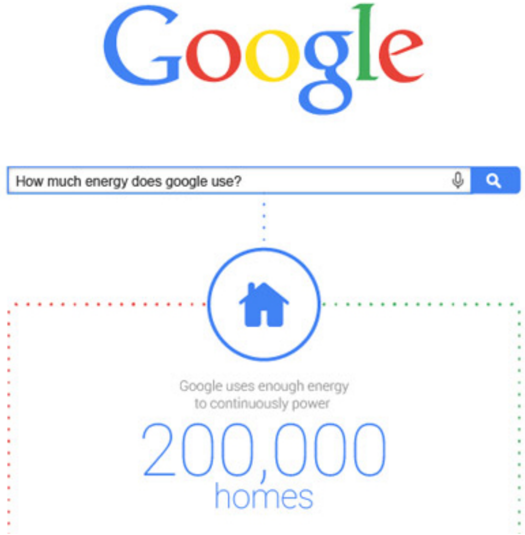
Defense for obtaining the master's degree in applied sciences

Béchir Bani

École Polytechnique de Montréal
Ptidej Team / SWAT Lab

Table of contents

Motivations



Motivations

Motivations

- Little is still known about the energy footprint of these applications and, in particular, of their databases
- Databases are the backbone of cloud-based applications

Motivations

Cloud Application = Databases + Cloud Patterns

Motivations

- None of previous works investigated the combined impact of databases and cloud patterns on the energy consumption of cloud-based applications
- The benefits and trade-offs of different databases and combinations of cloud patterns are mostly intuitive and not validated

Objectives

1. Propose an approach to collect energy measures of cloud-based applications implemented with cloud patterns in conjunction with databases in a cloud environment
2. Evaluate the impact on energy consumption of three cloud patterns: Local Database Proxy, Local Sharding-Based Router, and Priority Message Queue, individually, and also their combination, with three databases: MySQL, PostgreSQL, and MongoDB
3. Highlight the contrast response time with energy efficiency of databases so that developers are aware of the trade-offs between these two quality indicators when selecting a database for their application

Relevant Literature Review

1. Energy Consumption and Applications Design
2. Performance of Relational and NoSQL Databases
3. Impact of Cloud Patterns on Applications Performance

Relevant Literature Review

1. Energy Consumption and Applications Design
2. Performance of Relational and NoSQL Databases
3. Impact of Cloud Patterns on Applications Performance

Energy Consumption and Applications Design

1. **How Green Are Cloud Patterns? (Abtahizadeh et al.)**
   - Compared the energy efficiency of three cloud patterns
   - Showed that cloud patterns can effectively reduce the energy consumption of a cloud application
   - Only considered MySQL database and a RESTful application

2. **Investigating the impacts of web servers on web application energy usage (Manotas et al.)**
   - Investigated the impact of four Web servers on the energy consumption of a Web application
   - Showed that the energy consumption of a Web application depends on the Web server used to handle requests

Energy Consumption and Applications Design

1. **Initial explorations on design pattern energy usage (Sahin et al.)**
   - Investigated the energy efficiency of design patterns
   - Showed that design patterns have a significant impact on energy consumption

2. **How do code refactorings affect energy usage? (Sahin et al.)**
   - Showed that code refactorings affect the energy consumption of applications

Relevant Literature Review

1. Energy Consumption and Applications Design
2. Performance of Relational and NoSQL Databases
3. Impact of Cloud Patterns on Applications Performance

Performance of Relational and NoSQL Databases

1. **Comparison of NoSQL and SQL Databases in the Cloud (Hammes et al.)**
   - Highlighted the performance of both PostgreSQL database and MongoDB database
   - Observed that PostgreSQL databases perform better than MongoDB databases in cloud environments

2. **A comprehensive comparison of SQL and MongoDB Databases (Aghi et al.)**
   - Highlighted the performance of MySQL and MongoDB Databases
   - Showed that MongoDB database performs better than MySQL for complex queries
   - Showed that MySQL databases perform better than MongoDB databases for small datasets

Relevant Literature Review

Impact of Cloud Patterns on Applications Performance

1. **An empirical Study of the impact of cloud patterns on Quality of Service (QoS) (Hecht et al.)**
   - Studied the impact of three cloud patterns on QoS
   - Reported that the implementation of the Local Database Proxy pattern can significantly impact the QoS

2. **Scalability patterns for platform-as-a-service (Ardagna et al.)**
   - Evaluated the impact of five scalability patterns on the performance of a Platform as a Service (PaaS)
   - Showed that each pattern can affect the way virtual machine resources are added and removed

Methodology

1. Research Questions and Hypothesis
2. Objects and Design
3. Research variables
   - Independent Variables
   - Dependent Variables
4. Data Extraction Process
5. Analysis Method

Methodology

Research Questions and Hypothesis

RQ1: Does the choice of MySQL, PostgreSQL, and MongoDB databases affect the energy consumption of cloud applications (when no cloud patterns are implemented)?

- H^1_{0yz}: There is no difference between the average amount of energy consumed by applications implementing databases D_y and D_z (without any cloud pattern)
- H^2_{0yz}: There is no difference between the average response time of databases D_y and D_z (without any cloud pattern)

Research Questions and Hypothesis

RQ2: Does the implementation of Local Database Proxy, Local Sharding Based Router, and Priority Message Queue patterns affect the energy consumption of cloud applications using MySQL, PostgreSQL, and MongoDB Databases?

- H^1_{xyz}: There is no difference between the average amount of energy consumed by applications implementing databases D_y and D_z in conjunction with patterns Px
- H^2_{xyz}: There is no difference between the average response time of databases D_y and D_z by applying the design Px

Research Questions and Hypothesis

RQ3: Do the interactions between Local Database Proxy, Local Sharding Based Router, and Priority Message Queue patterns affect the energy consumption of cloud applications using MySQL, PostgreSQL, and MongoDB databases?

- H^1_{xyz7}: There is no difference between the average amount of energy consumed by applications implementing databases D_y and D_z in conjunction with the combination of patterns Px and P7
- H^2_{xyz7}: There is no difference between the average response time of databases D_y and D_z by applying the combination of designs Px and P7

Methodology

Objects and Design

RESTful multi-threaded application

- Communicates through REST calls
- Use *Sakila sample database* (provided by MySQL)
- Adapt the schema of *Sakila sample database* to PostgreSQL and MongoDB databases
- Implemented with different patterns and strategies
- Clients are simulated using a multi-threaded architecture (100; 250; 500; 1,000; 1,500)

Objects and Design

DVD Store application

- Standard cloud-based application
- Open source simulation of an e-commerce web site
- We refactor the code of DVD Store to allow it to connect to a MongoDB database
- Clients are simulated using a multi-threaded architecture (100, 250; 500; 1,000; 1,500)

Objects and Design

JPetStore application

- Standard cloud-based application
- Open source simulation of an e-commerce web application
- We refactor the code of JPetStore to allow it to connect to a PostgreSQL and MongoDB databases
- Clients are simulated using a multi-threaded architecture (100; 250; 500; 1,000; 1,500)

Objects and Design

Power-API

- Provides power information (in watts converted to joules to measure the energy) per PID for each system component (CPU, memory, etc.)
- Uses sensors and analytical models for its energy estimation
- Allows to estimate the amount of power required by the CPU to execute a process (*at the corresponding PID*)
- Does not introduce noise in its measurements

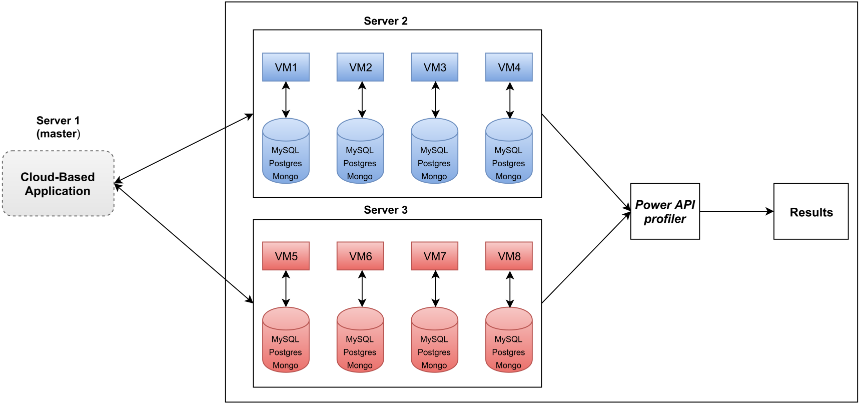
Methodology

Research Variables

1. **Independent Variables**
   - **Databases**: MySQL, PostgreSQL, MongoDB
   - **Cloud Patterns**: Local Database Proxy, Local Sharding-Based Router, Priority Message Queue
2. **Dependent Variables**
   - **Response Time**: Corresponding to Select and Insert requests (milliseconds)
   - **Energy Consumption**: Using Power-API profiler (Joules)

Methodology

Data Extraction Process



Energy Consumption Data Extraction Process

Data Extraction Process

Energy Data Collection Procedure

```
CollectData(VMs, CloudApp, Profiler)
Begin
StartCloudApp()
ExecuteCloudApp(x) // Seconds
    for all VM ∈ VMs do
        StartProfiler()
        ExecuteProfiler(x) // Seconds
        FinishExecProfiler()
    end for
FinishExecCloudApp()
End
```

Methodology

1. Research Questions and Hypothesis
2. Objects
3. Design
4. Research variables
   - Independent Variables
   - Dependent Variables
5. Data Extraction Process
6. Analysis Method

Analysis Method

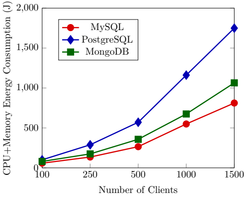
- **Mann-Whitney U test**
  - A non-parametric statistical test whose relevance is reflected in the assessment of two independent distributions
  - The null hypothesis is rejected (there is a significant difference between the the two distributions) when its p-value < 0.05

- **Cliff's δ effect size**
  - Represents the degree of interlock between two sample distributions
  - Its value ranges from -1 to +1: **negligeable** ($\delta < 0.147$), **small** ($0.147 < \delta < 0.33$), **medium** ($0.33 < \delta < 0.474$) and **large** ($\delta > 0.474$)

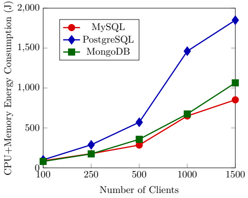
RQ1:

Does the choice of MySQL, PostgreSQL, and MongoDB databases affect the energy consumption of cloud applications (when no cloud patterns are implemented)?

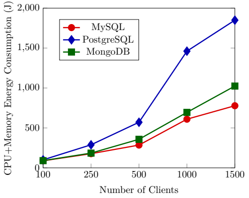
Without Cloud Patterns



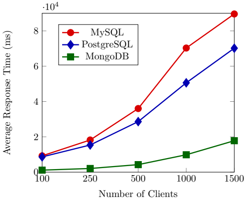
(a) RESTful (Energy)



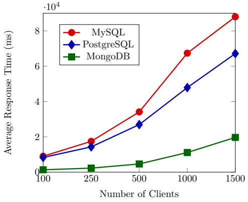
(b) DVDStore (Energy)



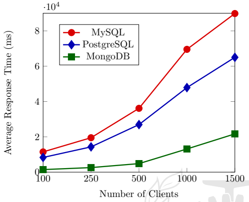
(c) JPetStore (Energy)



(d) RESTful (Time)



(e) DVDStore (Time)



(f) JPetStore (Time)

Without Cloud Patterns

Energy Consumption *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P0	262.5	568.2	0.01	**medium**	262.5	354.7	0.24	**small**	568.2	354.7	0.09	**small**

Response Time *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P0	36018.6	28615.7	0.09	**small**	36018.6	4253.8	$< 10e{-}6$	**large**	28615.7	4253.8	$< 10e{-}6$	**large**

RQ2:

Does the implementation of Local Database Proxy, Local Sharding Based Router, and Priority Message Queue patterns affect the energy consumption of cloud applications using MySQL, PostgreSQL, and MongoDB Databases?

Local Database Proxy Pattern

Energy Consumption *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P1	490.2	1391.1	$< 10e{-6}$	**large**	490.2	890.0	$< 10e{-6}$	**large**	1391.1	890.0	0.09	**small**
P2	495.2	1529.9	$< 10e{-6}$	**large**	495.2	915.9	$< 10e{-6}$	**large**	1529.9	915.9	0.04	**medium**
P3	495.0	1476.5	$< 10e{-6}$	**large**	495.0	904.5	$< 10e{-6}$	**large**	1476.5	904.5	0.04	**medium**

Response Time *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P1	30430.0	27867.8	0.23	**small**	30430.0	3639.8	$< 10e{-6}$	**large**	27867.8	3639.8	$< 10e{-6}$	**large**
P2	29504.1	27036.5	0.23	small	29504.1	3214.2	$< 10e{-6}$	**large**	27036.5	3214.2	$< 10e{-6}$	**large**
P3	29825.2	26129.6	0.23	**small**	29825.2	3275.0	$< 10e{-6}$	**large**	26129.6	3275.0	$< 10e{-6}$	**large**

Local Sharding-Based Router

Energy Consumption p-value and Cliff's δ

Pattern	MySQL	PostgreSQL	p-value	Cliff's δ	MySQL	MongoDB	p-value	Cliff's δ	PostgreSQL	MongoDB	p-value	Cliff's δ
P4	1331.9	6330.2	$< 10e{-6}$	**large**	1331.9	5826.4	$< 10e{-6}$	**large**	6330.2	5826.4	0.23	**small**
P5	611.6	4245.1	$< 10e{-6}$	**large**	611.6	3821.8	$< 10e{-6}$	**large**	4245.1	3821.8	0.23	**small**
P6	824.1	4929.4	$< 10e{-6}$	**large**	824.1	4194.4	$< 10e{-6}$	**large**	4929.4	4194.4	0.23	**small**

Response Time p-value and Cliff's δ

Pattern	MySQL	PostgreSQL	p-value	Cliff's δ	MySQL	MongoDB	p-value	Cliff's δ	PostgreSQL	MongoDB	p-value	Cliff's δ
P4	170693.1	138026.6	0.09	**small**	170693.1	26259.5	$< 10e{-6}$	**large**	138026.6	26259.5	$< 10e{-6}$	**large**
P5	165250.7	145382.6	0.09	**small**	165250.7	27897.8	$< 10e{-6}$	**large**	145382.6	27897.8	$< 10e{-6}$	**large**
P6	168786.5	130585.0	0.09	**small**	168786.5	24680.3	$< 10e{-6}$	**large**	130585.0	24680.3	$< 10e{-6}$	**large**

RQ3:

Do the interactions between Local Database Proxy, Local Sharding Based Router, and Priority Message Queue patterns affect the energy consumption of cloud applications using MySQL, PostgreSQL, and MongoDB databases?

Combination Proxy Pattern and Message Queue Pattern

Energy Consumption *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P1+P7	442.7	1379.8	$< 10e{-6}$	**large**	442.7	814.3	$< 10e{-6}$	**large**	1379.8	814.3	0.03	**medium**
P2+P7	468.8	1482.5	$< 10e{-6}$	**large**	468.8	891.9	$< 10e{-6}$	**large**	1482.5	891.9	0.03	**medium**
P3+P7	490.2	1391.1	$< 10e{-6}$	**large**	490.2	890.0	$< 10e{-6}$	**large**	1391.1	890.0	0.09	**small**

Response Time *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P1+P7	27826.2	22299.8	0.48	**negligible**	27826.2	3747.1	$< 10e{-6}$	**large**	22299.8	3747.1	$< 10e{-6}$	**large**
P2+P7	26703.4	25706.8	0.48	**negligible**	26703.4	3127.5	$< 10e{-6}$	**large**	25706.8	3127.5	$< 10e{-6}$	**large**
P3+P7	29339.7	23153.6	0.23	**small**	29339.7	4210.2	$< 10e{-6}$	**large**	23153.6	4210.2	$< 10e{-6}$	**large**

Combination Sharding and Message Queue Patterns

Energy Consumption *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P4+P7	1255.5	5777.4	$< 10e{-6}$	**large**	1255.5	5622.9	$< 10e{-6}$	**large**	5777.4	5622.9	0.82	**negligible**
P5+P7	492.2	3884.5	$< 10e{-6}$	**large**	492.2	3386.6	$< 10e{-6}$	**large**	3884.5	3386.6	0.23	**small**
P6+P7	775.9	4526.8	$< 10e{-6}$	**large**	775.9	4127.4	$< 10e{-6}$	**large**	4526.8	4127.4	0.23	**small**

Response Time *p*-value and Cliff's δ

Pattern	MySQL	PostgreSQL	*p*-value	Cliff's δ	MySQL	MongoDB	*p*-value	Cliff's δ	PostgreSQL	MongoDB	*p*-value	Cliff's δ
P4+P7	37584.7	29287.7	0.23	**small**	37584.7	2716.3	$< 10e{-6}$	**large**	29287.7	2716.3	$< 10e{-6}$	**large**
P5+P7	38153.7	26445.6	0.09	**small**	38153.7	2869.7	$< 10e{-6}$	**large**	26445.6	2869.7	$< 10e{-6}$	**large**
P6+P7	34183.0	27507.3	0.23	**small**	34183.0	20609.3	0.03	**medium**	27507.3	20609.3	0.09	**small**

Important Results

- **MySQL** database is the least energy consuming but is the slowest among the three databases
- **PostgreSQL** database is the most energy consuming among the three databases, but is faster than **MySQL** but slower than **MongoDB**
- **MongoDB** database consumes more energy than **MySQL** but less than **PostgreSQL** and is the fastest among the three databases

Discussions

- PostgreSQL database generates multiple parallel processes to run the requests sent by the RESTful cloud-based application
- MySQL and MongoDB generate only one process at a time to handle requests sent by the cloud-based application

Discussions

- MySQL and PostgreSQL follow the ACID model
- MongoDB database follow the BASE model

  $\Rightarrow$ MongoDB is faster than the other two relational databases

  $\Rightarrow$ requests processed by relational databases must be executed one by one and cannot be executed in a Simultaneous way

Summary

- We carried on a series of experiments on different versions of three cloud applications
- We contrasted the performance of various combinations of databases and cloud patterns in terms of energy consumption and response time of the cloud-based applications
- Databases can reduce the energy consumption of cloud-based applications
- Cloud patterns do not impact the behavior of the databases

Limitations of the proposed approach

- Energy measurements are subject to perturbations depending of hardware and network
- More studies should be conducted with possibly more accurate tools to verify our findings
- Our findings may still be specific to our studied applications, which were designed specifically for the experiments: future works should replicate this study on other cloud-based applications

Future work

- Expand our study to different NoSQL databases like HBase, Cassandra and HANA
- Investigate the energy impact of data modeling strategies like denormalization and data duplication
- Examine how a match/mismatch between the selected database and the workload characteristic affects energy efficiency

Publication

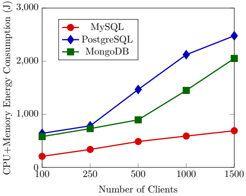
Earlier study in the thesis is published as follows:

- A Study of the Energy Consumption of Databases and Cloud Patterns
  Béchir Bani, Foutse Khomh and Yann-Gaël Guéhéneuc, in *Proceedings of the 14th International Conference On Service Oriented Computing (ICSOC), Banff, Alberta, Canada, 10-13 October, 2016.*

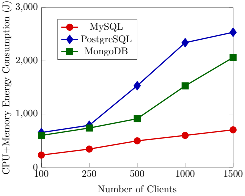
  My contribution: Methodology, analysis and paper writing.

- We carried on a series of experiments on different versions of three cloud applications
- We contrasted the performance of various combinations of databases and cloud patterns in terms of energy consumption and response time of the cloud-based applications
- Databases can reduce the energy consumption of cloud-based applications
- Cloud patterns do not impact the behavior of the databases

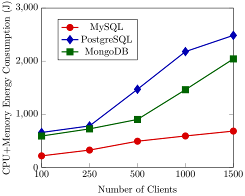
Local Database Proxy Pattern



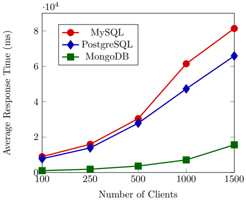
(a) Random (Energy)



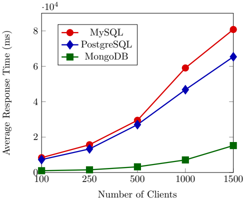
(b) R-R (Energy)



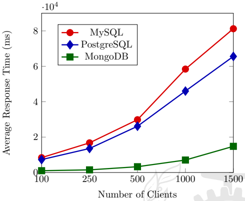
(c) Custom (Energy)



(d) Random (Time)

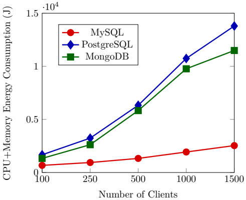


(e) R-R (Time)

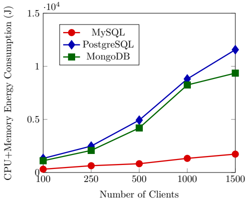


(f) Custom (Time)

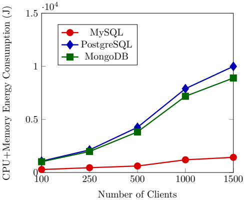
Local Sharding-Based Router Pattern



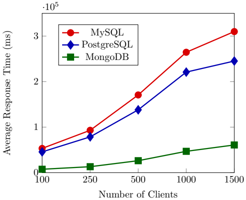
(a) Modulo (Energy)



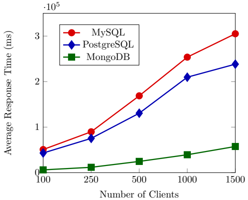
(b) LookUp (Energy)



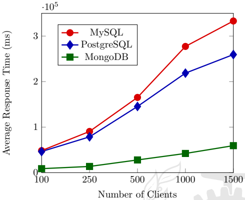
(c) Consistent (Energy)



(d) Modulo (Time)

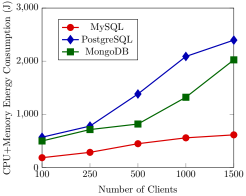


(e) LookUp (Time)

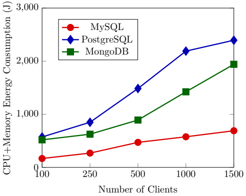


(f) Consistent (Time)

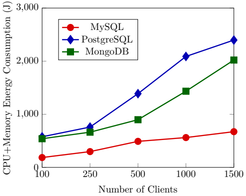
Combination Proxy Pattern and Message Queue Pattern



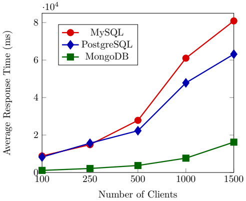
(a) Random* (Energy)



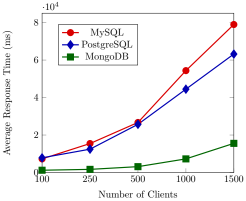
(b) R-R* (Energy)



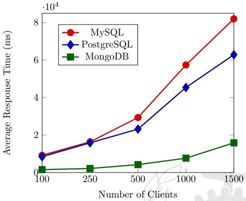
(c) Custom* (Energy)



(d) Random* (Time)

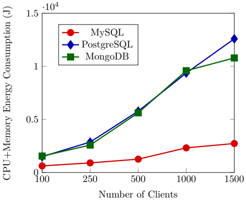


(e) R-R* (Time)

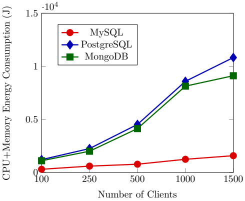


(f) Custom* (Time)

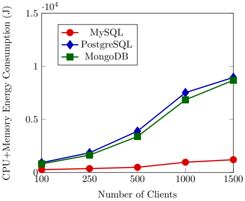
Combination Sharding and Message Queue Patterns



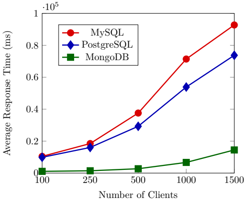
(a) Modulo* (Energy)



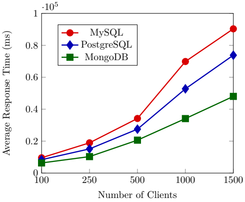
(b) LookUp* (Energy)



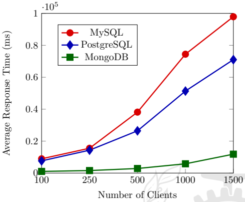
(c) Consistent*(Energy)



(d) Modulo* (Time)



(e) LookUp* (Time)



(f) Consistent*(Time)