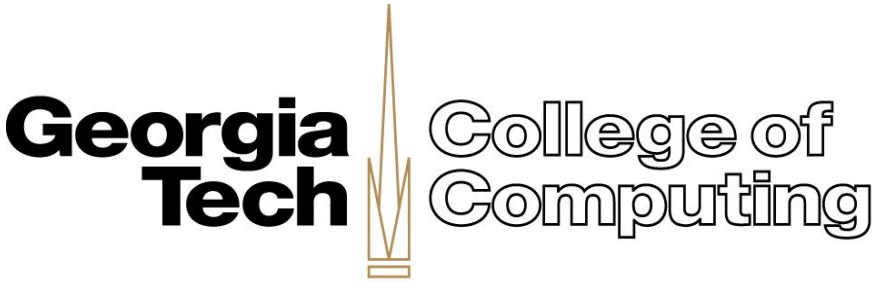
# Variations in Performance and Scalability when Migrating n-Tier Applications to Different Clouds

**Deepal Jayasinghe**, Simon Malkowski, Qingyang Wang, Jack Li, Pengcheng Xiong, Calton Pu

# Outline

- Motivation
- Experiment setup
- Results on Emulab/ Open Cirrus
- Results on EC2
  - Horizontal scalability
  - Vertical Scalability
- Issue 1 : Multi-threading overhead
- Issue 2 : Network driver overhead
- Conclusion

# Motivation

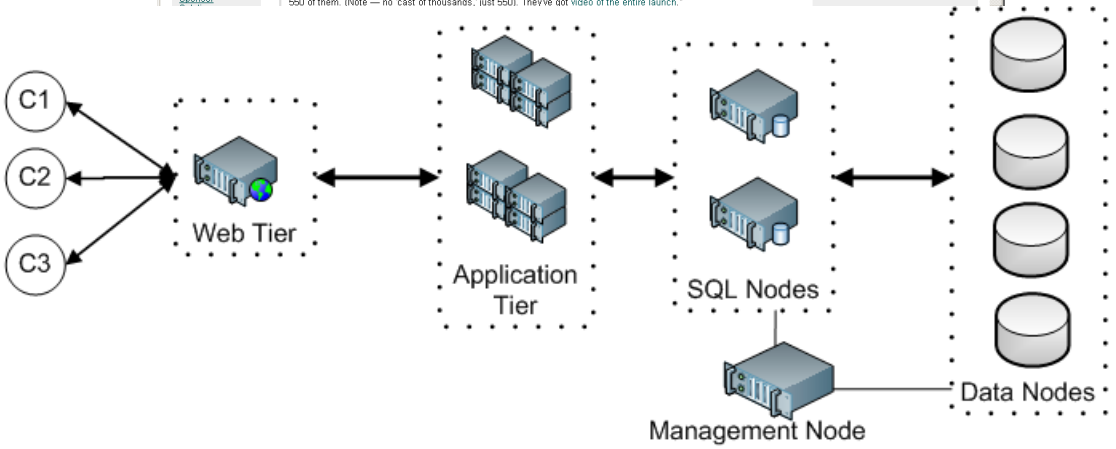
- Clouds are much popular, but not a mature technology.
- More experimental studies are needed to better understand them.
- N-Tier applications are complex and migrating them to clouds is a non-trivial task.

We analyzed the performance and scalability when migrating n-tier applications from a traditional datacenter to an Infrastructure as a Service (IaaS) cloud.

# Experiment Setup

- RUBBoS benchmark
  - Bulletin board system like Slashdot (www.slashdot.org)
  - Typical 3-tier or 4-tier architecture
  - Two types of workload
    - Browsing only
    - Read/Write mix
  - 24 web interactions


- MySQL Cluster
  - Middleware for databa scale-out
    - Multi-master
    - In-memory




RUBBoS Deployment Topology with MySQL Cluster

# Experiment Environment - Emulab

- Emulab (http://www.emulab.net)
  - Relatively modest testbed originally for network research
  - Virtual network & physical machines (not VM)

| Hardware | Specifications |
|---|---|
| Server type | PC3000 in Emulab |
| Processor | Xeon 3GHz 64bit |
| Memory | 2GB |
| Network | 1Gbps |
| Disk | 2 x 146GB 10,000rpm |

# Experiment Environment – Open Cirrus

- Open Cirrus (https://opencirrus.org/)
  - Open cloud-computing research testbed
  - Designed to support research into the design, provisioning, and management of services at a global, multi-datacenter scale.

| Hardware | Specifications |
|---|---|
| Server type | X3210 |
| Processor | 3.00GHz (Quad Core) |
| Memory | 7.5 GB |
| Network | InfiniBand |

# Experiment Environment – Amazon EC2

- Amazon EC2 – (http://aws.amazon.com/ec2/)
  - An Elastic Compute Cloud that provides resizable compute capacity in the cloud.
  - It is designed to make web-scale computing easier for developers.

| Node Type | CPU | Memory | Platform | Price/Hour |
|---|---|---|---|---|
| Small | 1 EC2 Unit[1] | 1.7 GB | 32 bit | $ 0.085 |
| Large | 4 EC2 Unit | 7.5 GB | 64 bit | $ 0.34 |
| Ex-Large | 8 EC2 Unit | 15 GB | 64 bit | $ 0.68 |
| Cluster | 33.5 EC2 Unit | 23 GB | 64 bit | $ 1.60 |

1: EC2 Unit = 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor

# Software Settings

| Function | Software |
| --- | --- |
| Web server | Apache 2.0.54 |
| Application server | Apache Tomcat 5.5.17 |
| DB clustering middleware | MySQL Cluster -5.0.51a |
| Java | Sun jdk1.6.0_14 |
| Operating system | FC4 / FC8 |
| System Monitor | dstat |

# Summary of Experiments

- Automated experiment management approach (Elba[1])

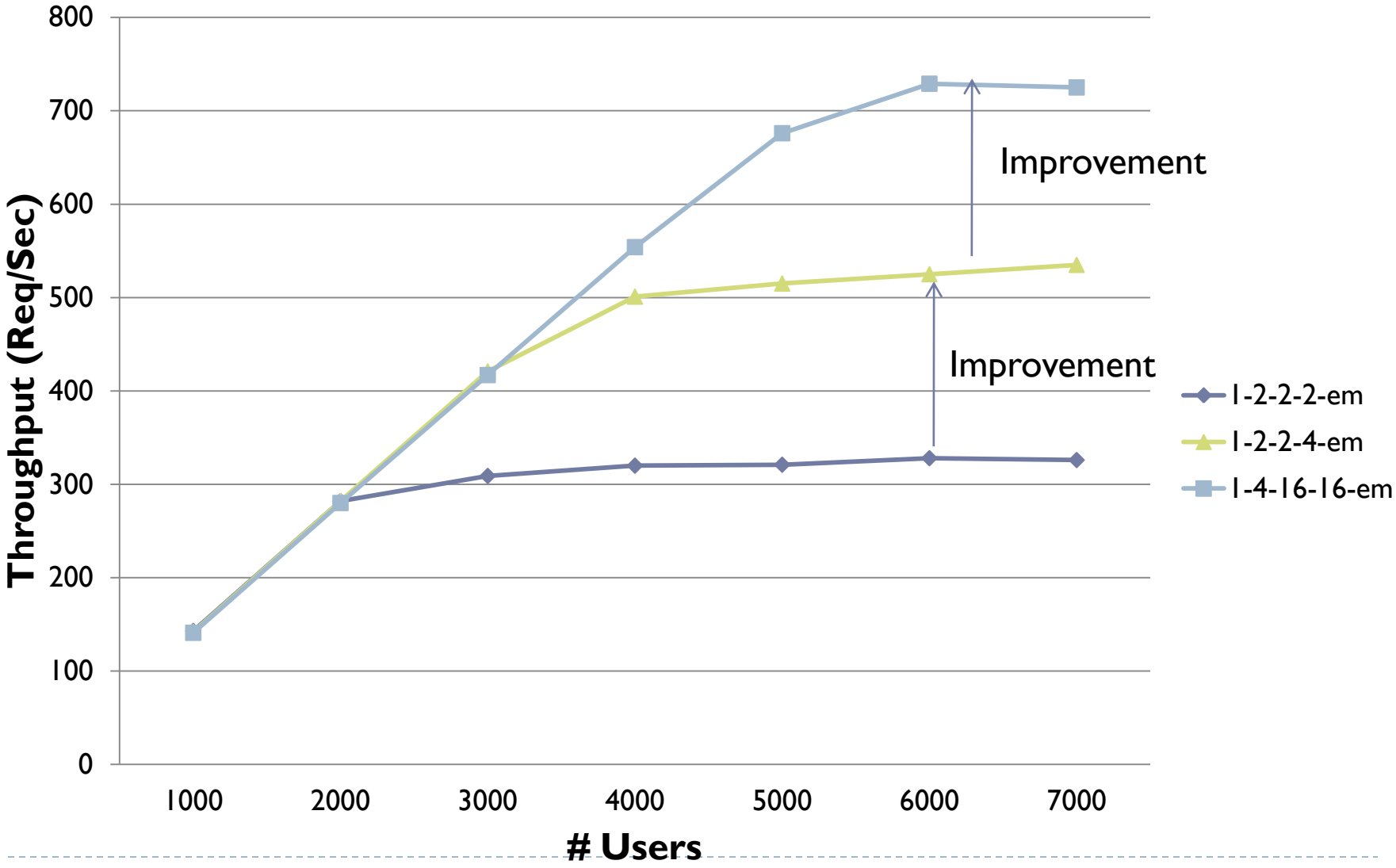
| Type | Emulab | Open Cirrus | Amazon EC2 |
|---|---|---|---|
| Experiments | 8124 | 430 | 1436 |
| Node | 95682 | 4480 | 25846 |
| Configurations | 342 | 23 | 86 |
| Data points | 3210.6 M | 2.3 M | 672 M |

1: http://www.cc.gatech.edu/systems/projects/Elba/

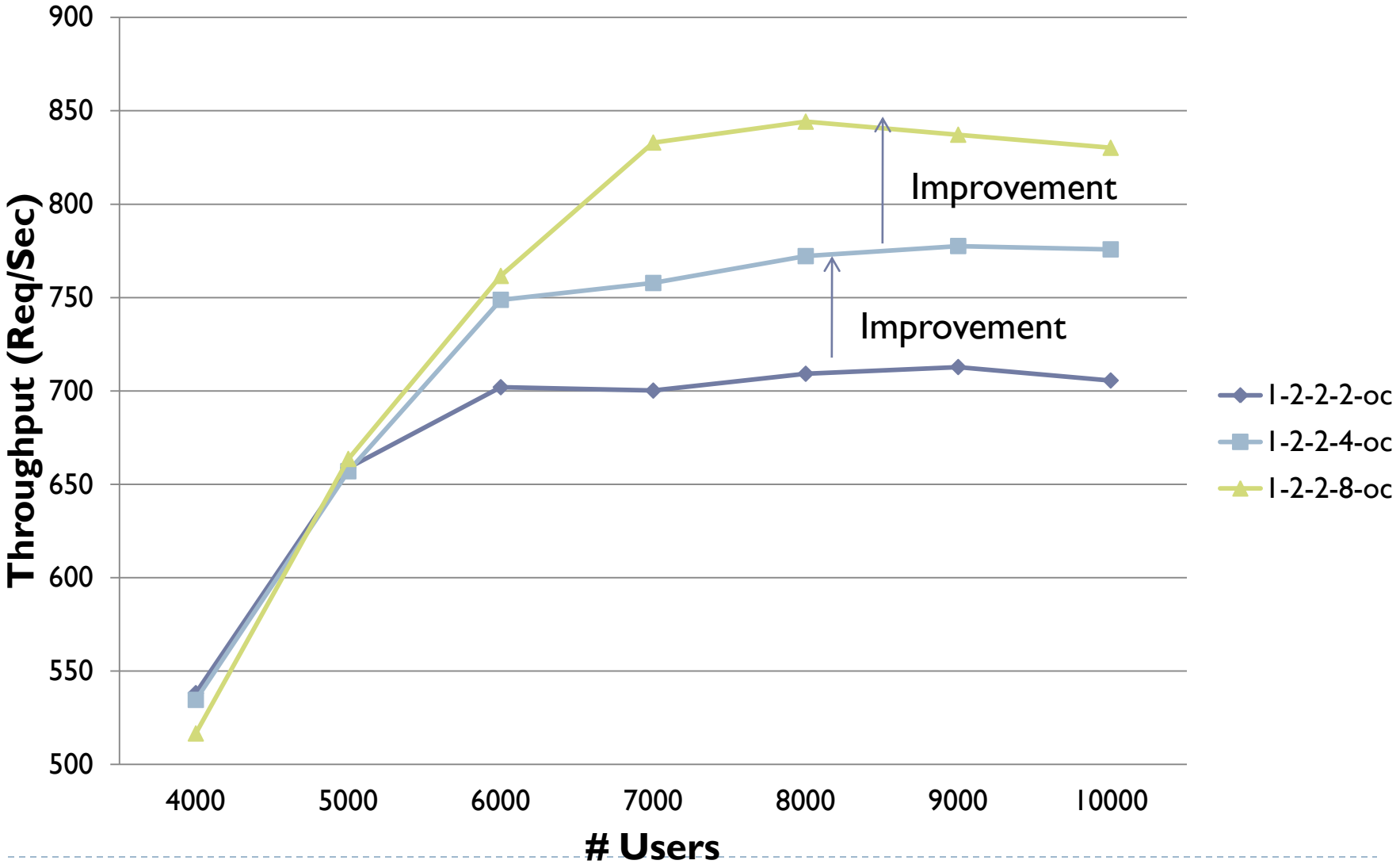
# Notation

- We use the notation #W-#A-#S-#D-p
  - #W – Number of Web server (all experiments in this paper with one Web server)
  - #A – Number of Application servers
  - #S – Number of SQL nodes
  - #D – Number of Data nodes
  - p – Platforms (em- for Emulab, oc – for Open Cirrus, ec2- for EC2)
- 1-2-2-2-em
  - Emulab configuration with 1-Apache, 2-Tomcats, 2-SQL nodes and 2-Data nodes

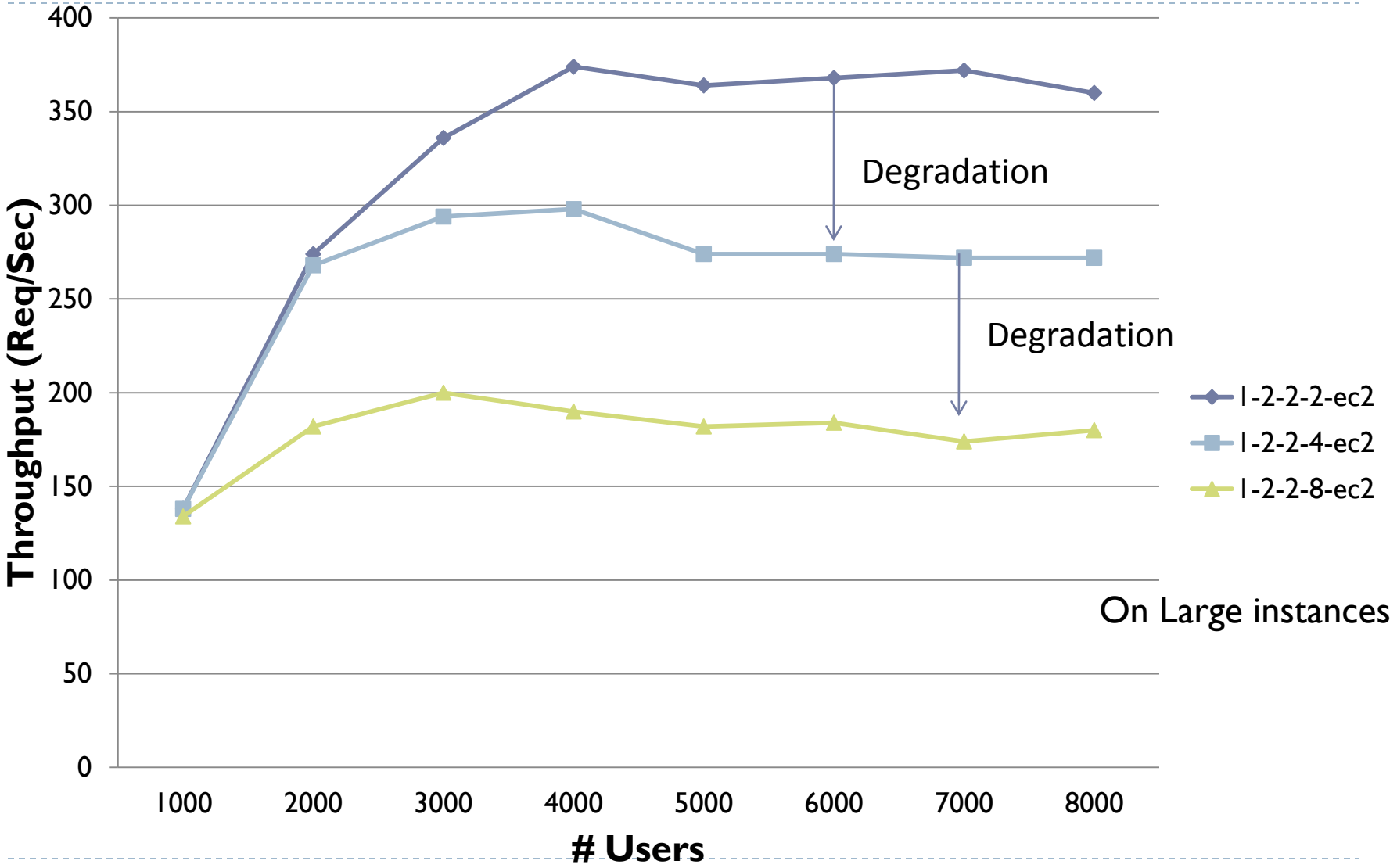
# Performance on Emulab



# Performance on Open Cirrus

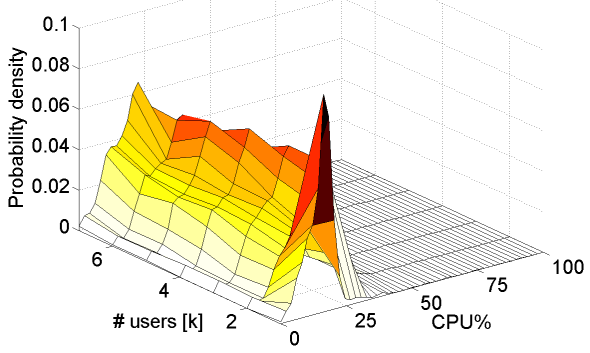


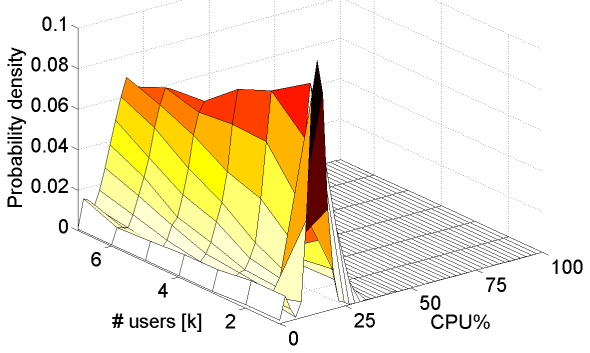
# Performance on EC2

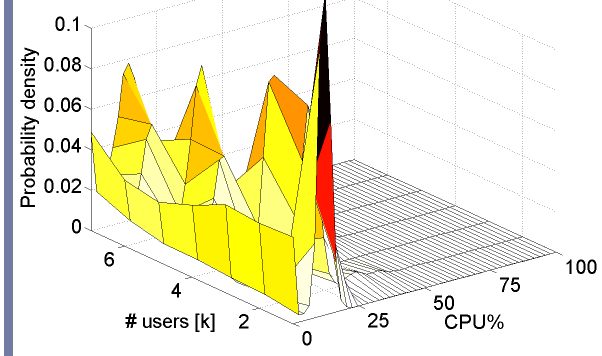


# CPU Utilization (1-2-2-2)

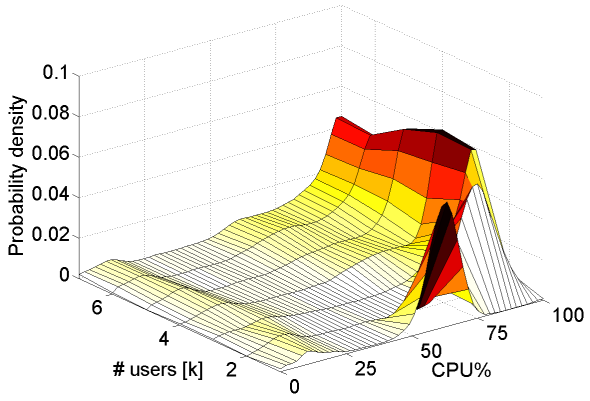
EC2

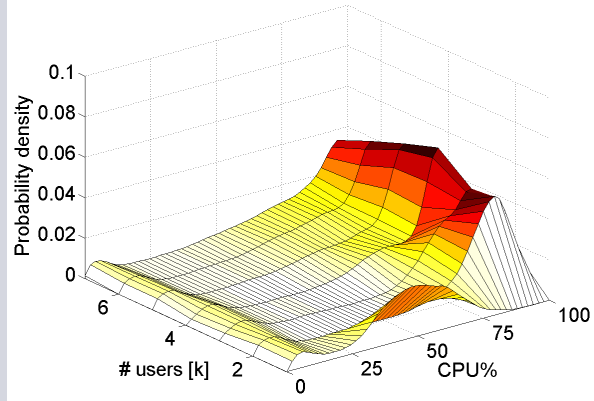


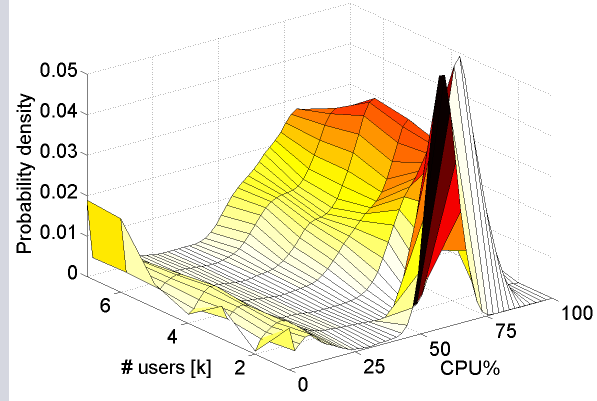




Emulab

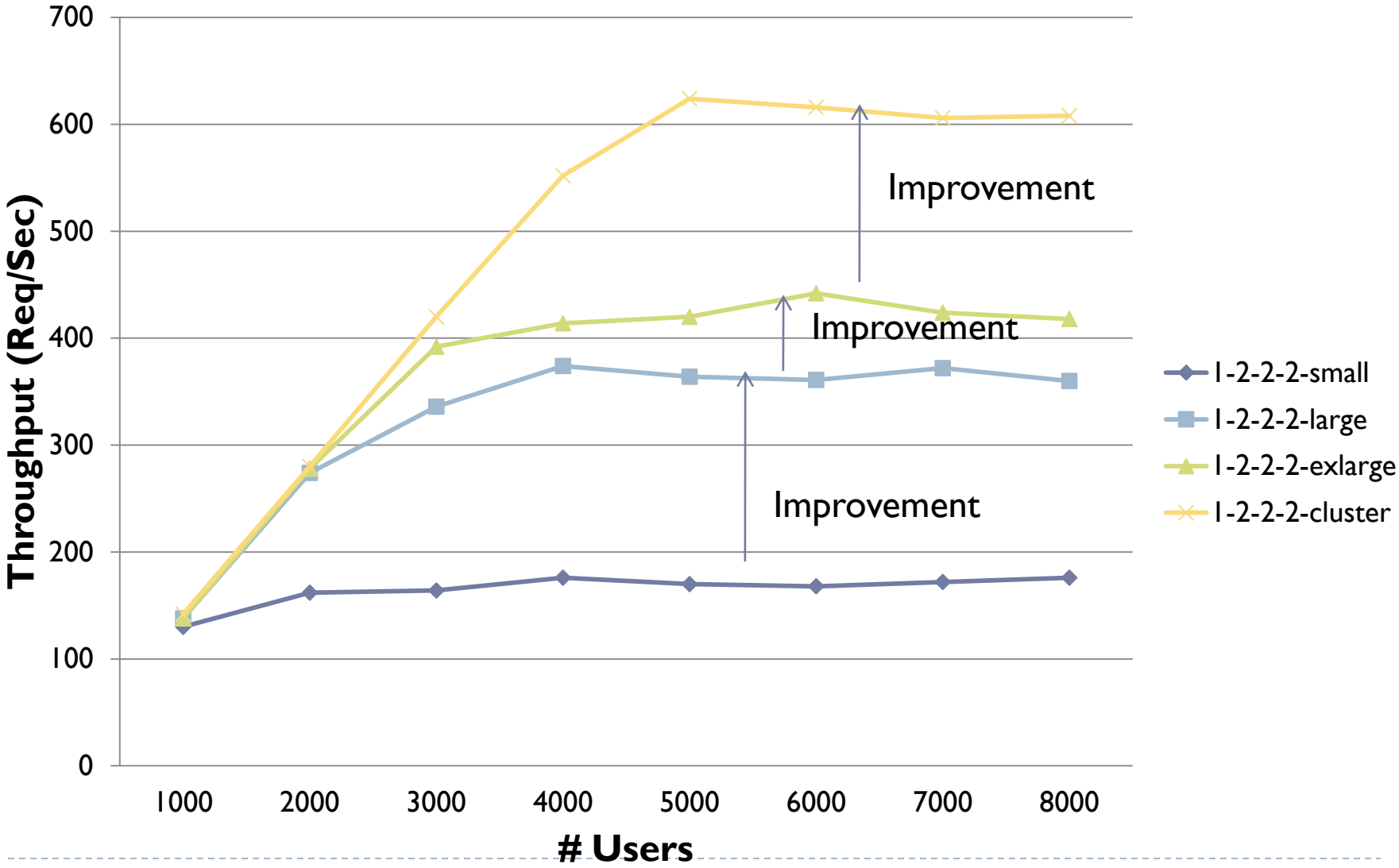



Tomcat MySQL Data Node

# Vertical Scalability on EC2



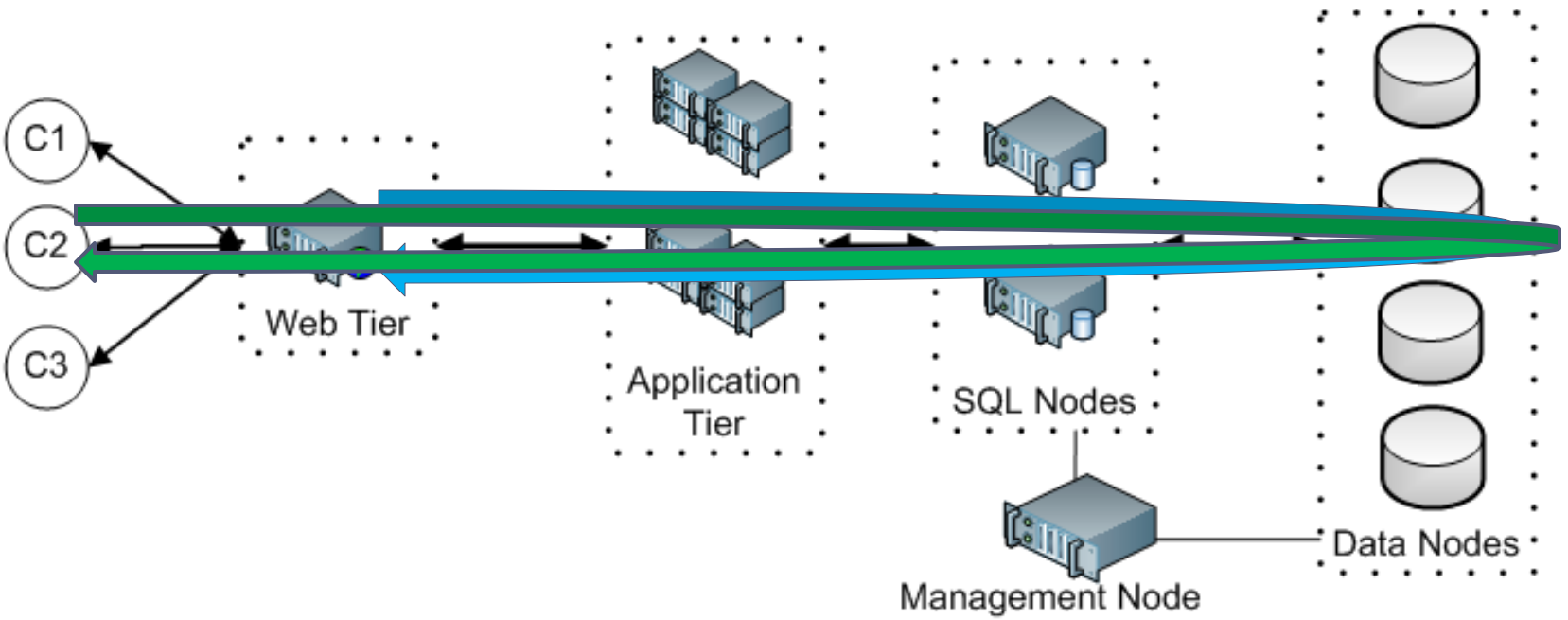
# Summary

- Emulab and Open Cirrus show better horizontal scalability
- EC2 show good vertical scalability, but not better horizontal scalability

- Our analysis resulted in two findings:
  - Multi-threading overhead
  - Network driver overhead

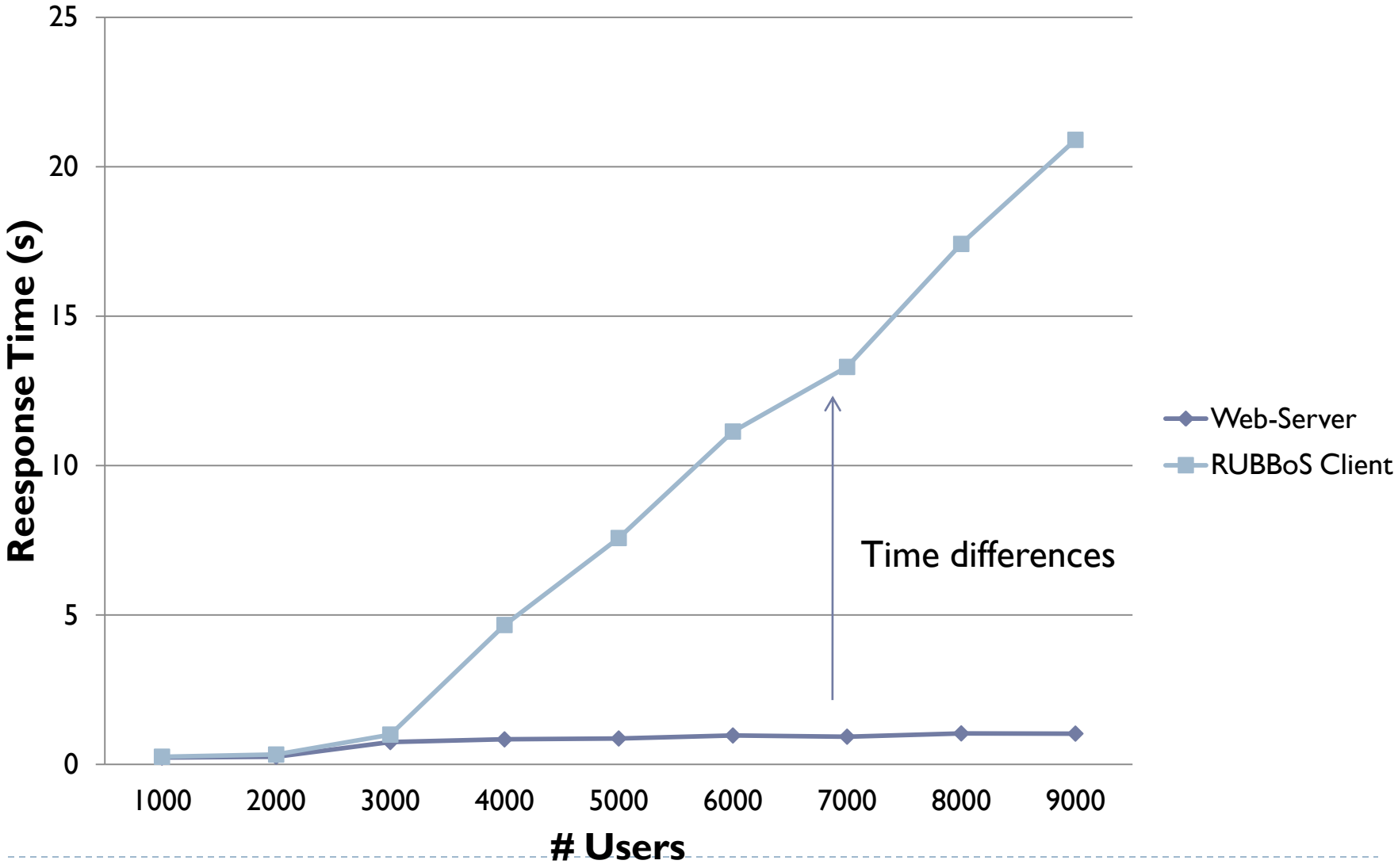
# Multi-Threading Overhead

# End-to-End Response Time

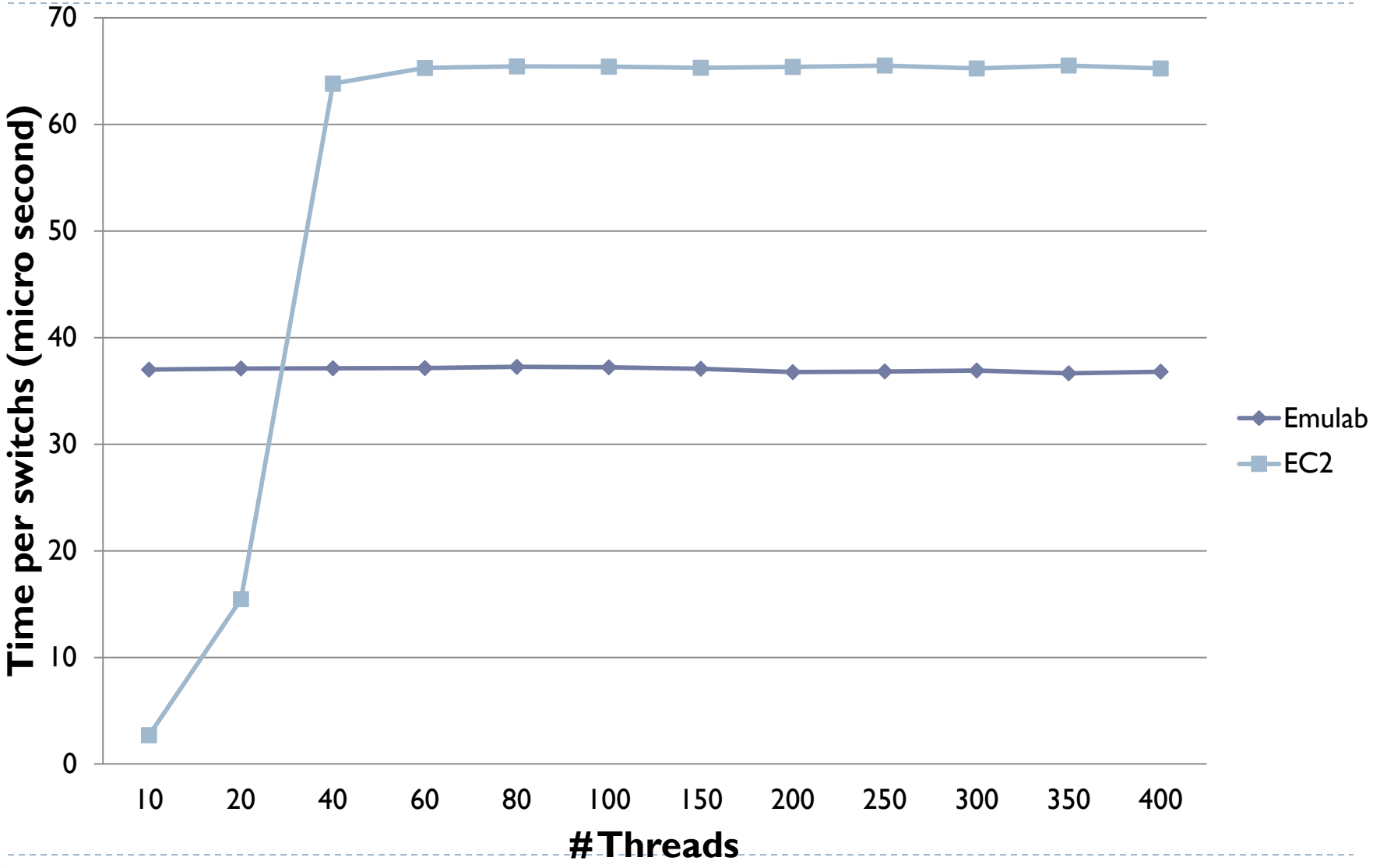



Client End-to-End
Web Server End-to-End

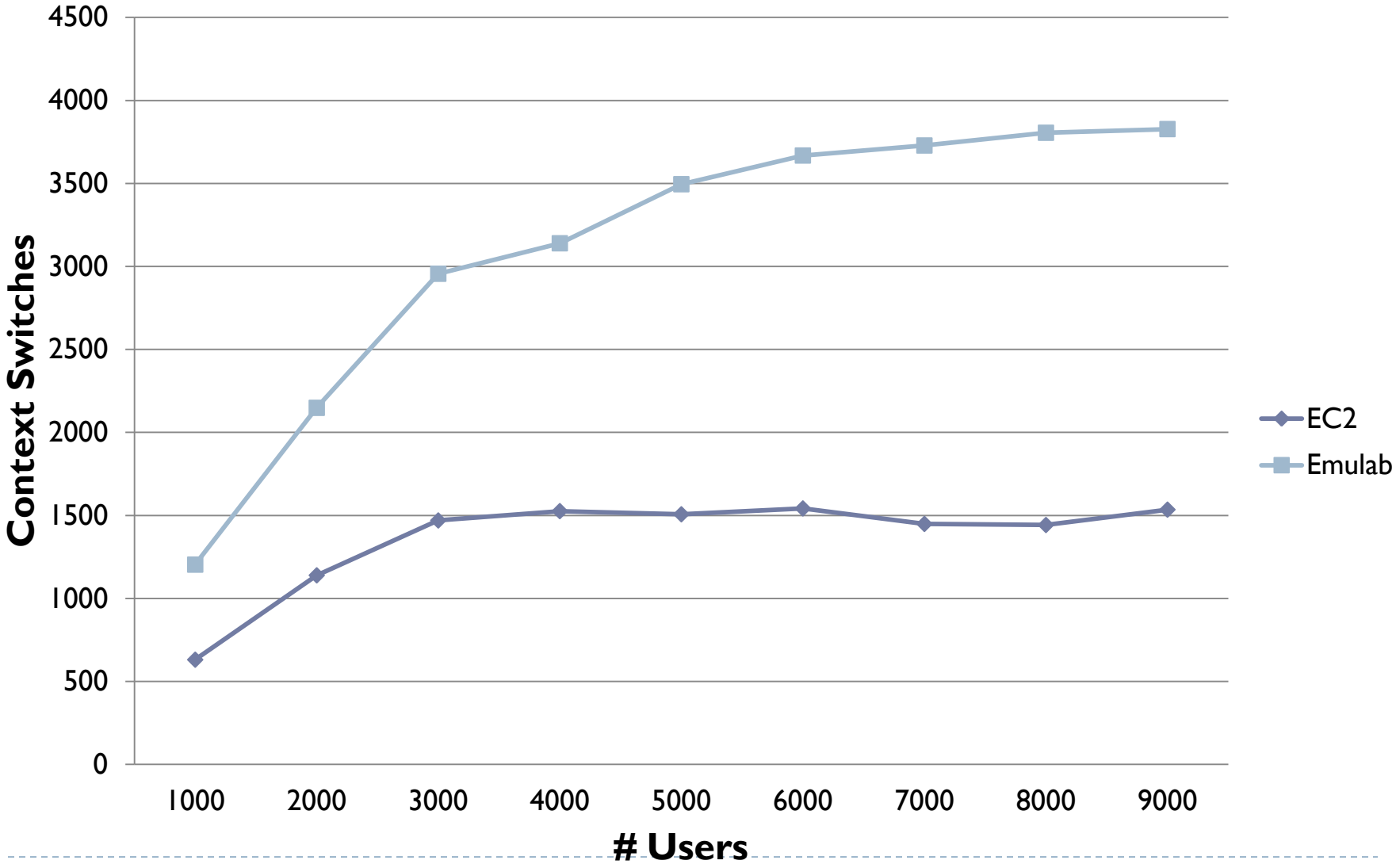
# Differences in Response Time



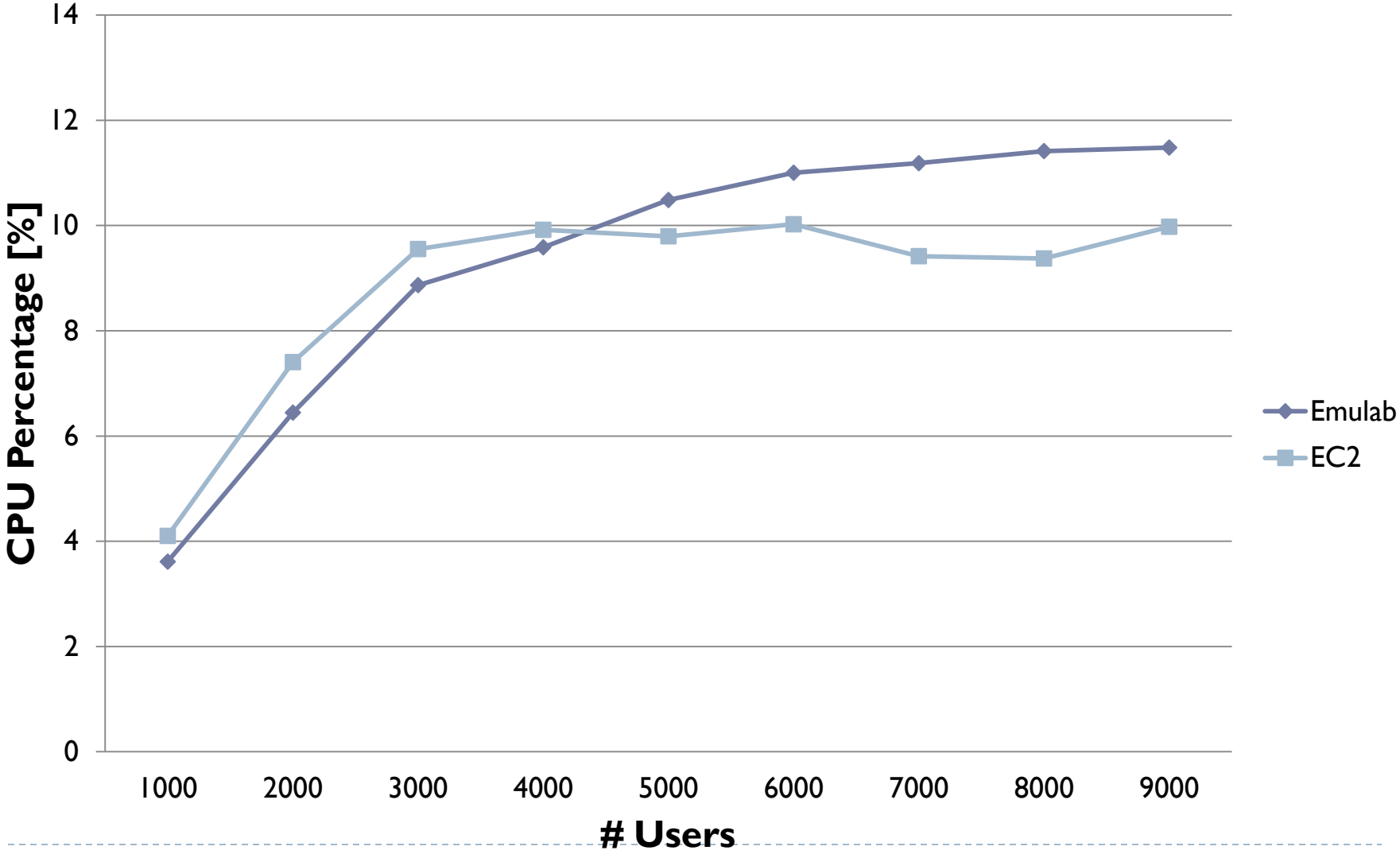
# Context Switching (using LMBench)



# Number of Context Switches



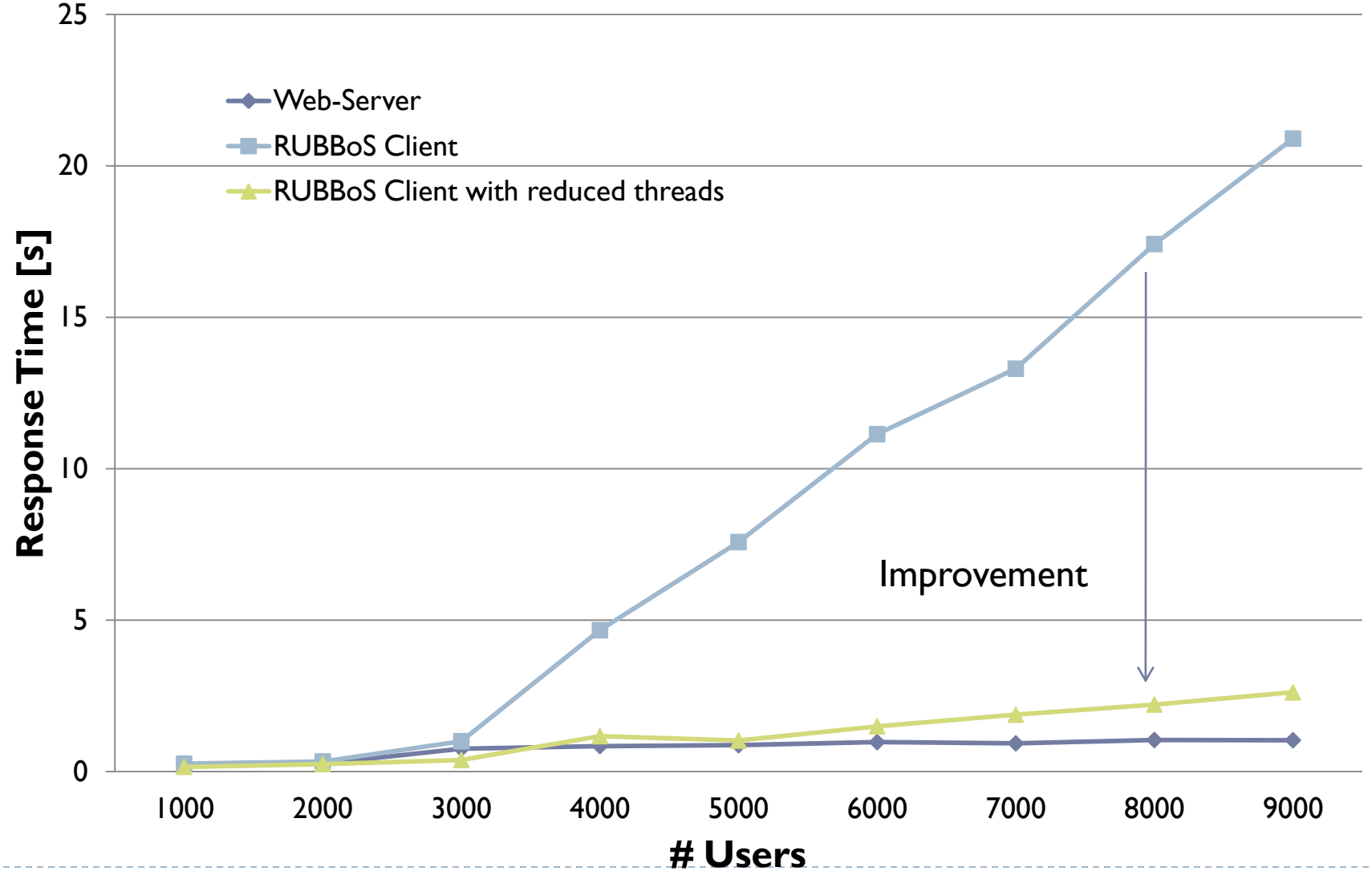
# CPU Percentage



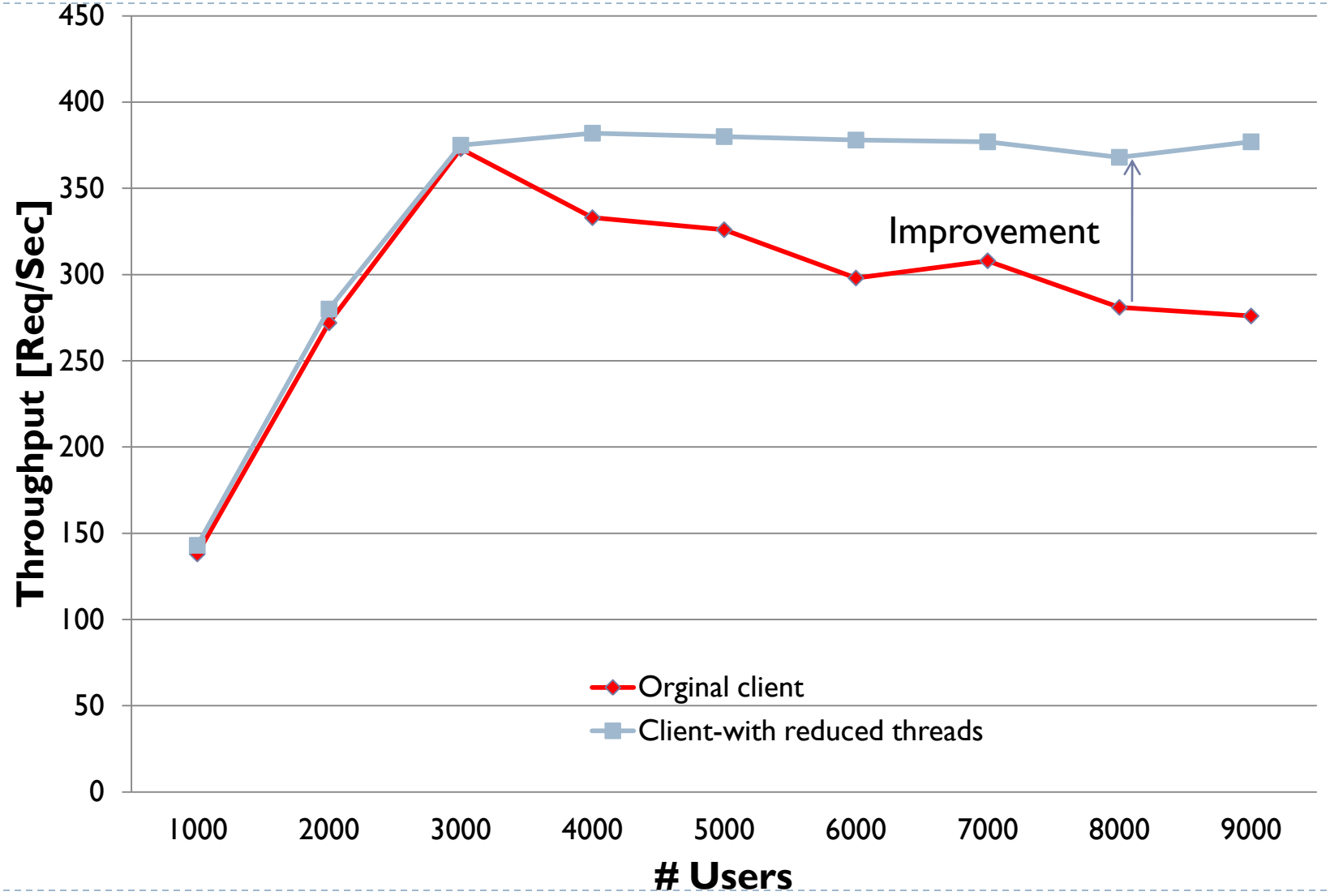
# Issues and Solution

- Overall throughput and number of context switches are related
- But, in EC2 we get less switches and similar overhead as Emulab
- When the number of threads is higher the application become unstable in EC2
- Solutions:
  - Rent many instances
  - Re-write the application with less number of threads

# With our Solution

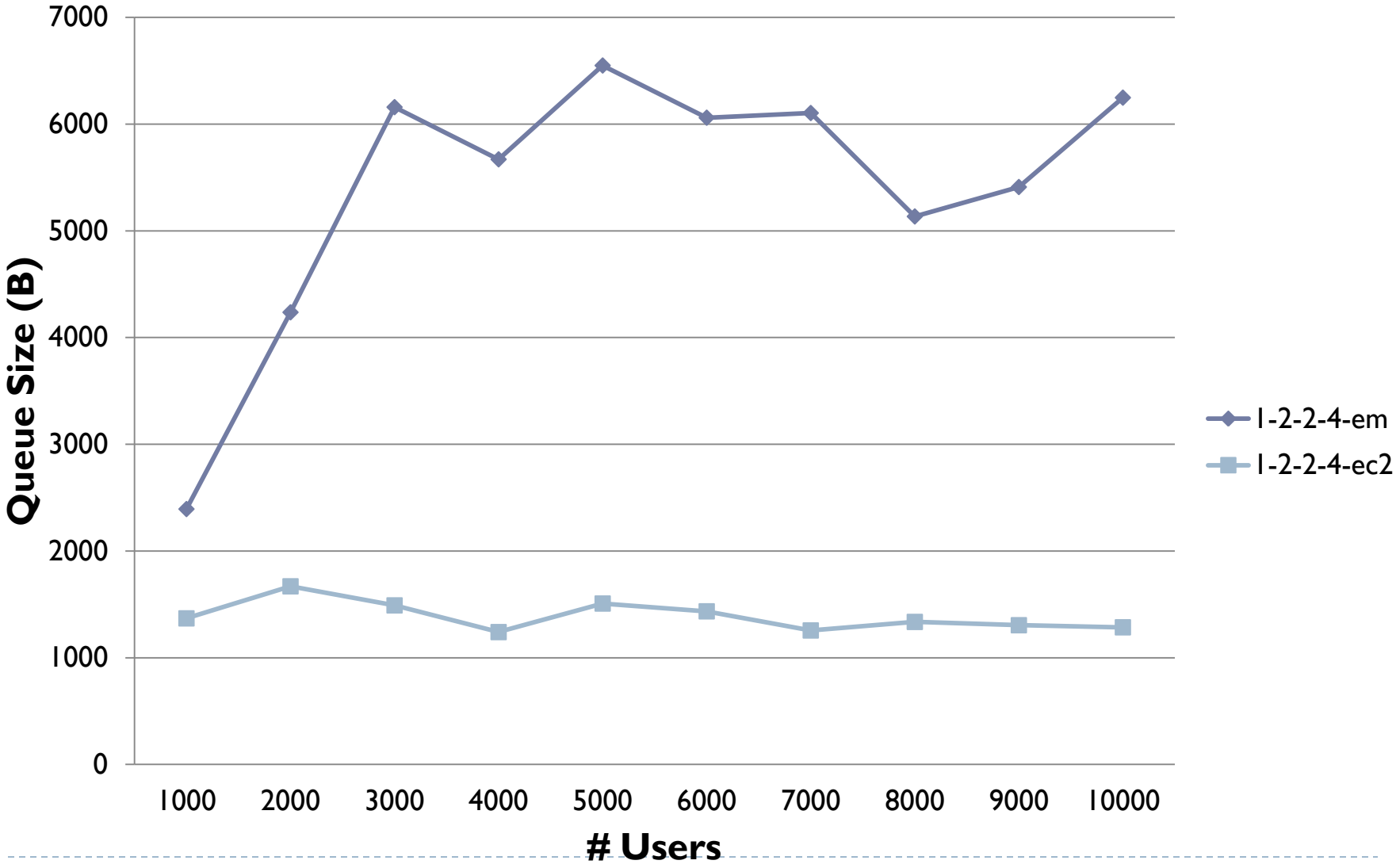


# Solution - Improvements

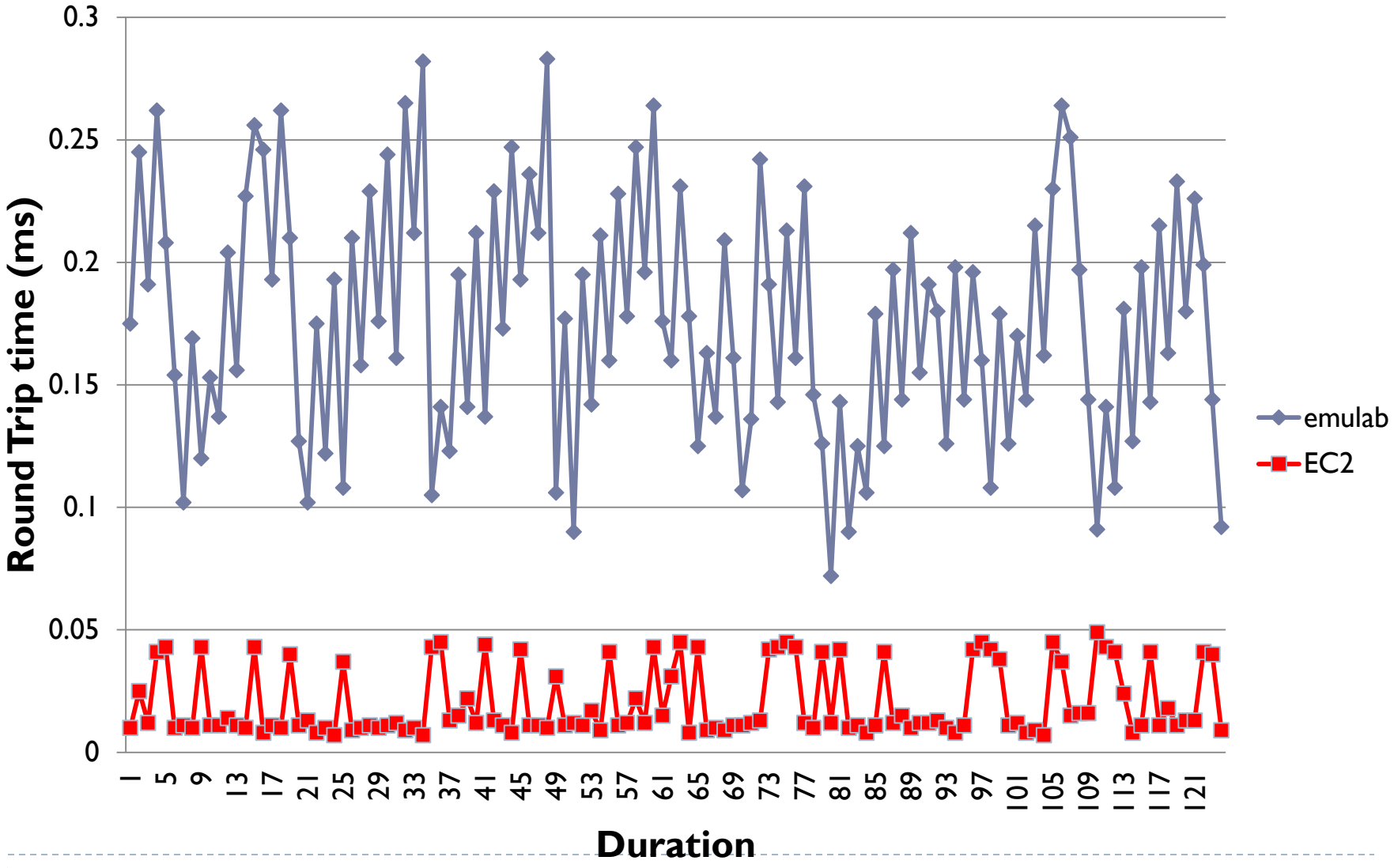


# Network Driver Overhead

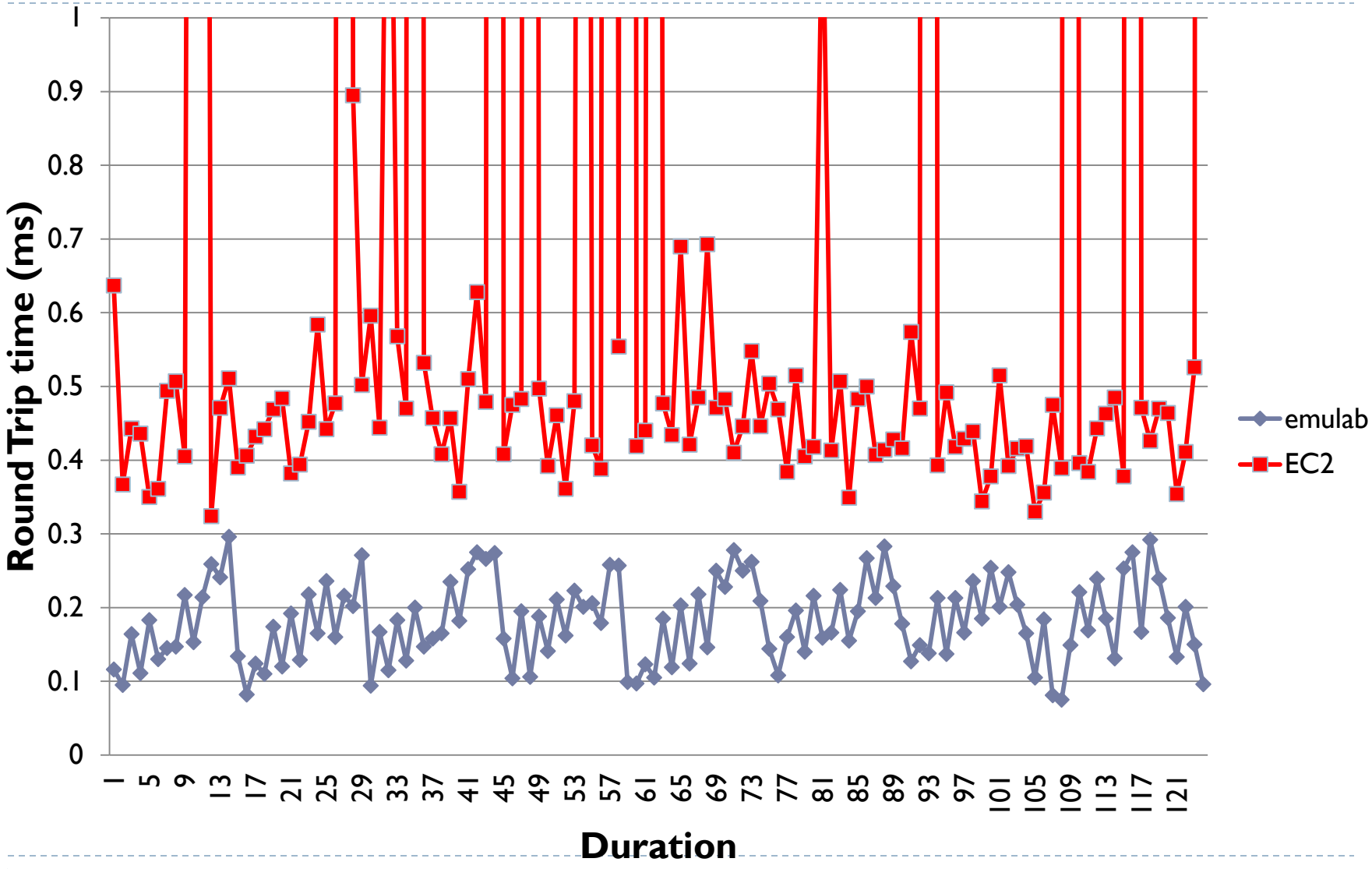
# Data Node Send Queue



# Ping-Pong Time without a Load



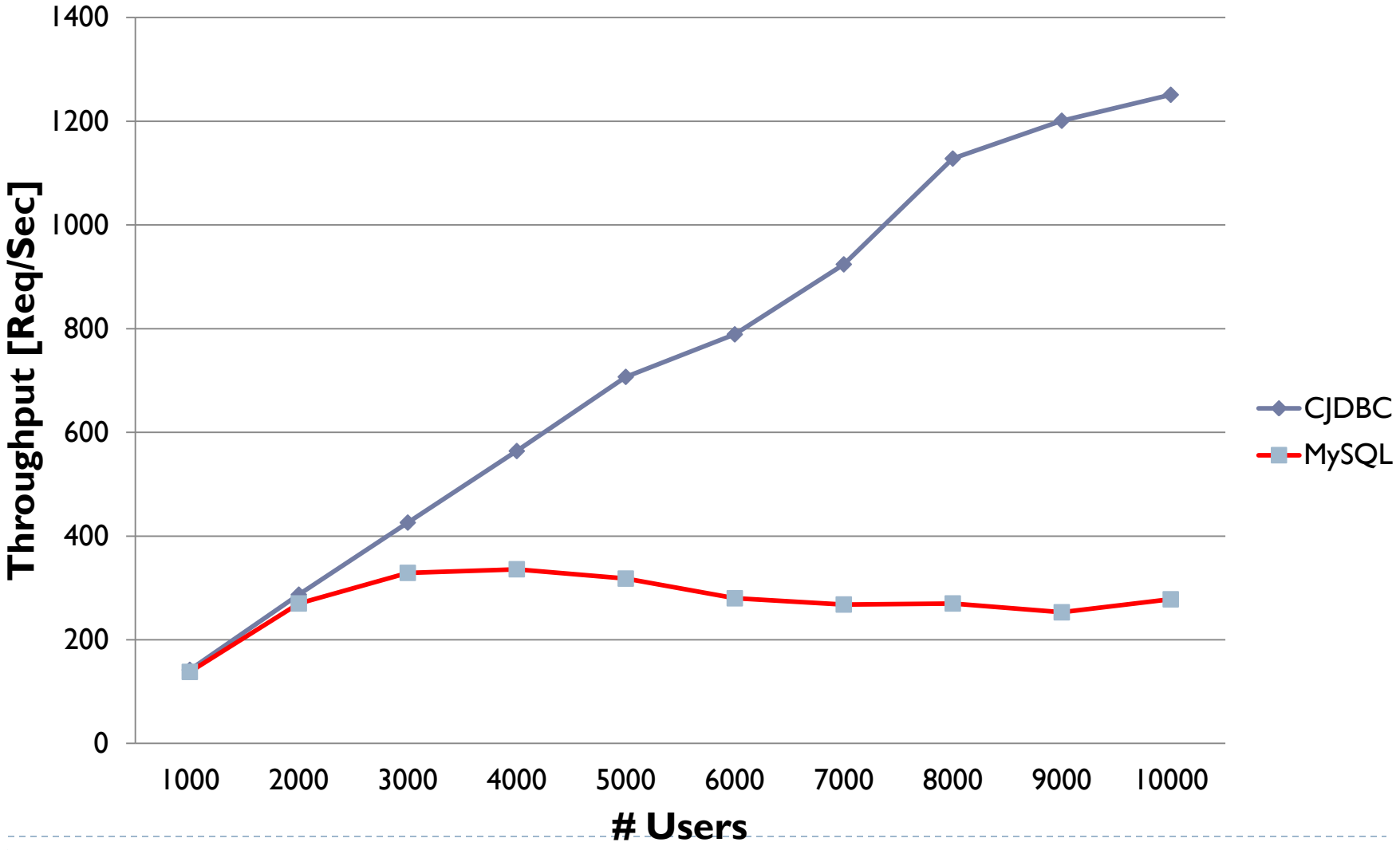
# Ping-Pong with a Load



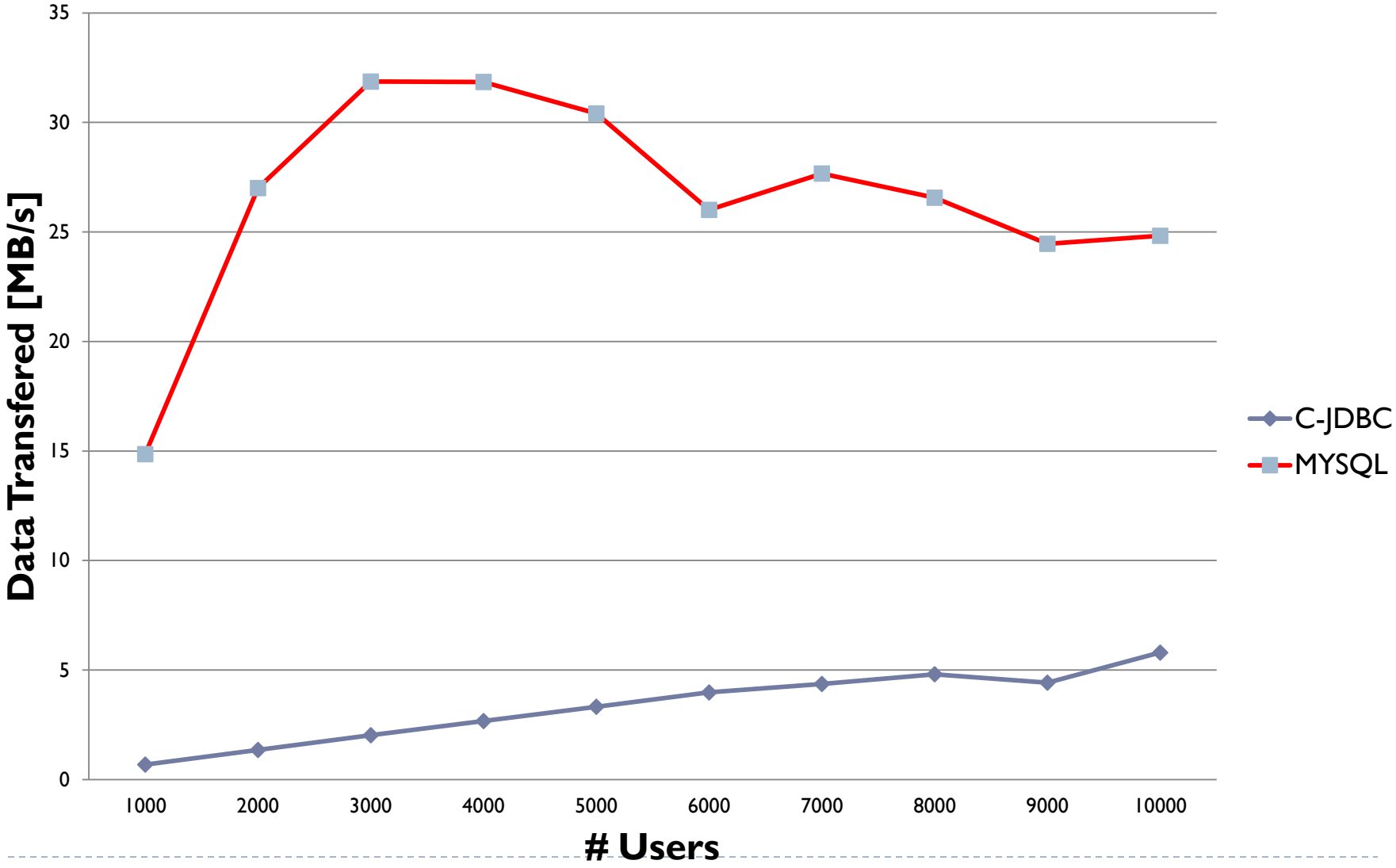
# Solution

- Use network friendly database middleware
  - Reduce the pressure on the network
- Evaluated C-JDBC, an open source database middleware
  - Performed well and showed a very good scalability
  - Reduce the network traffic significantly

# C-JDBC vs. MySQL Cluster Performance



# C-JDBC vs. MySQL Cluster Network Traffic



# Conclusion & Future works

- Studied the performance and scalability variations when migrating n-tier applications to clouds.
- Our results show, that cloud are new and need more experimental studies to better understand them.
- More specifically, application re-design is needed to handle cloud challenges.

- Future works
  - Extend our analysis into other clouds (e.g., Wipro)
  - Micro level study to dig deep into observe phenomena
  - With other database monument systems (e.g., Oracle, DB2)

# Thank you.