ANOMALY DETECTION IN CLOUD SYSTEMS USING MACHINE LEARNING

SAHIL CHADHA MANISH CHOUDHARY SHAILESH LOHIA ADITYA UPERTI

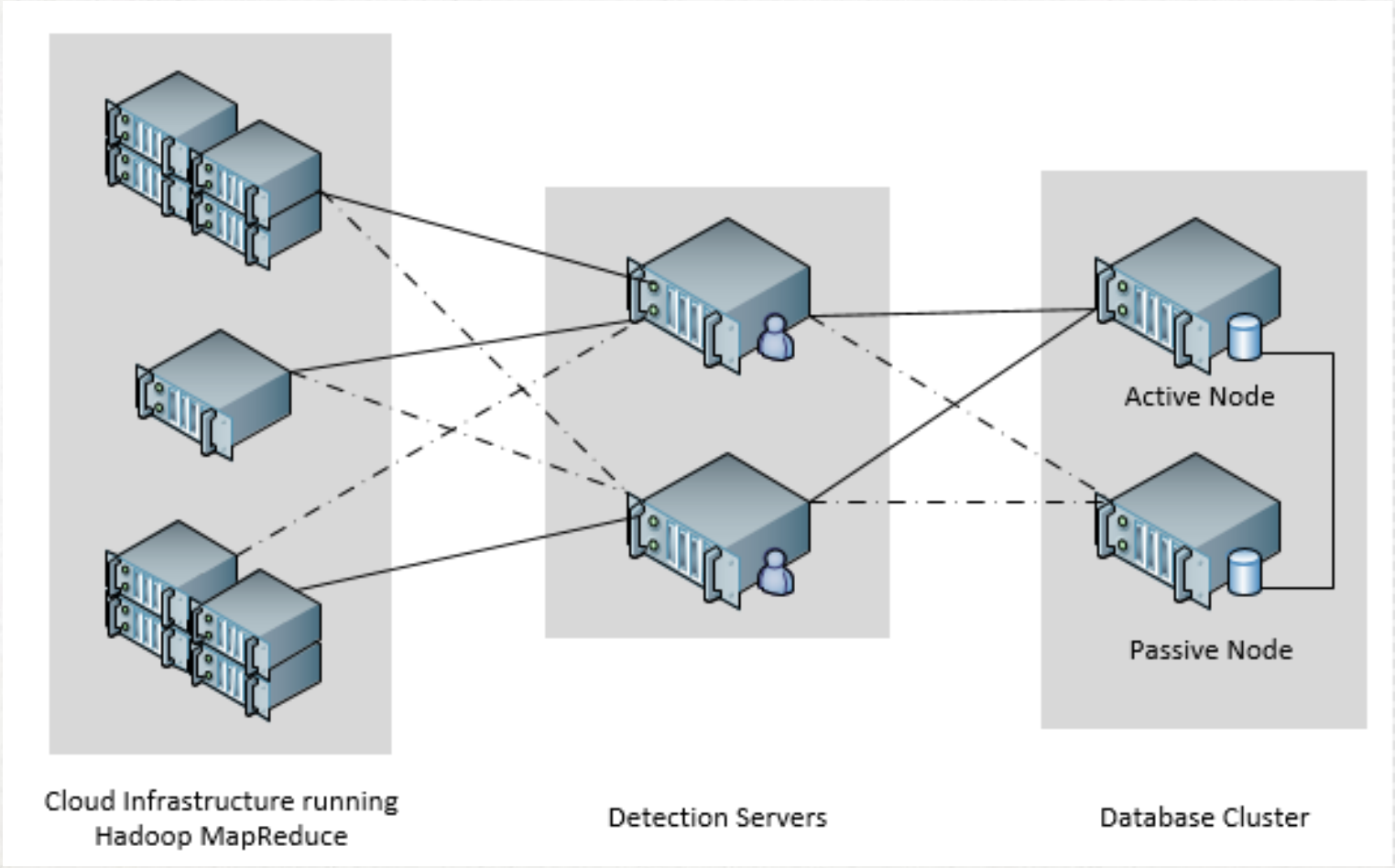
AGENDA

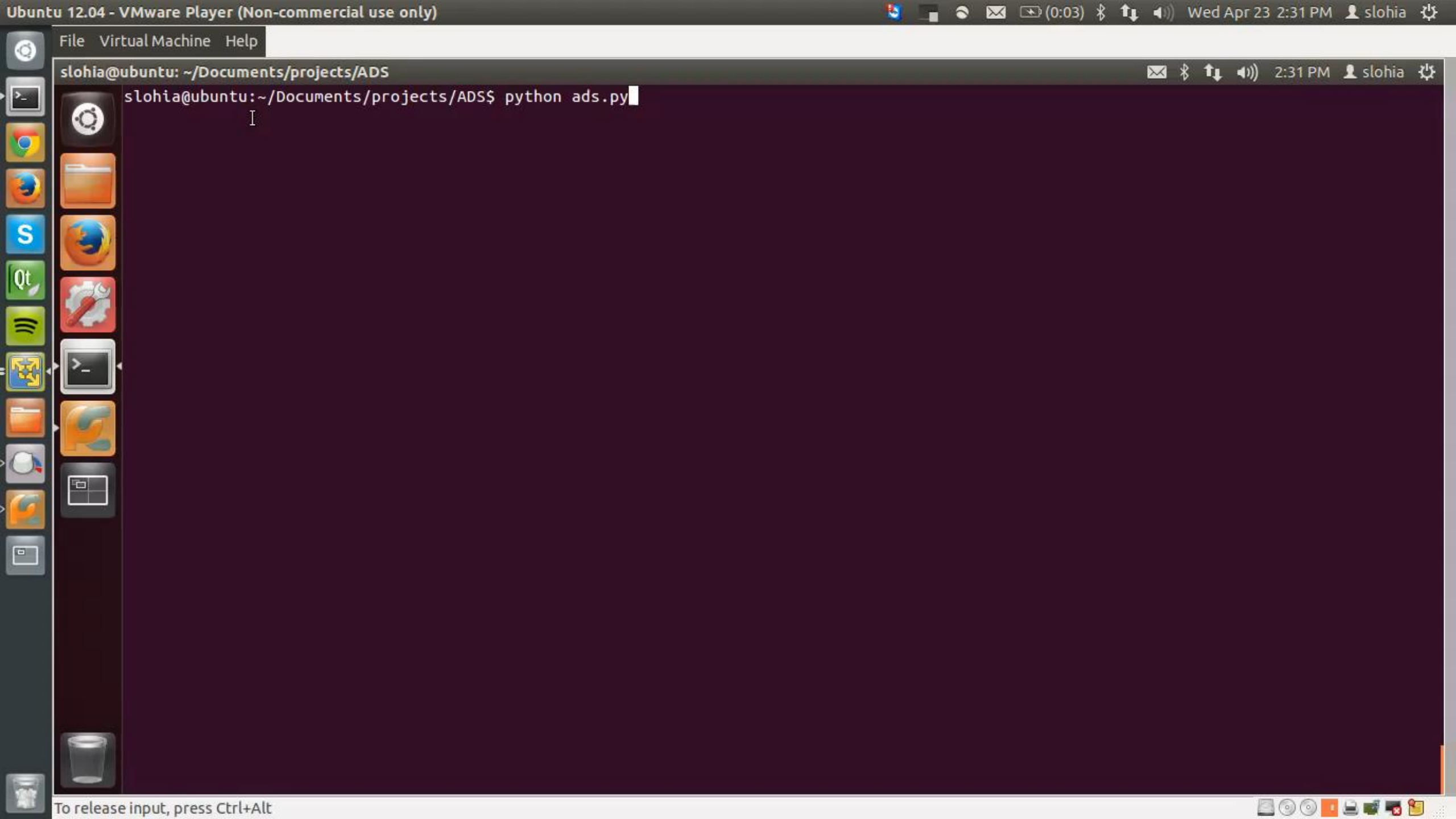
- Overview
- Architecture
- Demo
- Evaluation
- Discussion
- Q & A

PROJECT OVERVIEW

- Anomaly Detection in Cloud Systems
  - Learn optimum performance
  - Monitor the infrastructure
  - Detect anomalies
  - Report it.

ARCHITECTURE: PROTOTYPE



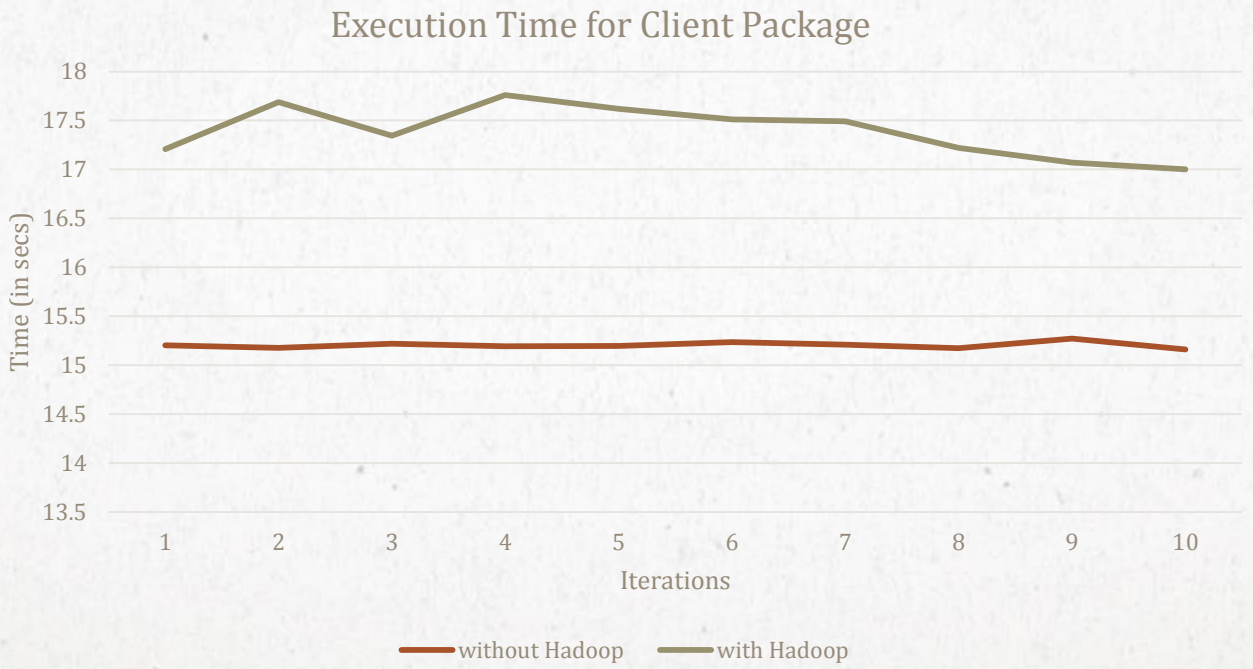

Ubuntu 12.04 - VMware Player (Non-commercial use only)
(0:03)
Wed Apr 23 2:31 PM
slohia
File
Virtual Machine
Help
slohia@ubuntu: ~/Documents/projects/ADS
2:31 PM
slohia
slohia@ubuntu:~/Documents/projects/ADS$ python ads.py
To release input, press Ctrl+Alt

EVALUATION

- Each node had the following configuration:
  - Dell PowerEdge 2850s with a single 3GHz processor
  - 2 processors
  - 2 GB of RAM
  - 2 10,000 RPM 146GB SCSI disks

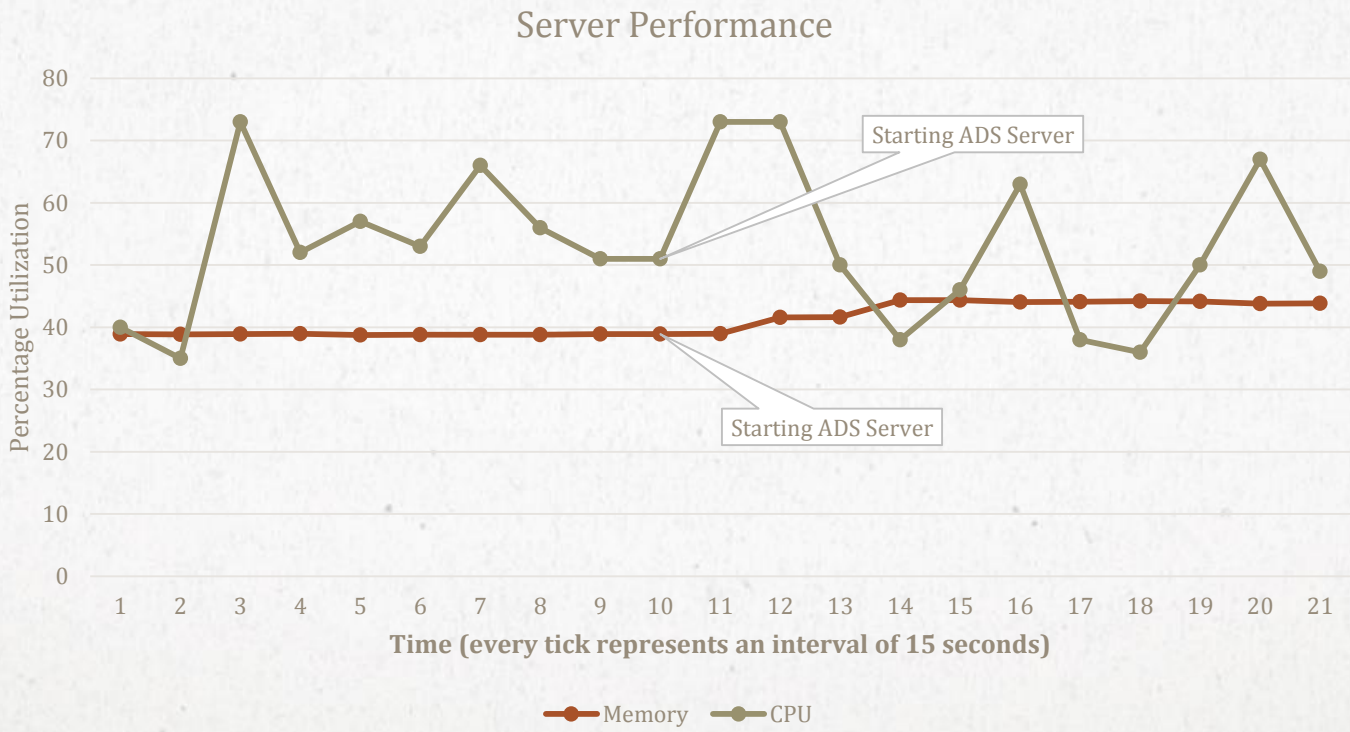
EVALUATION

- CLIENT PERFORMANCE



EVALUATION

- SERVER PERFORMANCE

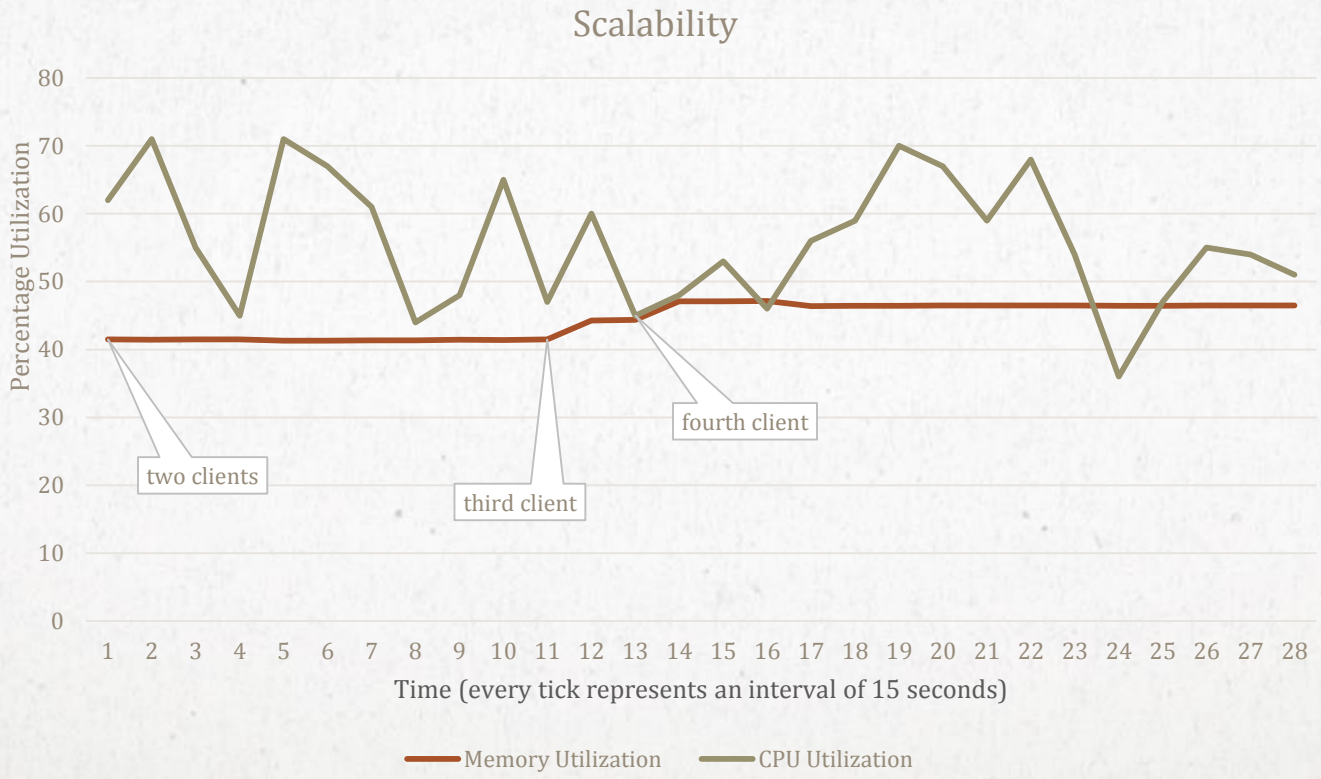


EVALUATION

- SERVER FAILOVER
  - Failover time is dependent on the configurable timeout period (50 seconds)
  - Additional cost incurred – one RPC call
    - Repo creation
    - Process spawning for learning
    - Time taken: 115 ms

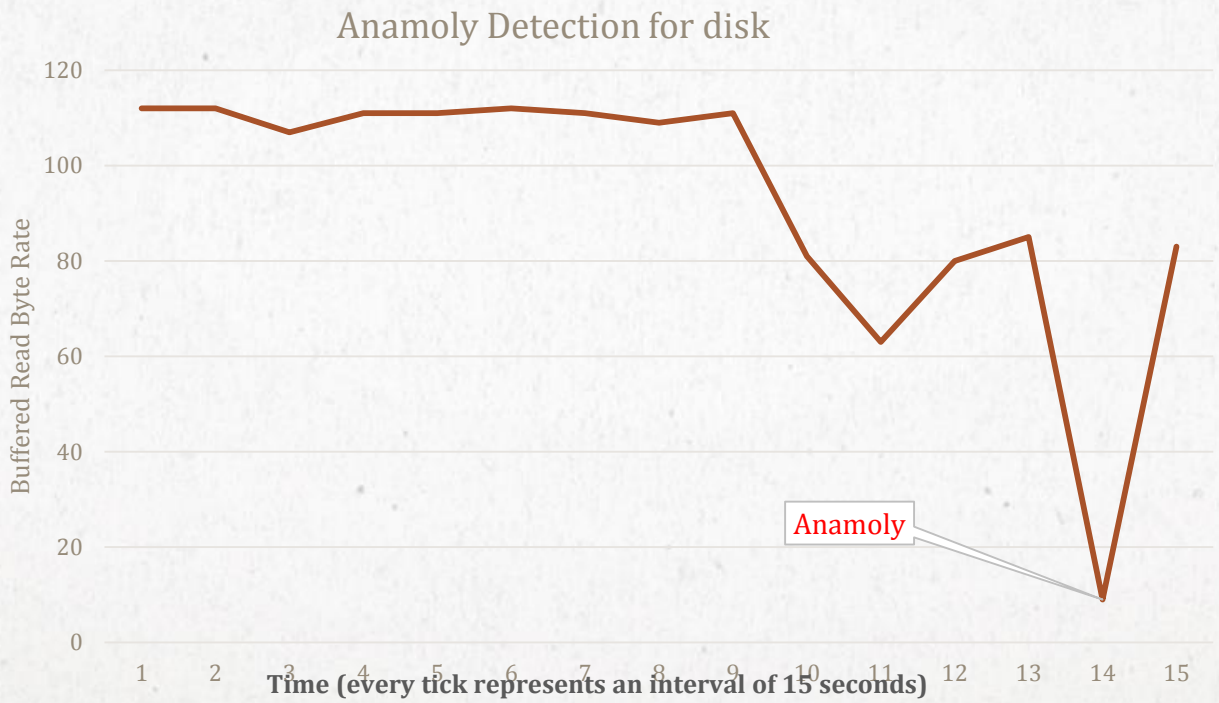
EVALUATION

- SCALABILITY



EVALUATION

- ANAMOLY DETECTION



DISCUSSION

- System built with a prototype cloud infrastructure
- Fault Tolerance
- Scalability
- Correctness

FUTURE WORK

- Implement other learning algorithms
- Perform a comparison study on which learning algorithm performs better
- Implementation of GUI
- Ability for administrators to make the system relearn the performance of a particular ADS client