R6.sm4 »

EC2 Performance Analysis for Resource Provisioning of Service-Oriented Applications

Rais Mense
Fri, 02 Mar 2012

Jiang Dejun, Guillaume Pierre, Chi-Hung Chi
VU University Amsterdam, Tsinghua University Beijing

What does that mean?

Can we profile Amazon EC2 instances so we can more efficiently deploy applications?

Roadmap

- What are we dealing with here?
- Experiment setup
- Results
- Conclusion

Roadmap

- What are we dealing with here?
- Experiment setup
- Results
- Conclusion

What about cloud?

- Pay as you go
- On demand scaling
- VM on a shared hardware node

Why do we want to do this?

- Scientific applications
- Time critical applications
- Multi tiered applications

Why current system don't work on clouds

- Ressource provisioning systems exist, but...
  - Assume pre defined numbers of nodes
  - Assume ressources are stable
  - Assume ressources are homogeneous

Why current system don't work on clouds

- Ressource provisioning systems exist, but...
  - Assume pre defined numbers of nodes
  - Assume ressources are stable
  - Assume ressources are homogeneous

They are just not made for clouds.

Roadmap

- What are we dealing with here?
- Experiment setup
- Results
- Conclusion

Experiment 1/3

- Amazon EC2
- Small instances
- Different locations

Experiment 2/3

- T1 - CPU
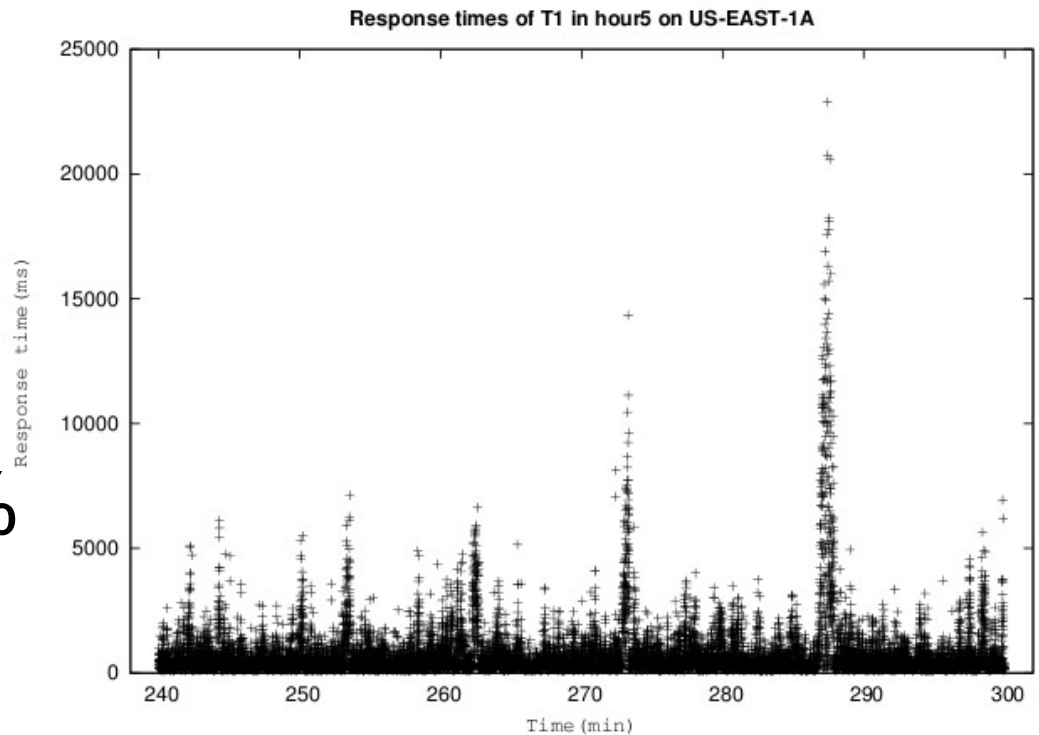- T2 - Database reads
- T3 - Database writes

Experiment 3/3

- Stability - (24h)
- Homogeneity - (5 x 6h with a new instance)

- Correlation - (6h concurrent)

Roadmap

- What are we dealing with here?
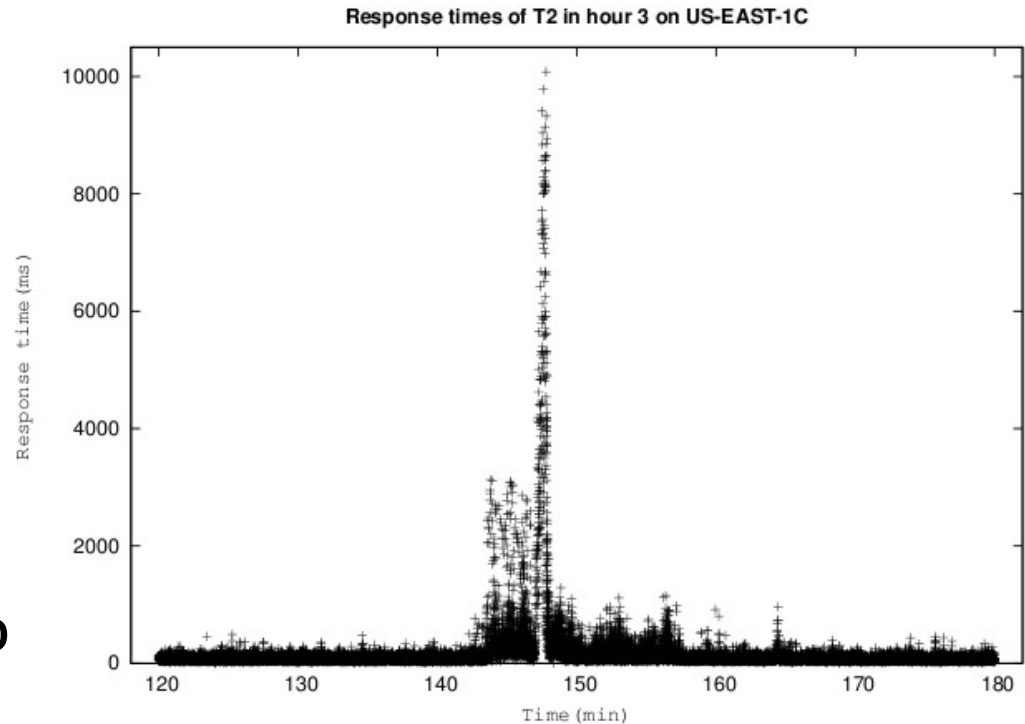- Experiment setup
- Results
- Conclusion

Results - Stability

- T1 (CPU)
  - Fairly reliable performance
  - Std dev. 1,9% and 6,8%

Response times of T1 in hour5 on US-EAST-1A

(a) Hour 5

Results - Stability

- T2 (Reads)
  - Also mostly reliable performance
  - Std dev. 1,7% and 8,0%

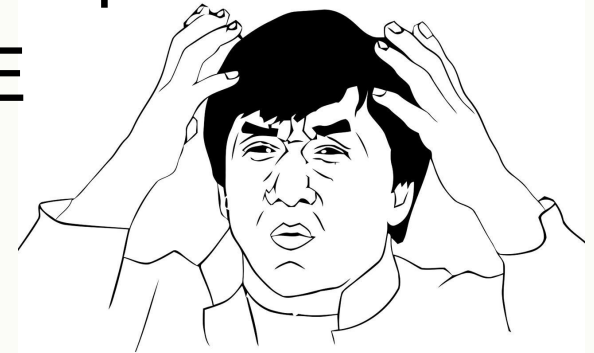Response times of T2 in hour 3 on US-EAST-1C

(a) Hour 3

Results - Stability

- T3 (Writes)
  - INSERT : Pretty fast, std dev. 0,9% max.
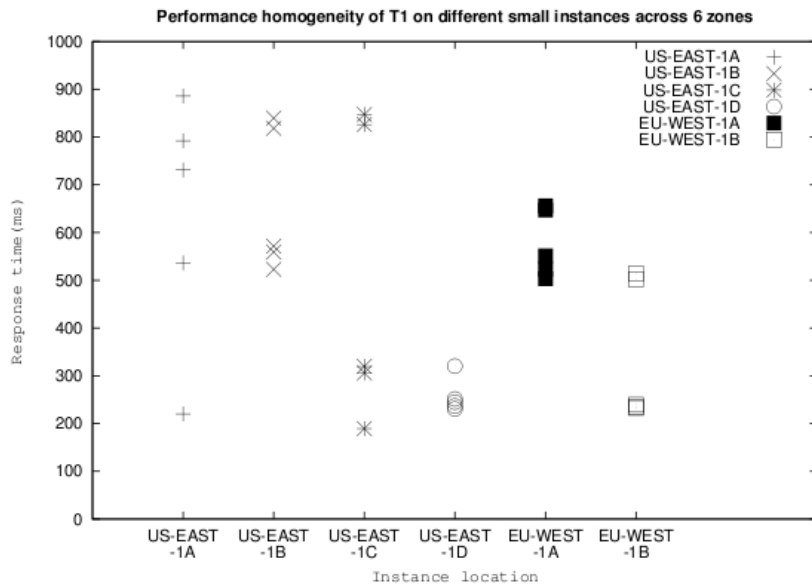 - UPDATE : Still fast, std dev. 2,3% max.

Results - Stability

- ## T3 (Writes)
  - INSERT : Pretty fast, std dev. 0,9% max.
  - UPDATE : Still fast, std dev. 2,3% max.
  - DELETE : Just terrible, std dev. up to **71.1%** and ~10x slower than UPDATE
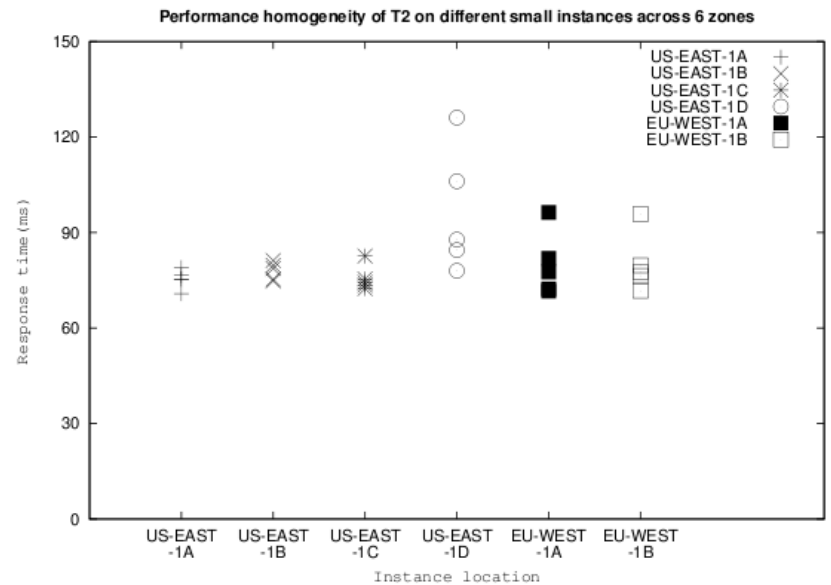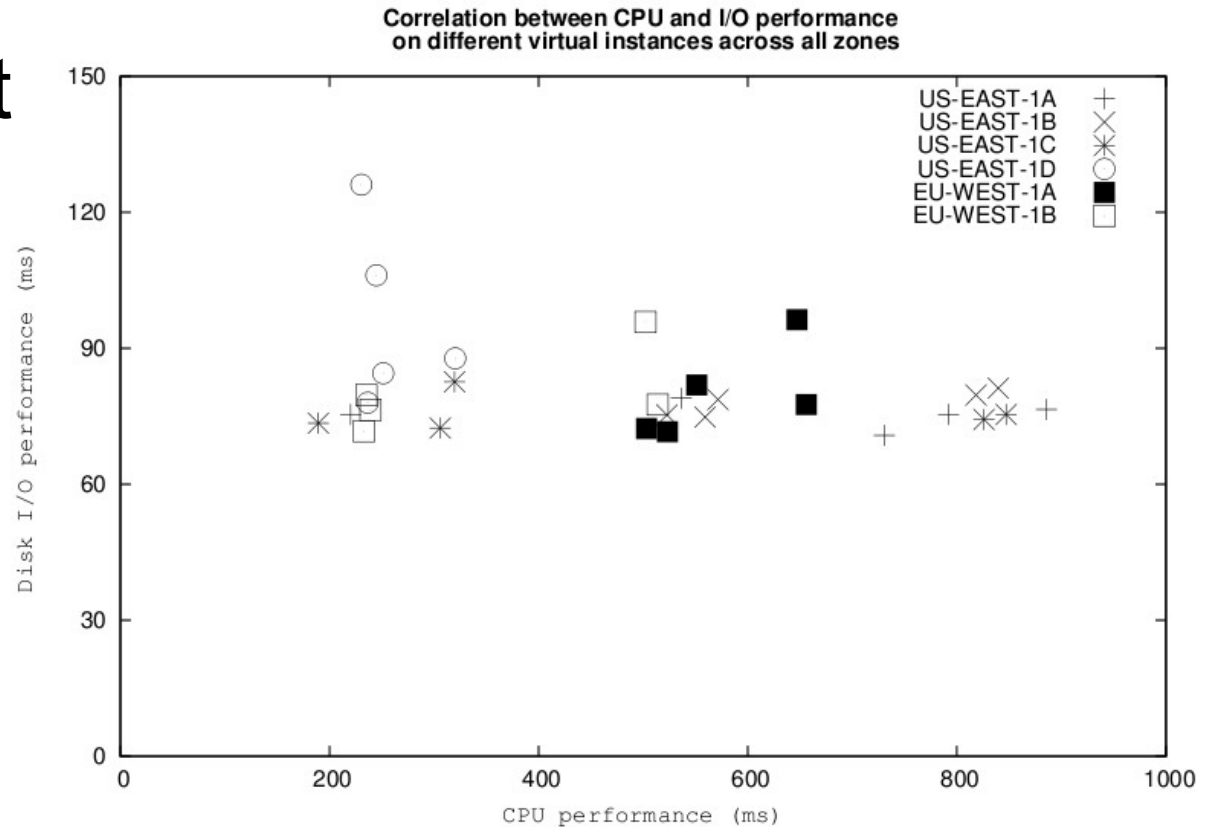
Results - Homogeneity

- Up to 4x difference

(a) CPU performance homogeneity

(b) Disk I/O performance homogeneity

Results - Correlation

- There isn't any

Correlation between CPU and I/O performance
on different virtual instances across all zones

Roadmap

- What are we dealing with here?
- Experiment setup
- Results
- **Conclusion**

In conclusion

- Instances of the same type do not guarantee the same preformance

- Not all instances are created equally

- **Profile your virtual machine instances before deploying applications!**