T-110.5121 Resource Provisioning 28.11.2012

Yrjö Raivio,
Ramasivakarthik Mallavarapu
Aalto University, School of Science
Department of Computer Science and Engineering
Data Communications Software
Email: yrjo.raivio(at)aalto.fi
Course email: t-110.5121(at)tkk.fi

Aalto University
School of Science

Agenda

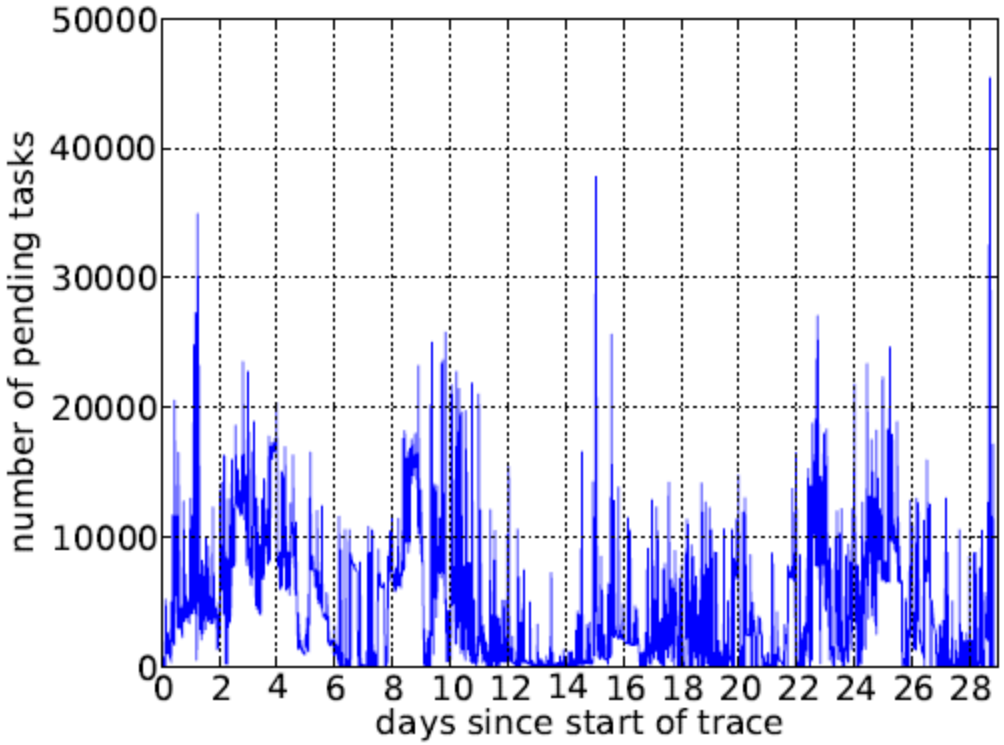
- Load migration
- Load balancing
- Auto scaling
- Reactive model
- Predictive model
- Algorithms and examples
- Conclusion

Cloud computing can improve scalability and availability

How The Weather Company survived a 1,000% traffic spike during Hurricane Sandy

Source: http://venturebeat.com/2012/11/02/how-the-weather-company-survived-a-1000-traffic-spike-during-hurricane-sandy/

Large Google computer cluster trace

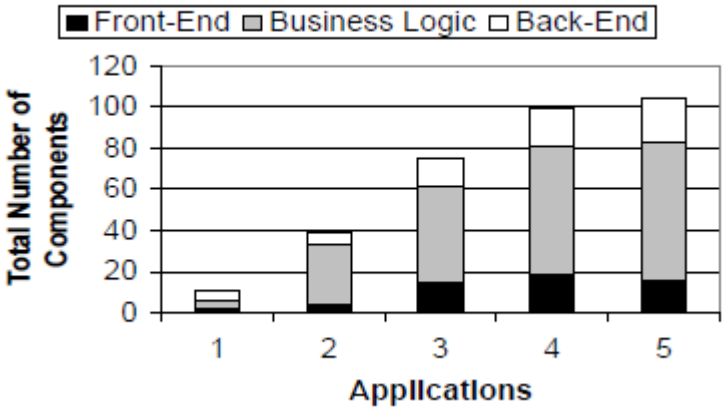


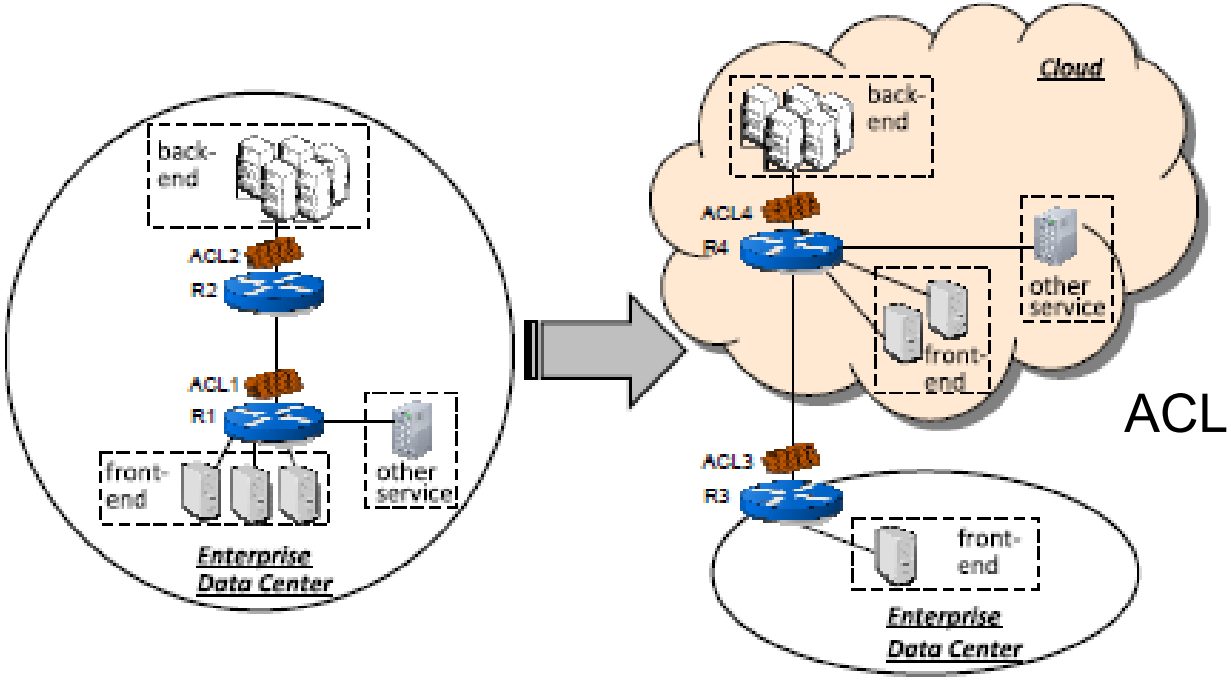
Source: C. Reiss et co, Towards understanding heterogeneous clouds at scale: Google trace analysis. 2012

Background

- Traditional Datacenters
  - Fixed and dedicated infrastructure → Expensive and inefficient
  - Unexpected workload peaks → Performance degrade
  - QoS critical services cater to peak workloads → under-utilized infrastructure
- Public IaaS Cloud Environments
  - Pay-per-use → Cost effective
  - On demand → Efficient
  - Elastic → Scalable

Cloud migration

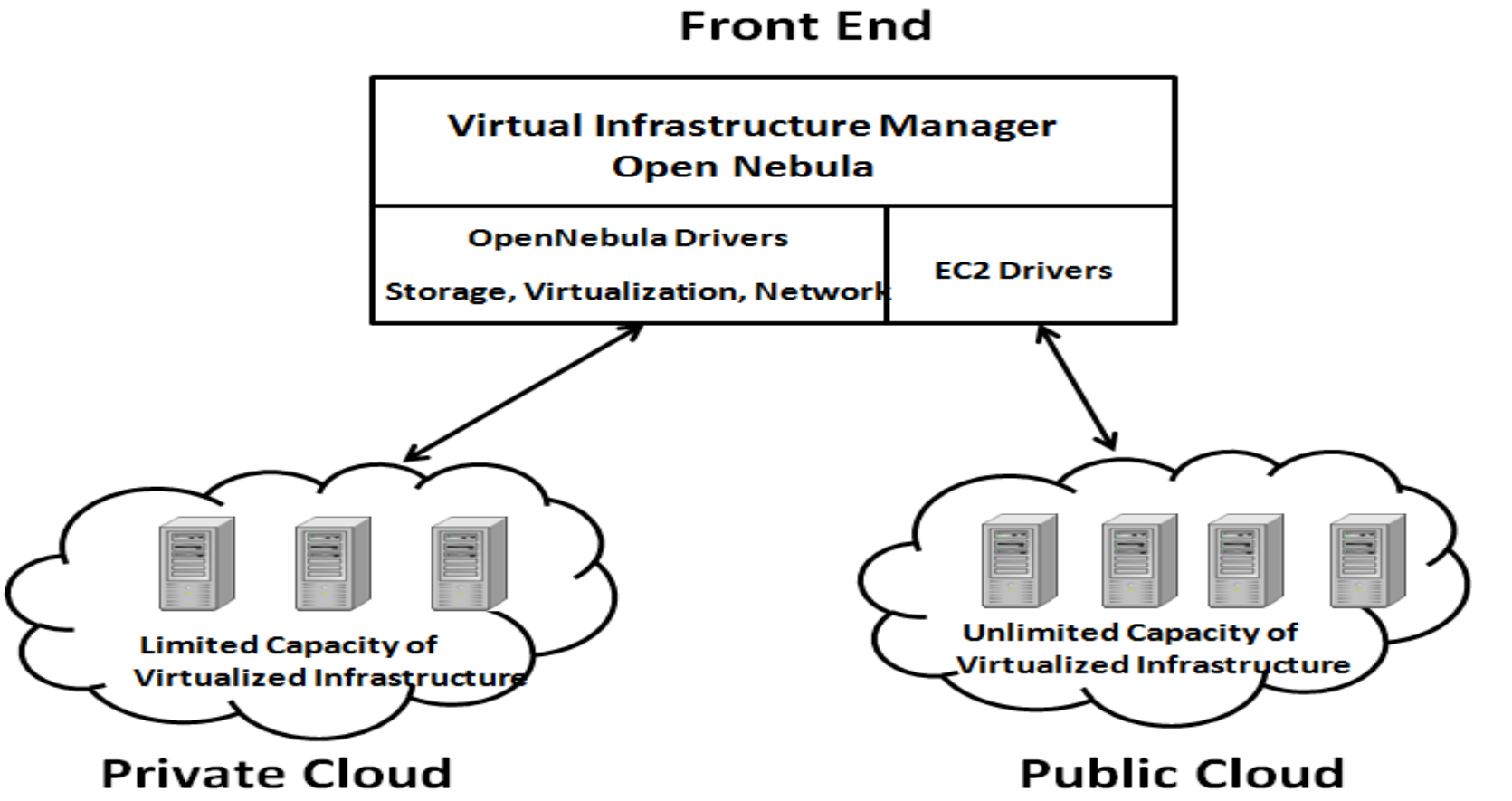




ACL = Access Control List

Source: M. Hajjat et co, Cloudward Bound: Planning for Beneficial Migration of Enterprise Applications to the Cloud, 2010

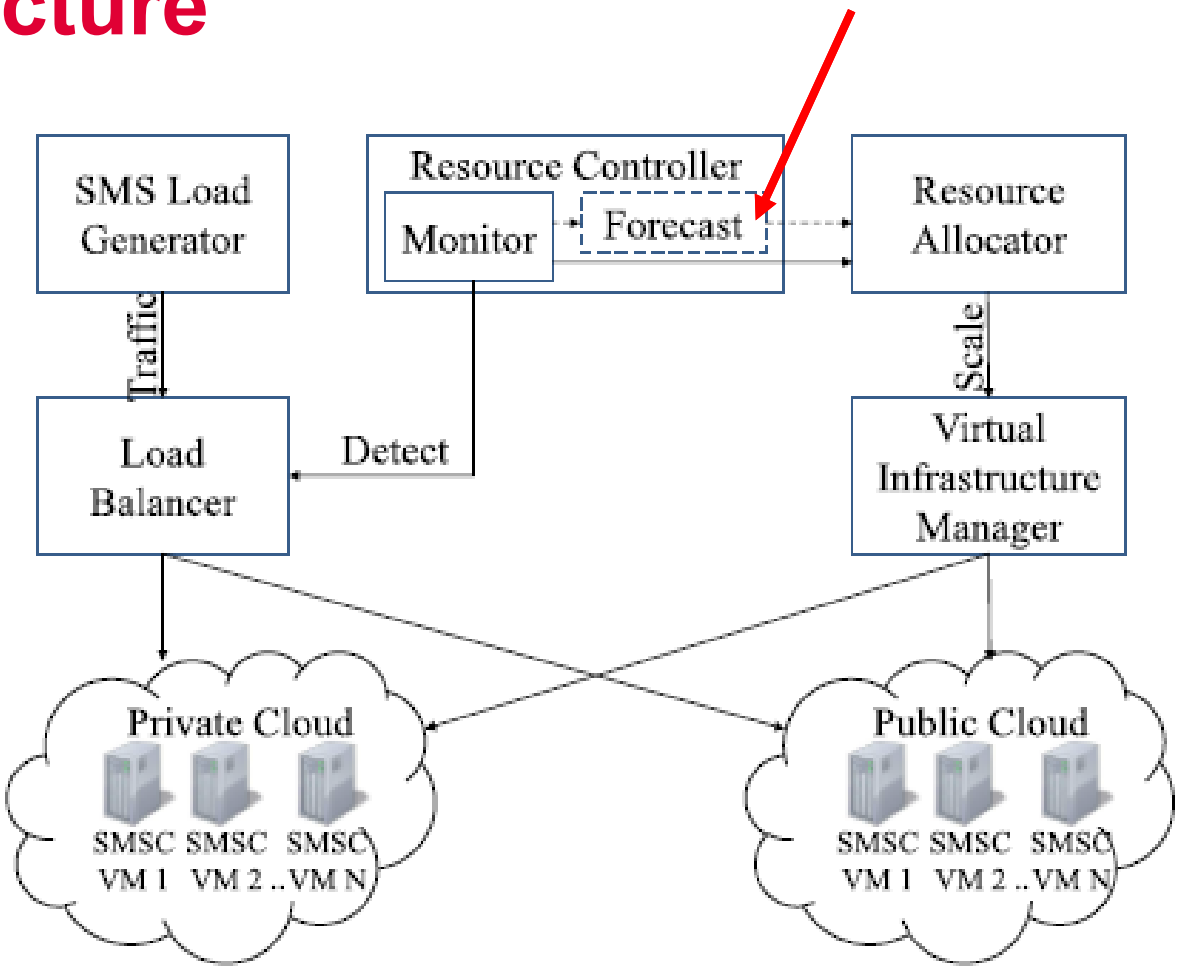
Load balancing



Auto scaling

- Auto-scaling refers to dynamically adapting the infrastructure by scaling up/down of resources based on the incoming workload traffic pattern
- Resource controller must
  - Monitor
  - Analyze
  - Act
- Metrics that trigger the infrastructure changes are termed as “Key Performance Indicators” (KPI)
- KPI typically, could be
  - CPU/Memory usage
  - Disk I/O
  - Network I/O

Architecture



Classification

- **Resource controllers can be broadly classified in two types**
  1. Simple reactive resource controller (Reactive)
     - Detect changes in workload pattern and react to changes ***after*** the *event* occurs
     - Suitable for services with predictable workload patterns
     - Unreliable for QoS critical services
  2. Look ahead resource controller (Predictive)
     - Predict/forecast changes in workload based on a recent history and react ***before*** the *event* occurs
     - Can cater to variable and unpredictable workloads
     - Efficiency largely depends on the prediction algorithm

Reactive model

- Detect excess workload and scale resources accordingly
- Existing infrastructure must cater to the excess load until newly launched resources are operational
- VM launch times are non-trivial. Launch time for an Amazon EC2 Large instance is 70-80 seconds (at least 3-4 minutes for enterprise application servers)
- Services with a stringent SLA may have adverse effect
- Suitable for non-critical services

Predictive model

- Model the incoming workload pattern
- Based on a recent history of workload data, predict (forecast) the future workload
- Resources are scaled before occurrence of the *event*
- Suitable for performance/latency critical services
- Most useful for variable incoming traffic and unpredictable workload patterns
- Example use cases: Telecom components, online ticketing services, e-commerce applications etc.

Moving averages model

- Forecast is based on the most recent observations
- More than prediction, this technique is an estimation process
- Represented by the equation:

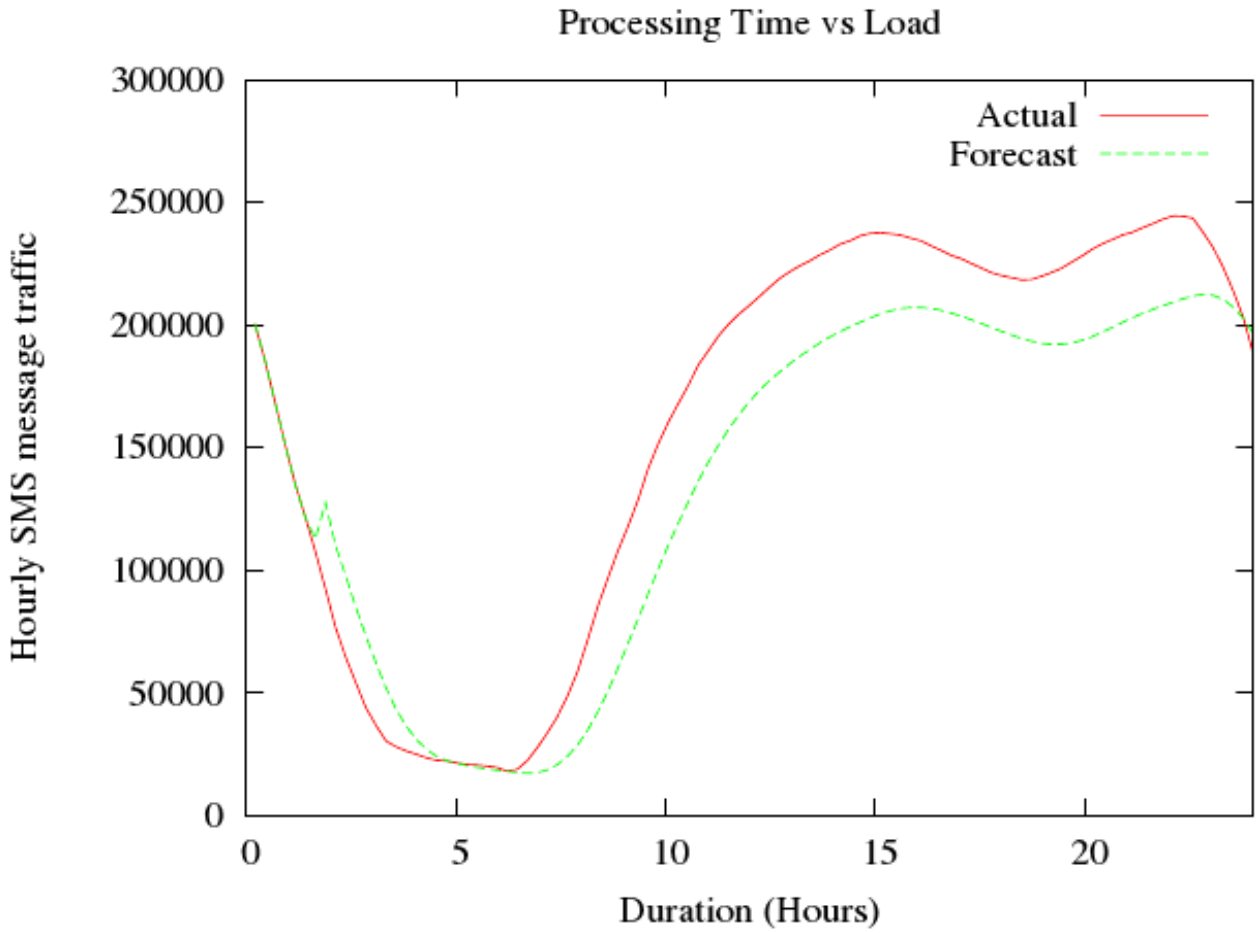
  $X'(t) = (X(t-1) + X(t-2) + \ldots + X(t-k)) / k$
- Value of k varies with the time series.
- Often, only the most recent observations are considered
- A slightly advanced version of MA model, is the weighted moving averages model
- Data observations are assigned weights in decreasing order
- Dampens the peaks, smoothens the valleys
- Simplistic estimation method, not very accurate

Algorithms

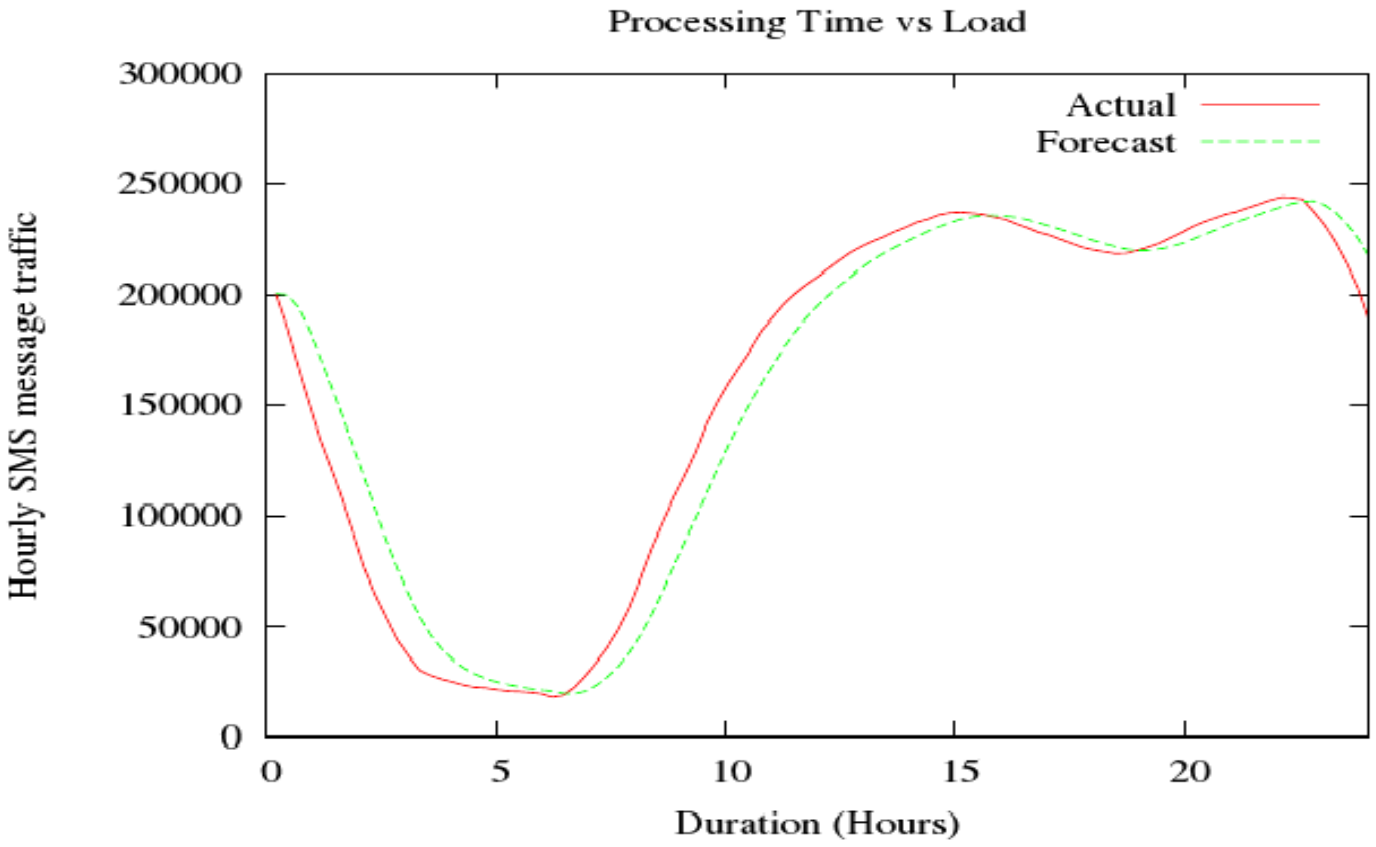
- Moving Average (MA)
- Exponential Smoothing
- Auto-Regressive Moving Average (ARMA)
- ARIMA (Integrated)
- ARFIMA (Fractional)

Source: P. A. Dinda and D.R. O' Hallaron: Host Load Prediction Using Linear Models, 2000

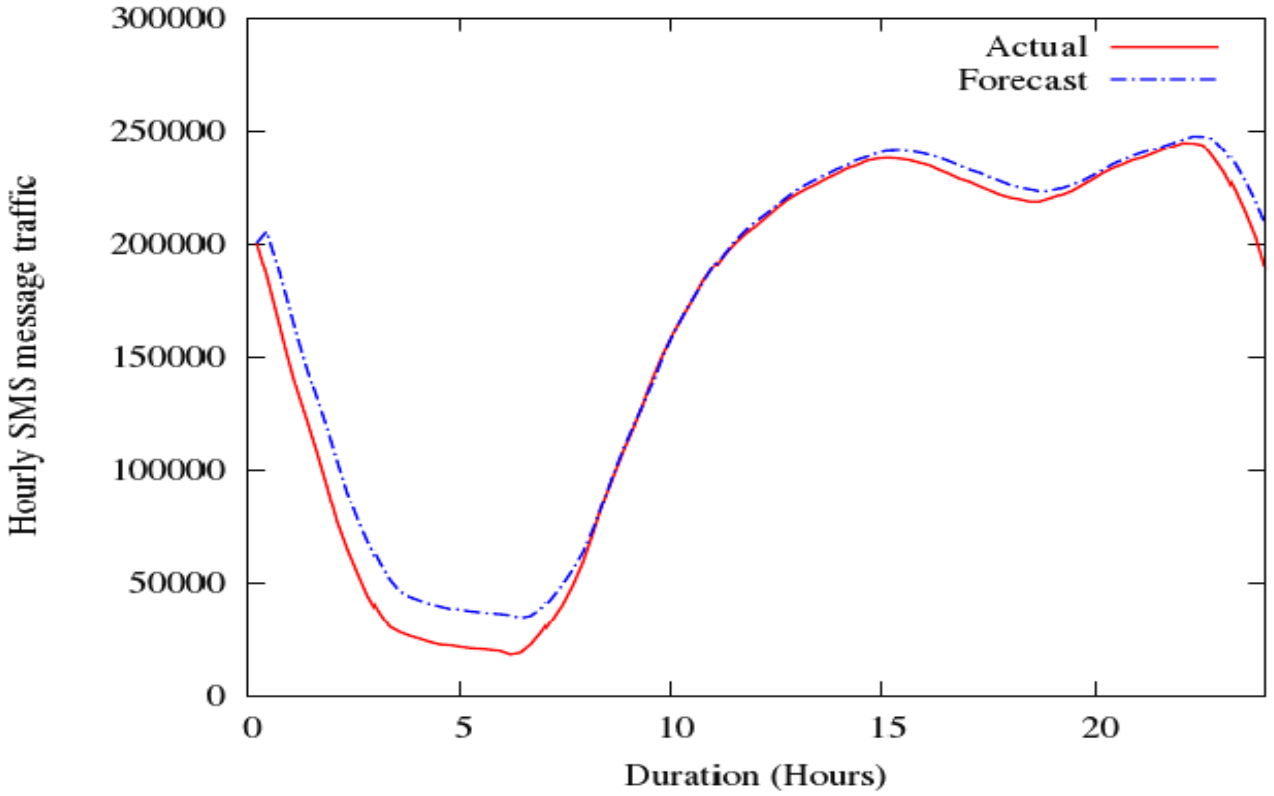
MA model: case SMSC



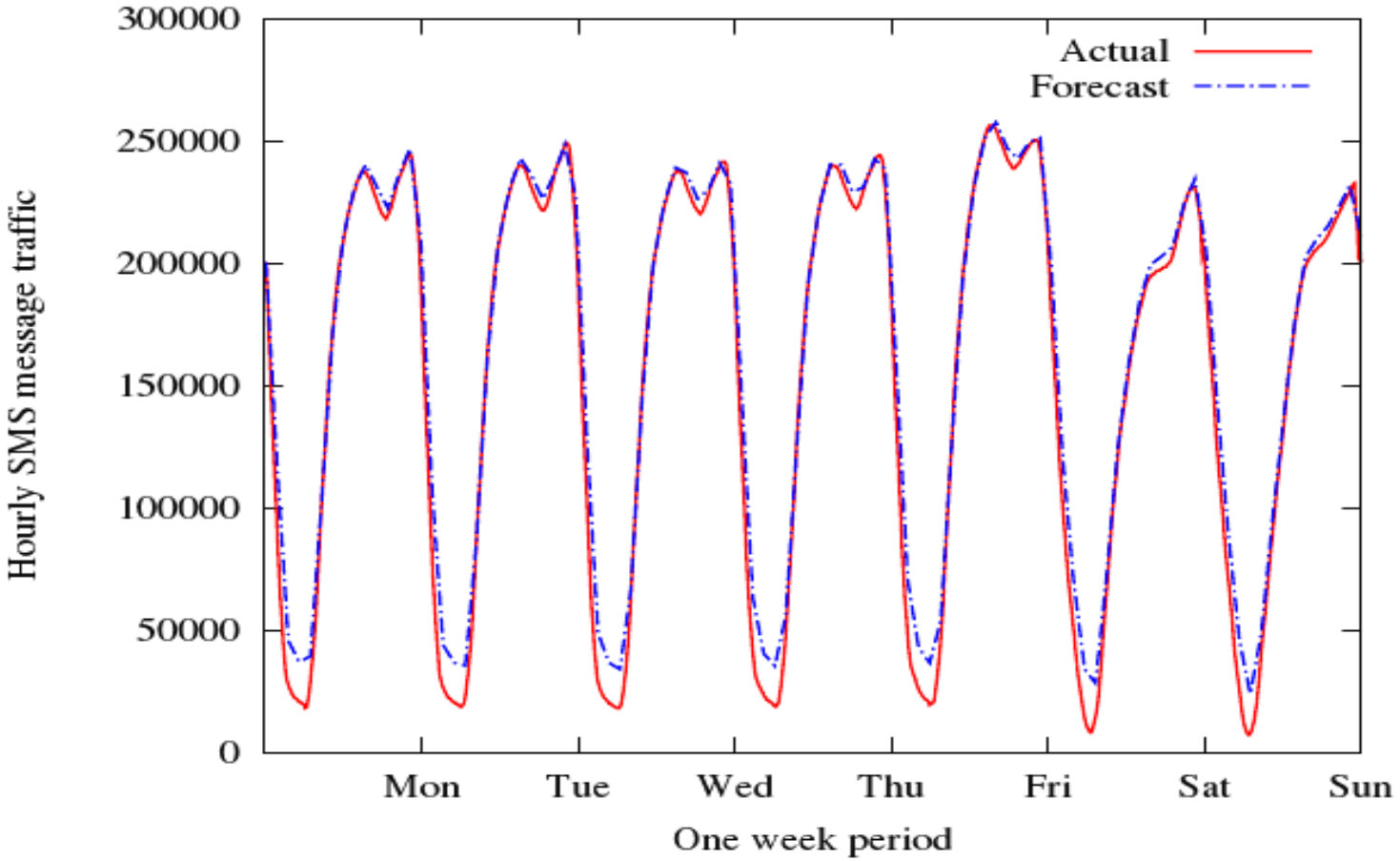
Exponential Smoothing: case SMSC



ARMA: Case SMSC - one day



ARMA: Case SMSC – one week



Conclusion and Future Work

- Conclusion
  - Reactive auto-scaling approach is not very feasible for QoS critical services
  - Unpredictable workload patterns and variable workloads can degrade the system performance
  - Workload modeling and predictive auto-scaling are imminent for latency sensitive applications
- Future Work
  - Explore alternative approaches and test the performance implications
  - Extend the approach to other use cases
  - Game theory: Nash Equilibrium (NE)
    - John Nash: See movie: A Beautiful Mind

Related research

1. T. Verleben, P. Simoens, F. De Turck and B. Dhoedt: Cloudlets: Bringing the Cloud to the Mobile User (MCS 2012)
2. J. C. Corbett et co: Spanner: Google's Globally-Distributed Database (OSDI 2012)
3. P. A. Dinda and D.R. O' Hallaron: Host Load Prediction Using Linear Models (Cluster Computing 3, 4, Oct 2000)
4. N. Roy, A. Dubey and A. Gokhale: Efficient Autoscaling in the Cloud using Predictive Models for Workload Forecasting (CLOUD 2011)
5. S. Venugopal, H. Li and P. Ray: Auto-scaling Emergency Call Centres using Cloud Resources to Handle Disasters (IWQoS 2011)
6. Reiss C, Tumanov A, Ganger GR, Katz RH, Kozuch MA: Towards understanding heterogeneous clouds at scale: Google trace analysis. 2012. (http://www.istc-cc.cmu.edu/publications/papers/2012/ISTC-CC-TR-12-101.pdf).
7. D. Ardagna, B. Panicucci and M. Passacantando: A Game Theoretic Formulation of the Service Provisioning Problem in Cloud Systems (WWW 2011)
8. R. Pal and P. Hui: On the Economics of Cloud Markets. CoRR 2011, abs/1103.0045.