Measuring and Characterizing the Performance of Multi-tier Cloud Applications

Ashiwan Sivakumar
Mohammad Hajjat, Shankaranarayanan P N, Sanjay Rao

LANMAN 2015
Motivation for our study

Interactive multi-tier applications are complex
- Multiple components with complex interactions
- Geo-distributed for high availability and low latency

E.g. Stocktrader – Components

- Config Service (CS)
- Data Base (DB)
- Front End (FE)
- Order Service (OS)
- Business Service (BS)
Motivation for our study

**Interactive multi-tier applications are complex**
- Multiple components with complex interactions
- Geo-distributed for high availability and low latency

**Require stringent SLA guarantees**
- Amazon: Every 100ms costs 1% in sales
- Google: 0.5 sec delay increase → traffic and revenue drop by 20%

**Cloud performance fluctuations**
- Can SLAs be met in the cloud?
Studies characterizing performance

Existing studies:
- Measure individual cloud services (E.g. EC2, Blob)
- Other classes of applications (E.g. High performance computing applications)

Our focus:
- Fine-grained per application component measurements of multi-tier apps
- Characterize performance issues experienced in the cloud
Complex transactions in geo-distributed multi-tier apps

E.g. Thumbnail, Stocktrader, ERP
Our Contributions

- Characterization of perf. in a geo-distributed setting
- Per-component measurements

Key findings:
- Replicas of a component are **uncorrelated** across DCs
- Attributed to a few app components at any time
- Performance issues are **short-lived**: 90% < 4 mins
- Choosing the best replica combination across DCs gives higher latency reduction
Outline

- Monitoring framework & Evaluation setup
- Characterization of poor performance
- Exploiting geo-distribution
- Conclusions
Monitoring framework

- **FE1**
  - Logging Server
  - Compute

- **BL1**
  - Logging Server
  - Compute

- **Queue service**
- **Cache service**
- **Collector Server**

X-Trace + aspects (AOP) for portability

Per-component measurement tools – “not general” (Dapper) and “not portable” (X-Trace)
Evaluation setup

- **Two cloud platforms** - Microsoft Azure, Amazon AWS
- **Four Applications**
  - Data-intensive: Thumbnail
  - Delay-sensitive: Stocktrader, Daytrader
  - Social: Twissandra
- **Real benchmark** workload (E.g. DaCapo)
- **Metric** – **server response time** (no internet delay)
Outline

- Monitoring framework & Evaluation setup
- Characterization of poor performance
- Exploiting geo-distribution
- Conclusions
Dissecting performance into constituent components

Long tail and Variation in all transactions
E.g. Simple login - 99.9%ile/median is 28

Few components show more variation
Analyzing bad performance episodes

IF avg. in a window > 95%ile – bad episode
Different subset responsible for bad performance at different times

StockTrader
Bad performance episodes are short-lived

90% last for 4 min
Bad performance episodes occur frequently

4-9 min of each other
Persistence of performance

- Auto-correlation function measure
  - Tendency for “server response time” to remain in the same state over time

Predictable over short time (<10 min)
Other result

- Performance of component replicas across DCs - Uncorrelated
Outline

- Monitoring framework & Evaluation setup
- Characterization of poor performance
- Exploiting geo-distribution
- Conclusions
Exploiting geo-distribution

Users

$FE_A \leftrightarrow BE_A \leftrightarrow BL_A \leftrightarrow DC_A$

$FE_B \leftrightarrow BE_B \leftrightarrow BL_B \leftrightarrow DC_B$
Cross DC path performs better sometimes
Best DC Vs. Best replica redirection strategies

- **Best DC**: Re-route entire request at the granularity of DCs
- **Best replica**: Select the best replica combination for each request
Best replica combination gives best results

Higher latency
Dealer: per-component request splitting

- **Dealer**: handle cloud variability in multi-tier interactive apps [CoNEXT 2012, JSAC 2013]
- **Per-component re-routing**: dynamically split user req’s across replicas in multiple DC’s at component granularity
- **Transient cloud variability**: performance problems in cloud services, workload spikes, failures, etc.
- **Performance tail improvement**:  
  - Natural cloud dynamics > 6x
Conclusions

- Presented a performance characterization of multi-tier apps in the cloud
  - To answer the question – Can SLA guarantees be met in the cloud?
- Applications experience short-term performance fluctuations frequently attributed to a few app components in a subset of DCs
- Choosing the best replica combination across DCs gives higher latency reduction than coarse-grained strategies
Q&A
Backup
Performance by transaction type - Stocktrader

Variation in all transactions
E.g. Simple login - 99.9%ile/median is 28

Long tail
## Correlation coefficients

<table>
<thead>
<tr>
<th></th>
<th>FE</th>
<th>DB</th>
<th>BS</th>
<th>OS</th>
<th>FE-BS</th>
<th>FE-CS</th>
<th>BS-CS</th>
<th>BS-OS</th>
<th>OS-CS</th>
</tr>
</thead>
<tbody>
<tr>
<td>FE</td>
<td>1</td>
<td>-0.08</td>
<td>-0.11</td>
<td>-0.04</td>
<td>-0.31</td>
<td>0.03</td>
<td>-0.32</td>
<td>-0.07</td>
<td>-0.04</td>
</tr>
<tr>
<td>DB</td>
<td>1</td>
<td>0.50</td>
<td>0.03</td>
<td>-0.01</td>
<td>-0.01</td>
<td>0.04</td>
<td>0.05</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>BS</td>
<td>1</td>
<td>0.14</td>
<td>0.08</td>
<td>-0.02</td>
<td>0.09</td>
<td>0.14</td>
<td>0.14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td>1</td>
<td>-0.37</td>
<td>-0.03</td>
<td>-0.40</td>
<td>0.66</td>
<td>0.74</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-BS</td>
<td>1</td>
<td>0.01</td>
<td>0.87</td>
<td>-0.31</td>
<td>-0.37</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FE-CS</td>
<td>1</td>
<td>-0.01</td>
<td>-0.02</td>
<td>-0.03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BS-CS</td>
<td>1</td>
<td>-0.34</td>
<td>-0.41</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BS-OS</td>
<td>1</td>
<td>0.71</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OS-CS</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(a) StockTrader