VQUERY: TRACKING CONFIGURATION + PERFORMANCE
Ilari Shafer, Charlene Zang, Greg Ganger (Carnegie Mellon University), Snorri Gylfason (VMware)

OVERVIEW

- Performance monitoring in IaaS datacenters
- Resources shared by many VMs
- Applications may span multiple VMs
- VM location/resources change dynamically
- vQuery: incorporate configuration monitoring
  - Track migration, resizing, etc. and performance effects
  - VMs as black boxes, monitored externally

- Configuration across different IaaS platforms
- Conceptually similar: entities, relationships
- Details vary: choice of configuration properties
- Examples: VMware vSphere, OpenStack, Tashi, ...

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>vSphere Configuration Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM</td>
<td>Host system, networks, datastore, name, annotation, memory, vCPUs, CPU allocation (reservation/limit/shares), virtual disk layout, guest OS type, guest OS state, IP address, ...</td>
</tr>
<tr>
<td>Host</td>
<td>Network, CPU (frequency, number of cores and packages), memory size, power state, ...</td>
</tr>
<tr>
<td>Network</td>
<td>Name, fence mode, parent, DNS (addresses, suffixes), netmask, IP ranges, ...</td>
</tr>
<tr>
<td>datastore</td>
<td>Name, capacity, free space, type, url, ...</td>
</tr>
</tbody>
</table>

- Configuration changes are frequent
- Some affect performance – many do not
- Primary goal: capture and mine these changes

CONFIGURATION TRACKING APPROACH

- Designed for diverse, historical, relational config data
- Collect configuration continuously, non-invasively
- Capture each change, not snapshots

- Collect configuration and performance data

- Fine-granularity, long-term storage for both
- Goal: integrate configuration in diagnosis, analysis, UI

OBSERVATIONS FROM INITIAL DEPLOYMENT

- Deployed on CMU vCloud, OpenCirrus (Tashi)
- Config changes co-occur: significant “major” events
  - Noisy, particularly due to some parameters
  - Example source of perf-affecting, root-cause changes
  - Migrations: manifest in perf. changes on VMs, hosts
  - Difficulty ahead: migrations “look” very different

CONTINUING WORK + CHALLENGES

- Find config changes that cause performance changes
- Filter raw configuration changes
- Identify significant performance changes
- Automatically correlate events + perf changes
- Provide intuition through visualization; mockup UI: