Building an autonomic CloudStack

Gabriel Beims Bräscher, Lucas Berri Cristofolini, and Rafael Weingärtner
January 19, 2017

Florianópolis, SC – Brazil
Current cloud environments

The cloud we want

How far are we from software like that?

Preliminary experiments
Current cloud environments
Poor management of computing resources
The cloud we want
How a cloud managed by software agents looks like
A bit of literature (Automatic Vs Autonomic)

**Automatic**

- Programmatic behavior
- Meant to be used in well known and closed environments
- It takes the form of a well defined script

**Autonomic**

- Can do everything that automatic systems do
- Has the ability to analyze a set of monitored data
- Can trace actions to be executed in order to fulfill goals
- An autonomic system is not tied to an implementation per se; it can adapt itself to the environment
Autonomous agents to manage and optimize the cloud
How far are we from software like that?
How far are we from software like that?

We are already there!
The source code is available at:

https://github.com/Autonomiccs/autonomiccs-platform
Agents basic execution flow diagram

Start -> loadManagement Heuristic -> searchClusterToBe Managed

sleep

found a cluster

no cluster found

rankHostsToShutdown

cannotShutdownHosts

shutdownIdleHosts

rankHostsToShutdown

cannotShutdownHosts

canShutdownHosts

mapAndMigrateVms ToHosts

rankHosts
Agents ability I (cloud consolidation)

Host 1 (On)

VM 1
VM 2

Host 2 (On)

VM 3

Powering off idle hosts

Host 1 (On)
 VM 1
VM 2

Host 2 (Off)
 VM 3
Agents ability II (cloud balance)

Balancing workload
Preliminary experiments
The experiment conducted

Description

- The proposal was added to Apache CloudStack
- used a consolidation heuristic
- adopted a host scoring approach based on more than one resource type
- the score is used during allocation and consolidation process
The cloud environment used

- **PV Intel**
  - Host 1 (On)
  - Host 2 (On)
  - vm6

- **HVM Intel A**
  - Host 3 (On)
  - Host 4 (On)
  - vm5
  - vm7
  - vm2
  - vm3
  - vm11
  - vm9
  - vm10

- **HVM Intel B**
  - Host 5 (On)

- **HVM AMD**
  - Host 6 (On)
  - Host 7 (On)
  - Host 8 (On)
  - vm4
  - vm8
  - vm1
The heuristic scoring approach

Mean of a cluster VMs resource allocation

\[ \mu_M \leftarrow \frac{\sum_{i=1}^{v} (memory_{VM_i})}{v} \] (1)

\[ \mu_{CpuFreq} \leftarrow \frac{\sum_{i=1}^{v} (cpuFrequency_{VM_i})}{v} \] (2)

\[ \mu_{cpuCores} \leftarrow \frac{\sum_{i=1}^{v} (cpuCores_{VM_i})}{v} \] (3)

Ranking of hosts for allocation and consolidation

\[ H_{score} = \left( \frac{H_m}{\mu M} \right) \times \left( \frac{H_{cpuFreq}}{\mu_{CpuFreq}} \right) \times \left( \frac{H_{cpuCores}}{\mu_{cpuCores}} \right) \] (4)
The cloud environment using Autonomicms
The deployment test

Deployed VMs

- 10 small virtual machines
- 5 large virtual machines

Table 1: Test VMs’ configuration.

<table>
<thead>
<tr>
<th>VM type</th>
<th>Amount</th>
<th>CPU</th>
<th>RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>10</td>
<td>1.0 Ghz</td>
<td>256 MB</td>
</tr>
<tr>
<td>L</td>
<td>5</td>
<td>1.8 Ghz</td>
<td>2.0 GB</td>
</tr>
</tbody>
</table>
The cloud environment after the deployment of test VMs
Figure 1: Deployment of VMs in our cloud environment without Autonomiccs framework (using logarithmic scale).
**Figure 2:** Deployment of VMs in our cloud environment with Autonomiccs (using logarithmic scale).
The results

Table 2: Average time to deploy VMs

<table>
<thead>
<tr>
<th></th>
<th>CloudStack</th>
<th>CS + Autonomiccs</th>
<th>Difference in %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total time to deploy 15 VMs</td>
<td>32 seconds</td>
<td>558 seconds</td>
<td>1,643.75 %</td>
</tr>
<tr>
<td>Total time excluding S5 VM</td>
<td>32 seconds</td>
<td>186 seconds</td>
<td>481.25 %</td>
</tr>
</tbody>
</table>
Benefits perceived

- The agent identified hosts 2, 4, 5, 6 and 7 as available to power off
- it reduced the number of running servers in 62.5%
- no need to rely on system administrators

Problems found

- The delay on servers activation if needed
- unexpected behavior found on Apache CloudStack monitoring module when it is used in cluster
Conclusion

In summary

- Autonomiccs framework enables the autonomic management of Apache CloudStack
- as shown before, despite some hiccups, Autonomiccs framework has potential
- the source code is open and available at: https://github.com/Autonomiccs/autonomiccs-platform
Let’s start a partnership?

We need help to push our startup further.

Let’s be partners!
Thank you all!
Any questions?

for further inquiries: rafael@apache.org