The Wisdom of Virtual Crowds: Mining Datacenter Telemetry to Collaboratively Debug Performance

Dragos Ionescu (dionescu@mit.edu), and Rean Griffith (rean@vmware.com)

Abstract

Explaining the (mis)behavior of virtual machines in large-scale cloud environments presents a number of challenges with respect to both scale and making sense of torrents of datacenter telemetry emanating from multiple levels of the stack. In this paper we leverage VM-similarity to explain the behavior or performance of a VM using its cohort as a reference (or by contrasting it against groups of VMs outside of its cohort). The key insight is that virtual machines (VMs) running the same application (components or workloads), or VMs colocated within the same (logical) tier of a complex application exhibit similar telemetry patterns.

The power of similarity relationships stems from the additional context that similarity provides. The quantitative or qualitative "distance" between a VM and its expected cohort could be used to explain or diagnose any discrepancy. Similarly, the distance between a VM and one in another cohort can be used to explain why the VMs are dissimilar. As an example we apply our data-mining techniques to debugging ViewPlanner performance. ViewPlanner is a tool used to emulate and evaluate large-scale deployments of virtual desktops. Using a ViewPlanner deployment of 175 VMs we collect ~ 300 metrics-per-VM, sampled at 20-second frequency over multiple 1 hour epochs, from the PerformanceManager [4] on ESX and automatically filter (using entropy measures [2]) and cluster them using K-means [1]. We use the median value of each metric within an epoch to summarize the VM's behavior during that epoch.

We introduce spread/diffusion metrics to explain the difference between VMs. Spread metrics are those such that the expected value of the order statistic (in our case the median) of a metric, m, $E[m]$ differs between two clusters, i.e., the expected value is conditioned on the cluster, $E[m_i|clusterA] \neq E[m_i|clusterB]$. Within a cluster of VMs, differences in the distribution of a particular metric, m_i, may be explained by conditioning m_i on other metrics, $\{c_0,\ldots,c_n\}$, where $E[m_i] \neq E[m_i|c_0,\ldots,c_n]$. We automatically find potentially interesting m_i's using Silverman's test [3] for multi-modality and we use Mutual Information [2] to find associated c_i's.

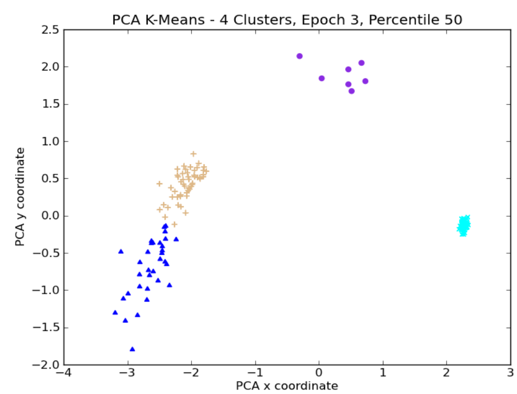

Figure 1: PCA projection of View Planner data arranged in four clusters by K-means. The 84 Desktop VMs (friendly-name prefix wexe-*) are distributed between two *adjacent* clusters (51 +'s and 33 $\triangle$'s). 84 Client VMs (friendly-name prefix rx1-*) are tightly-knit x's and (7) infrastructure VMs are o's.

$\frac{\lvert E[m\lvert A]-E[m\lvert B]\rvert}{m_{max}}$	**Metric Name**	$\frac{E[m\lvert A]}{m_{max}}$	$\frac{E[m\lvert B]}{m_{max}}$
0.453	cpu.ready.summation	0.291	0.744
0.443	cpu.latency.average	0.247	0.690
0.386	rescpu.actpk1.latest	0.324	0.711
0.375	rescpu.runav1.latest	0.335	0.710
0.372	cpu.usagemhz.average	0.312	0.684

Table 1: Top 5 spread/diffusion metrics separating cluster A (51 View Desktops) from cluster B (the remaining 33 View Desktops). Note that the largest differences are attributed to variations in CPU ready time and CPU (scheduling) latency.

SoCC'13, 1–3 Oct. 2013, Santa Clara, California, USA.
ACM 978-1-4503-2428-1.
http://dx.doi.org/10.1145/2523616.2525941

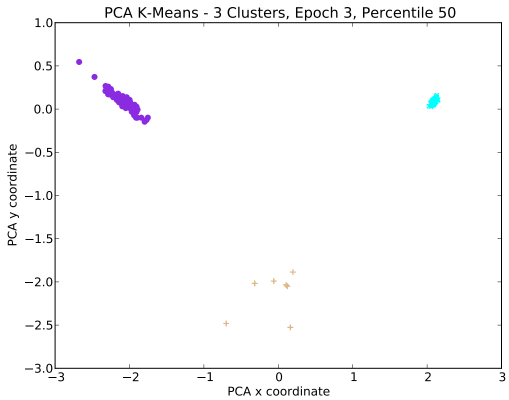

Figure 2: PCA projection of View Planner data in 3 clusters after removing the spread metrics. The 84 Desktop VMs are now in a tighter-knit cluster of o's. The 84 View Clients remain in a tight cluster of x's and the 7 infrastructure VMs are separated into their own cluster of +'s.

References

[1] T. Hastie, R. Tibshirani, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction: with 200 full-color illustrations*. New York: Springer-Verlag, 2001.

[2] A. Hyvarinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, Inc., 2001.

[3] B. W. Silverman. Using kernel density estimates to investigate multimodality. *J. R. Statist. Soc. B*, 43(1):97–99, 1981.

[4] VMware Inc. vsphere performance. `http://pubs.vmware.com/vsphere-50/index.jsp?topic=%2Fcom.vmware.wssdk.pg.doc_50%2FPPG_Ch16_Performance.18.1.html`.