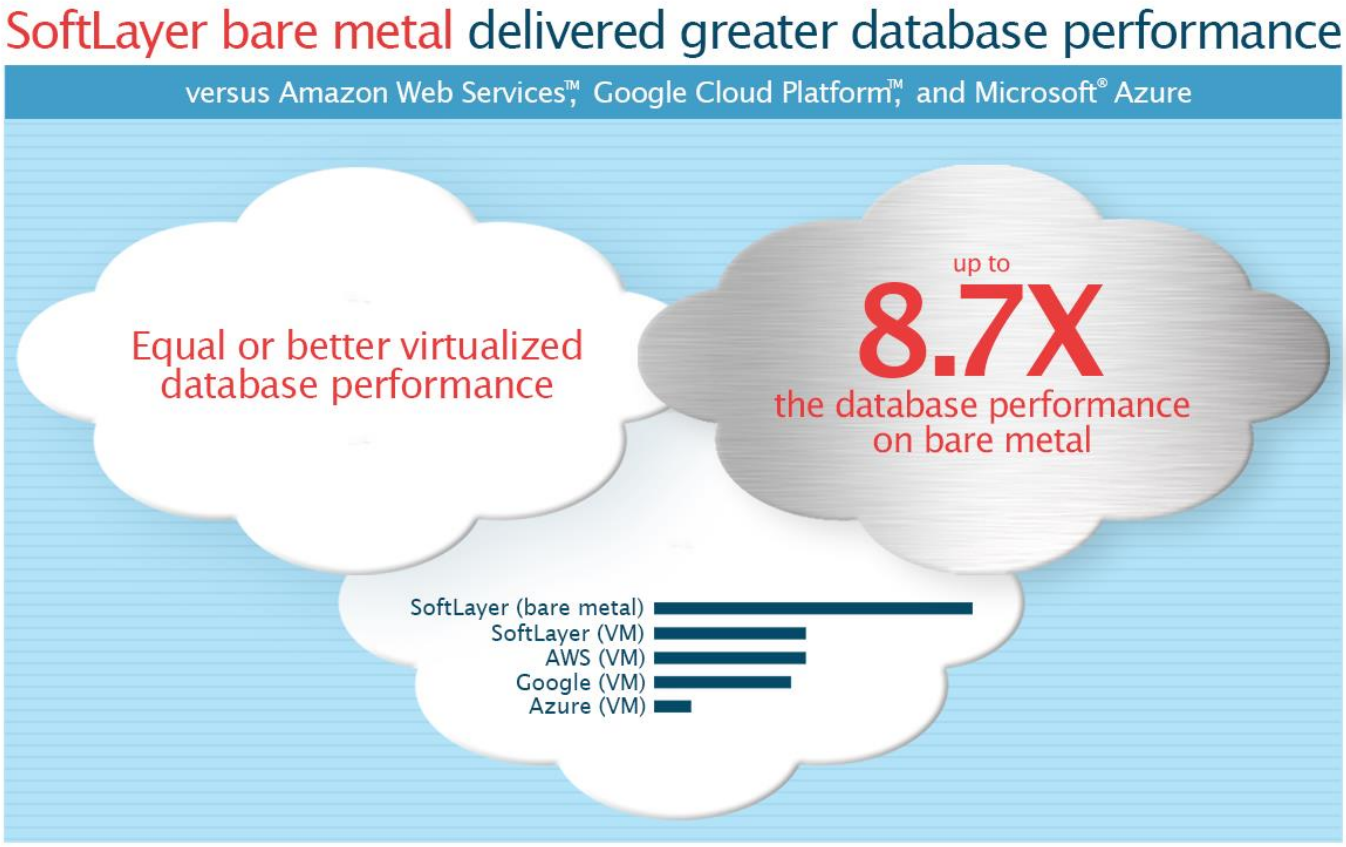
BETTER PUBLIC CLOUD PERFORMANCE WITH SOFTLAYER



The public cloud service provider you select to host your organization's applications can have a big impact on performance. Even when you choose similar resource allotments, the service each provider delivers can vary widely. Which provider can maximize your performance?

To find out, we set up accounts with four public cloud providers: SoftLayer, an IBM Company; Amazon Web Services (AWS); Google Cloud Platform; and Microsoft Azure. We used a database workload for testing, and using comparable database VMs, we found that SoftLayer, an IBM Company, essentially matched the performance of AWS, outperformed Google Cloud by 10 percent, and outperformed Azure by 313 percent.

An option that SoftLayer offers but the other three services do not is running workloads on physical machines (i.e., bare metal servers) instead of only on virtual machines, which can increase performance by eliminating resource contention. In our tests, SoftLayer's bare metal servers delivered more than twice the performance of the virtualized workloads on AWS and Google and yielded more than eight times the performance of Azure. This kind of dramatic boost can have a major impact on your business's important workloads.

MARCH 2015
A PRINCIPLED TECHNOLOGIES REPORT
Prepared for SoftLayer

ABOUT SOFTLAYER

Cloud computing can sound complex or vague; it makes many computing operations appear seamless to the user and lets you set up servers with ease. But each cloud offering and platform is made up of physical hardware located in data centers, and every cloud service provider is unique. The combination of the physical hardware, the virtual platform, and the way the cloud service provider offers these to customers can affect the performance that users may see.

According to the SoftLayer Web site[1], they provide "the highest performing cloud infrastructure available. One platform that takes data centers around the world that are full of the widest range of cloud computing options, and then integrates and automates everything."

"Our data centers and network share a single, proprietary management system. One tool that ties together and lets you control everything—every bare metal server, virtual server, storage device, you name it—in a single pane of glass. All accessible by API, portal, and mobile applications."

To learn more about SoftLayer, see www.softlayer.com.

THE POWER OF BARE METAL

Unlike some cloud providers, including those in our study—AWS, Azure, and Google—SoftLayer offers not only the option to host virtual machines on its servers, but the company also offers the option to run your applications on physical servers with no virtualization. This approach, known as bare metal because the applications run directly on the server, gives you the raw horsepower you need for processor-intensive and disk I/O-intensive workloads.

Running your workloads in VMs can make you vulnerable to noisy neighbor syndrome, where an extremely busy VM running "next door" to yours can draw resources from your workload and cause performance to decline. The bare metal approach eliminates this possibility because the entire server is yours, and you are in control. VMs also make your workloads susceptible to the hypervisor tax: the hypervisor uses processing power to manage resources between physical machine and VM, which means your workloads don't get the full performance possible. Virtual machines that use network storage can also suffer storage latency, and sometimes it's harder to see or configure the underlying hardware in a virtual environment. Running workloads on bare metal, which SoftLayer offers, can help you avoid these issues.

SoftLayer lets you configure its bare metal servers to exact specifications via its portal or API. You can choose from entry-level single-processor servers to quad-

[1] www.softlayer.com

processor, hex-core, and even GPU-powered servers. You are able to fully customize your bare metal server with RAM, SSD hard drives, network uplinks, and more. These capabilities are available on demand. According to SoftLayer, you can order a standard-configuration hourly bare metal server and have it online in 20 to 30 minutes.

To learn more about SoftLayer's bare metal server offerings, see www.softlayer.com/bare-metal-servers.

OUR TESTING

We will present the results of our testing below, but first let's take a look at how the workload, and many real-world applications, function. Each workload instance in our tests involved two components—a front-end application server and a backend database server. As in real-world environments, the front-end application server is what the users may see and interact with, such as an online store or catalog. This front-end application then traverses a network to the backend database that serves up the data in response to application requests. Typically, in a cloud environment, these two components reside in their own VMs, as the first row in Figure 1 illustrates. This is the model we first used for our virtualized testing on the four cloud services.

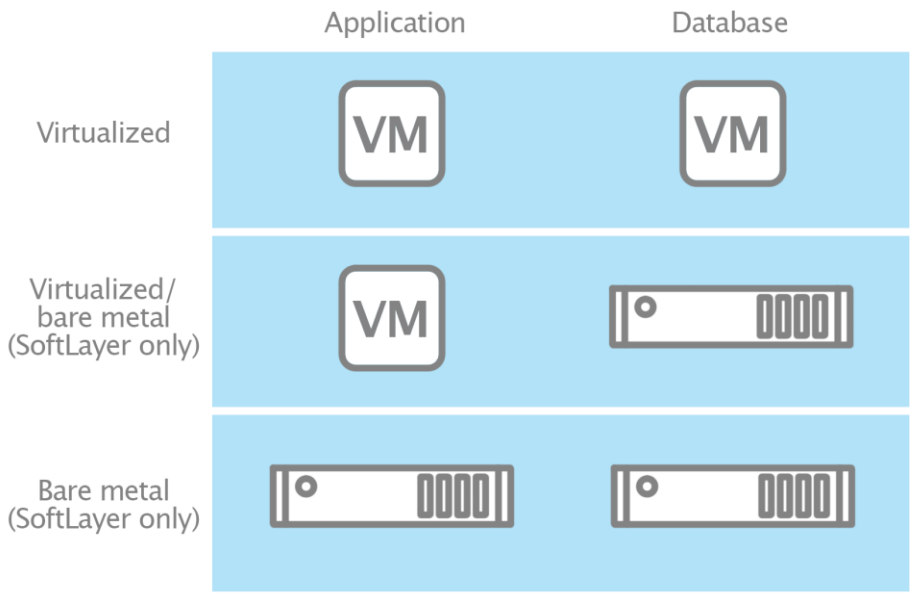


Figure 1: A visual summary of the virtualized and bare metal testing we conducted.

To compare the performance of SoftLayer to that of three other cloud services providers, Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform, we began by using each provider to run a virtualized database workload. We chose to test with database workloads because they utilize a good mix of CPU and I/O, which can be an indicator of general performance. We subscribed to the four services and then set

up configurations that were as comparable as possible. All of the configurations used eight vCPUs; the memory ranged from 28GB to 32GB. We used DVD Store, which is a benchmarking tool that measures database performance, to measure the number of orders per minute each solution delivered. After testing the virtualized servers, we explored the bare metal option available with SoftLayer.

Virtualized database performance

Figures 2 and 3 show the median number of orders per minute the four services achieved in our front- and backend virtualized testing. The results for SoftLayer and Amazon Web Services were almost identical—the median runs differed by only one-tenth of a percent. These services outperformed Google by 10 percent and Azure by a whopping 313 percent.

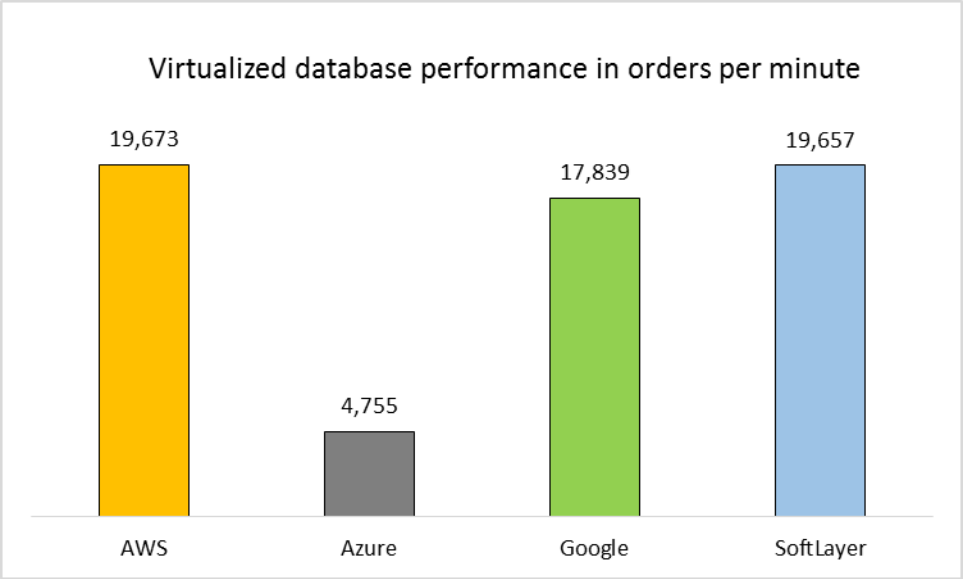


Figure 2: Virtualized database performance with SoftLayer essentially matched that of AWS and exceeded that of the other solutions.

	Run 1	Run 2	Run 3	Median
AWS	19,673	19,966	19,067	19,673
Azure	4,755	4,770	3,982	4,755
Google	17,857	17,839	17,418	17,839
SoftLayer application VM/database VM	19,657	19,469	19,697	19,657

Figure 3: Complete test results.

Moving to bare metal with SoftLayer boosts database performance

As we mentioned, SoftLayer provides the option of running all or part of the workload on bare metal servers. After we tested the front and backend virtualized solutions, we wanted to learn how performance would change if we shifted our workload from VMs to SoftLayer bare metal servers on either end. First, we kept the

application server running in a SoftLayer VM but moved the database workload to a bare metal server (the middle row in Figure 1). Next, we moved both to bare metal (the bottom row in Figure 1).

Figures 4 and 5 show the results when we shifted first the database workload and then both the application and database workloads from a VM to bare metal. As the middle bar in Figure 4 shows, running the application server on a VM and the database on bare metal yielded 27,633 OPM, an increase of 40.6 percent over the all-virtualized SoftLayer solution. The right-most bar shows the even more dramatic performance improvement we saw when we moved both the application and the database to SoftLayer's bare metal servers—more than double the performance of the VM-only solution.

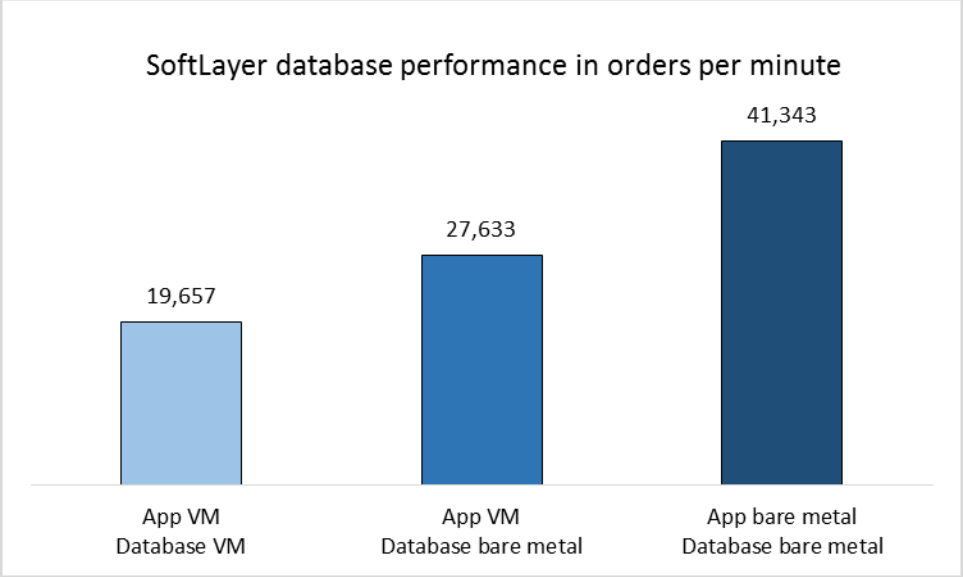


Figure 4: Database performance improved dramatically when we moved components of the SoftLayer solution from VMs to bare-metal servers.

	Run 1	Run 2	Run 3	Median
SoftLayer application VM/database VM	19,657	19,469	19,697	19,657
SoftLayer application VM/database bare metal	27,652	27,633	27,413	27,633
SoftLayer application bare metal/database bare metal	40,987	41,460	41,343	41,343

Figure 5: Complete test results.

Performance per cost per hour

The money that your organization has to spend for cloud computing is also an important consideration. Is the level of performance you get worth it? We compared the cloud solutions we tested in price per hour and found that SoftLayer offered the

best performance for the price per hour of the configurations we tested.[2] (See Figures 6 and 7.) The SoftLayer bare metal solution offered 32.5 percent more performance per cost than AWS, 440.4 percent more than Azure, and 6.8 percent more than Google.

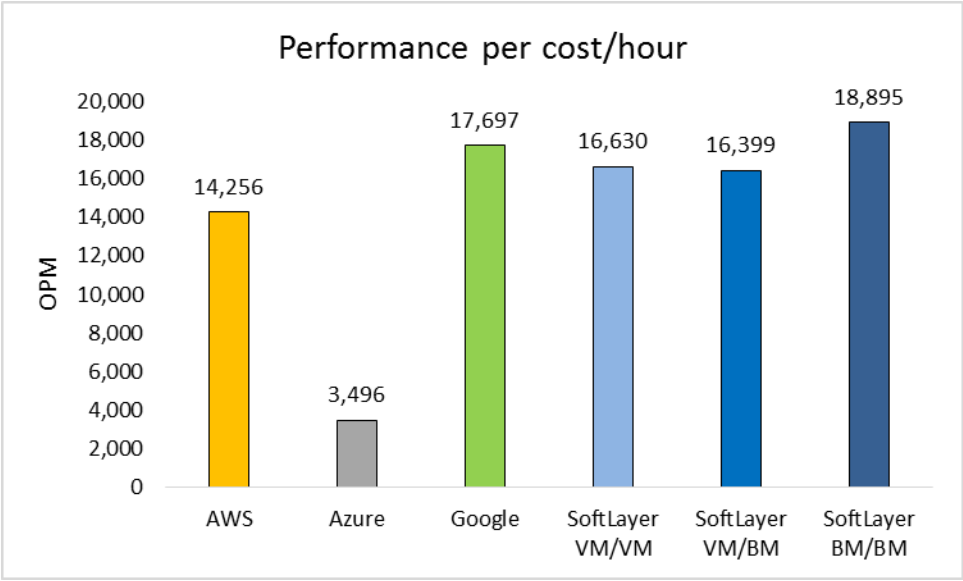


Figure 6: The performance each solution delivered compared to its cost per hour. Higher numbers are better.

	Application	Database	Total	Performance per cost/hr
AWS	0.690	0.690	1.380	14,256
Azure	0.680	0.680	1.360	3,496
Google	0.504	0.504	1.008	17,697
SoftLayer VM/VM	0.591	0.591	1.182	16,630
SoftLayer VM/BM	0.591	1.094	1.685	16,399
SoftLayer BM/BM	1.094	1.094	2.188	18,895

Figure 7: The SoftLayer bare metal solution delivered the best performance for cost. Costs are in USD.

IN CONCLUSION

In our virtualized database tests, we found that SoftLayer delivered performance comparable to that of AWS and outperformed Google by 10 percent and Azure by 313 percent.

When we shifted the application and database workloads from SoftLayer VMs to bare metal servers, available only from SoftLayer, performance more than doubled from that of the AWS and Google virtualized environments and was more than eight times that of Azure. These are important numbers to keep in mind as you strive to get maximum performance from your cloud service provider. When your business is

[2] Prices for AWS, Azure, and Google are from the monthly bills we received. SoftLayer pricing comes from the SoftLayer Web site. All prices were current as of 04/01/15 and do not include tax.

searching for a cloud solution, it is essential that you select the provider that can best handle your needs—whether your workloads can run comfortably in a virtualized environment or whether they require the power that comes with bare metal.

APPENDIX A – DETAILED TEST METHODOLOGY

For testing, we selected the default instances (see Figure 8). We configured similar instances with the same virtual processors and as close to same memory as possible. We used the same instance type for both Application and Database servers with one exception. On the database server, we added a 200GB attached disk to hold the database. In all cases, we used the fastest storage available. For AWS that was provisioned IOPs. Google was SSD Persistent Disk. Azure storage was abstracted from the end user and therefore unknown. To create the storage for Azure, we selected attach empty disk from the menu to add the additional disk.

Compute instance	Data center	Virtual CPU	Memory (GB)	Processor
AWS m3.2xlarge	us-east-1e	8	30	Intel Xeon E5-2670 v2 (2.50GHz)
Azure standard D4	East US	8	28	Intel Xeon E5-2660 (2.20GHz)
Google n1-standard-8	us-central1-a	8	30	Intel Xeon (2.60GHz)
SoftLayer (virtualized)	Dallas 9	8	32	Intel Xeon E5-2650 v2 (2.60GHz)
SoftLayer (bare metal)	Dallas 9	4 core, Hyper-threading enabled	32	Intel Xeon E3-1270 v3 (3.50GHz)

Figure 8: Test instance configurations.

We configured the instances using as close an OS comparison as possible. We used either Red Hat Enterprise Linux 6.5 or CentOS 6.4 with the latest updates depending on the available templates. In all cases, we used the available templates. For testing, we used kernel version 2.6.32-504.8.1.el6.x86_64.

We configured the application server with Apache. We performed the following commands to setup the server.

1. `yum groupinstall "Web Server" "PHP Support"`
2. `yum install http://yum.postgresql.org/9.4/redhat/rhel-6-x86_64/pgdg-redhat94-9.4-1.noarch.rpm`
3. `yum install postgresql94-libs`
4. `yum install php-pgsql`
5. Edit /var/www/html/dscommon.inc by changing the IP address of the Database server on the `$connstr` line.
6. Disable selinux

We used DVD Store 2.1 for testing. We created a 100GB database using the default DVD Store instructions for the creation. We installed the database on the attached 200GB disk. We used the following commands to set up the database server.

1. `yum install http://yum.postgresql.org/9.4/redhat/rhel-6-x86_64/pgdg-redhat94-9.4-1.noarch.rpm`
2. `yum install postgresql94-server postgresql94-contrib`
3. Make ds2 user:
   ```
   useradd ds2
   passwd ds2
   password = ds2
   ```
4. Edit /var/lib/pgsql/9.4/data/pg_hba.conf
   `host all all 0.0.0.0/0 trust (add this line to the bottom of file)`
5. Edit /var/lib/pgsql/9.4/data/postgresql.conf

```
listen_addresses = '*' (edit this line in file as shown)
```

6. Disable selinux

About our test tool, DVD Store Version 2.1

To create our real-world ecommerce workload, we used the DVD Store Version 2.1 benchmarking tool. DS2 models an online DVD store, where customers log in, search for movies, and make purchases. DS2 reports these actions in orders per minute (database requests) that the system could handle, to show what kind of performance you could expect for your customers. The DS2 workload includes other database requests, such as adding new customers, to exercise the wide range of database functions you would need to run your ecommerce environment.

For more details about the DS2 tool, see www.delltechcenter.com/page/DVD+Store.

For testing we performed three runs of DVD Store for 30 minutes each, restoring the database between each run. We took the median score. In between each run, we shut down the VMs and then powered them back on.

DVD Store has an executable that runs the test against the application and database server. It reports the orders per minute the server can handle. We ran the DVD Store executable on a separate Windows VM. We made sure the Windows VM was inside the same data center as the application and database server. The Windows VM had two virtual processors and 8GB of memory. We used a private internal network for all traffic between the servers. We created a Windows batch file with the following information in it and executed it for testing.

```
c:\DVD_Store\ds2webdriver.exe --target=ip address --ramp_rate=10 --run_time=30 --
n_threads=32 --db_size=100GB --think_time=0 --detailed_view=Y --warmup_time=1 --
pct_newcustomers=5 --csv_output=c:\dvd_store\client.csv
```

ABOUT PRINCIPLED TECHNOLOGIES

Principled Technologies, Inc.
1007 Slater Road, Suite 300
Durham, NC, 27703
www.principledtechnologies.com

We provide industry-leading technology assessment and fact-based marketing services. We bring to every assignment extensive experience with and expertise in all aspects of technology testing and analysis, from researching new technologies, to developing new methodologies, to testing with existing and new tools.

When the assessment is complete, we know how to present the results to a broad range of target audiences. We provide our clients with the materials they need, from market-focused data to use in their own collateral to custom sales aids, such as test reports, performance assessments, and white papers. Every document reflects the results of our trusted independent analysis.

We provide customized services that focus on our clients' individual requirements. Whether the technology involves hardware, software, Web sites, or services, we offer the experience, expertise, and tools to help our clients assess how it will fare against its competition, its performance, its market readiness, and its quality and reliability.

Our founders, Mark L. Van Name and Bill Catchings, have worked together in technology assessment for over 20 years. As journalists, they published over a thousand articles on a wide array of technology subjects. They created and led the Ziff-Davis Benchmark Operation, which developed such industry-standard benchmarks as Ziff Davis Media's Winstone and WebBench. They founded and led eTesting Labs, and after the acquisition of that company by Lionbridge Technologies were the head and CTO of VeriTest.