IBM

developerWorks.

Cloud computing service models, Part 1: Infrastructure as a Service

Dan Orlando
CEO
Creative RIA

25 January 2011

Learn about the key concepts of Infrastructure as a Service (IaaS). IaaS provides basic services, such as virtual servers, data storage, and databases into one platform for deploying and running your applications.
In this three-part series, straightforward, real-world examples of cloud computing help eliminate the confusion around the concepts. Each article in this series covers one of the three service models: Infrastructure as a Service, Platform as a Service, and Software as a Service. After reading this series, cloud computing will be more than simply a buzzword.

View more content in this series

Frequently used acronyms

- API: Application programming interface
- IT: Information technology
- ROI: Return on investment
- SLA: Service level agreement
- UI: User interface

In this article, learn about the first of the three classifications of the cloud: Infrastructure as a Service (IaaS). Some of the key concepts of IaaS include:

- Cloudbursting
- Multi-tenant computing
- Resource pooling
- The hypervisor

Most importantly, learn about the two primary facets that make IaaS special: elasticity and virtualization.

The value of IaaS

Develop skills on this topic

This content is part of progressive knowledge paths for advancing your skills. See:

- Cloud computing: Fundamentals
- Cloud computing: Introduction to Infrastructure as a Service

For businesses, the greatest value of IaaS is through a concept known as *cloudbursting*—the process of off-loading tasks to the cloud during times when the most compute resources are needed. The potential for capital savings through cloud bursting is significant, because businesses won't need to invest in additional servers that only run at 70% capacity two or three times in the year, the rest of the time sitting at 7-10% load.

However, for businesses to take advantage of IaaS in this capacity, IT departments must be able to build and implement the software that handles the ability to re-allocate processes to an IaaS cloud. There are four important considerations to building and implementing software that can manage such re-allocation processes:

- Developing for a specific vendor's proprietary IaaS could prove to be a costly mistake if the vendor were to go out of business.
- The complexity of well-written resource allocation software is significant and generally requires top-notch developer resources that do not come cheap. You'll save yourself and your organization a lot of time, frustration, and unanticipated expenses by budgeting more up front for the best resources you can find.
- What will you be sending off to be processed in the cloud? Sending data such as personal identities, financial information, and health care data put an organization's compliance at risk with U.S. Sarbanes-Oxley (SOX) Act, Payment Card Industry (PCI), or Health Insurance Portability and Accountability Act (HIPAA) regulations.
- Understand the dangers of shipping off processes that are critical to the day-to-day operation of the business. A good idea is to start by drawing a table and placing processes that involve compliance-critical data in one column, business-critical tasks in the second column, and non-critical tasks in the third column. Then, plan on having the software only off-load the items in the third column for its first iteration.

In addition, organizations need to be careful of the current state of the cloud computing marketplace in terms of vendor lock-in. Having virtual machines (VMs) that can be moved to the cloud from data centers and between vendors' clouds can be an asset for businesses, but doing so requires that vendors support a standardized file format, which they have been reluctant to do.

The reality of the situation is that currently there is no specification placed in the open and under the authority of a standards body. In other words, there currently is no truly standardized format, which complicates things at best, because there is no guarantee that the format around which you build will be supported by anyone down the road. It is worth noting, however, that it is often possible to port a virtual appliance to another format, provided that the specification of the new format is open or that you have access to it. On a more promising note, major advances have been made recently in support of the Open Virtualization Format (OVF), which is a promising candidate to become a standard. Another promising candidate is the Virtual Machine Disk (VMDK) format. VMDK was originally a proprietary format for VMware, but now that the specification is open, it is supported by a number of third parties.

Infrastructure as an asset

To illustrate the evolution of cloud computing, consider how the automobile industry has evolved over the course of the past five decades. Competitive advantage for auto manufacturers was most often won by the amount of sheer horsepower and torque that could be squeezed out of the automobile through the 1960s and 1970s. In the 1980s, however, this paradigm proved unfavorable for the marketplace and the environment, which forced a paradigm shift from infrastructure as an asset to Infrastructure as a Service.

Similarly, a vast majority of successful companies in the past 50 years have spent a massive amount of precious time and resources building infrastructures, with the goal of gaining competitive advantage by creating a bigger, faster, and stronger network than their competitors. The "infrastructure as an asset" paradigm in IT shares several of the same or similar inefficiencies and unfavorable characteristics that the muscle cars of the '60s and '70s had. With respect to enterprise computing, these inefficiencies include:

- Large tracts of unused compute power and capacity that carry costs associated with the large amount of space consumed by the hardware in large, expensive data centers.
- Expensive manpower requirements, including 24-hour monitoring by network administrators located in the data centers where the infrastructure assets (servers, routers, switches, and so on) are held.
- A massive barrier to the Green Computing initiative as a result of the high level of wasteful energy consumption.

To assist you in understanding the three classifications of cloud computing, I created a cross-concept matrix for your reference (see Table 1). A *paradigm* is a model to which the majority of users conform. As mentioned a moment ago, IaaS marks the shift from the paradigm of infrastructure as an asset to that of Infrastructure as a Service. The other two classifications of cloud computing shown in Table 1 also mark a paradigm shift. For Platform as a Service (PaaS), the shift is from the paradigm of platform as an asset, where licenses are purchased in mass quantities. The same can be said for Software as a Service (SaaS), where the paradigm shift is from software being assets of the organization in the form of licenses to software being provided as a service. You will learn more about PaaS and SaaS in parts 2 and 3 of this series.

Table 1. Cross-concept matrix of the three classifications of cloud computing

	Paradigm shift	Characteristics	Key terms	Advantages	Disadvantages and risks	When not to use
IaaS	Infrastructure as an asset	Usually platform independent; infrastructure costs are shared and thus reduced; SLAs; pay by usage; self-scaling	Grid computing, utility computing, compute instance, hypervisor, cloudbursting, multi-tenant computing, resource pooling	Avoid capital expenditure on hardware and human resources; reduced ROI risk; low barriers to entry; streamlined and automated scaling	Business efficiency and productivity largely depends on the vendor's capabilities; potentially greater long-term cost; centralization requires new/different security measures	When capital budget is greater than operating budget
PaaS	License purchasing	Consumes cloud infrastructure; caters to agile project management methods	Solution stack	Streamlined version deployment	Centralization requires new/different security measures	N/A

SaaS	Software as an asset (business and consumer)	SLAs; UI powered by thin-client applications; cloud components; communication via APIs; stateless; loosely coupled; modular; semantic interoperability	Thin client; client-server application	Avoid capital expenditure on software and development resources; reduced ROI risk; streamlined and iterative updates	Centralization of data requires new/ different security measures	N/A

Primary facets of IaaS

Rather than imagining the Internet as a single global cloud, it is perhaps more accurate to imagine it as a system of many clouds, like a thunderstorm. With this metaphor, it could be logically asserted that lightning is the weather system equivalent of communication among clouds. This metaphor is perhaps more accurate in the sense that clouds systematically interact with each other to create a single result: the Internet.

It is unlikely that the Internet will be made up of one single cloud—at least in the near future —because of the lack of standards in cloud computing and obvious attempts by companies to capitalize long term through vendor lock-in. Nevertheless, cloud computing would not have advanced to where it is currently if it weren't for innovation in the spirit of capitalism. Perhaps one day, the Internet really will be a single, interconnected cloud in which VMs could be transferred effortlessly to "the cloud" without concern for file format and interconnected clusters of VMs could be managed across service providers, all through a single interface. But that day is a long way off. In the meantime, we'll speak of the Internet as consisting of many clouds. (Ironically, I'm using the Apple MobileMe cloud to store this article so I can work on it on across several devices.)

Meet the elastic infrastructure

Elasticity is the first critical facet of IaaS. To illustrate the concept of elasticity, I'm going to require you to use your imagination for a moment. Pretend that clouds are actually made of marshmallow clusters stuck together so that people can sit and ride on them. Each marshmallow cloud can hold a certain number of people, depending on the number of marshmallow clusters that make up the cloud and how many marshmallows are contained in those clusters. As more people get on to ride the marshmallow cloud, you can expand the marshmallow clusters by sticking more marshmallows to them, increasing the surface area. As you have probably already figured out, the people represent the applications that require compute resources, such as those that host Web sites and run software services. The marshmallow clusters represent clusters of VMs, with each marshmallow a VM.

Although this might sound like something you'd expect to find in a Dr. Seuss book, it provides a means of understanding a concept considered by many a dark art: *elastic clustering.* Clustering of physical servers to form a virtual cloud is a concept known as *cloud clustering,* and if it is in fact a dark art, then mastery is measured by the scalability of an artist's system design.

Let's look at an example. Say that you're a statistical researcher working for the U.S. government. The government is a bit short-handed, and you've just been tasked with compiling all the data from the latest U.S. census. You're responsible for formulating the necessary statistical data so that

Congress can make important decisions regarding the allocation of economic recovery funds and tax dollars three days from now. Needless to say, this is a pretty important job, and you're on a bit of a time crunch. What's more, the amount of data you must process is astronomical, and you just found out that the compute resources required to compile it is going to take the IT department three weeks to get ready!

This is exactly the kind of problem that you can easily mitigate using IaaS. As a matter of fact, using IaaS, you could have the entire U.S. census data analysis completed within an hour. You'd start by creating a single instance of a server that contains the database software to run queries on the data. This is called an *image.*

After you deploy the image and import the data into the database, you could then duplicate that image as many times as necessary and start running your data-processing tasks. While the tasks are running, you might manually or automatically add and remove resources. For example, if the compute tasks were not running quickly enough, simply add more duplicate machine instances to the cluster.

Now that you understand the concept of elasticity, let's take a look at the second major facet of IaaS: virtualization.

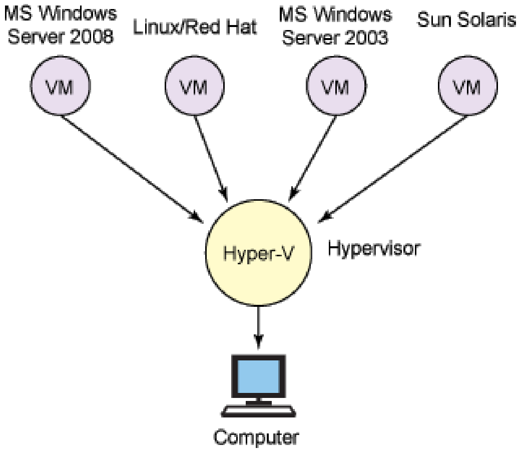
Machine virtualization

Sergey Brin and Larry Page, the founders of Google, had the right idea back in 1995 when they spent their evenings sifting through dumpsters behind Stanford University's computer science building, pulling out disregarded computer parts. They'd bring these random x86-based computer parts back to their dorm room to add to the Frankenstein machine hosting the legendary rogue Web crawler that took down Stanford's entire network—twice.

Today, it is estimated that Google has more than 1 million x86 servers in 12 major data centers and about 20 smaller centers on different continents. That's a pretty big cloud. Two key factors to the system design allowed them to scale the dorm-room beast in 1995, and it still holds true for the million-plus servers in the Google network today. To this day, Google continues to use *inexpensive x86 parts* instead of the much more expensive enterprise server components found in many corporate data centers. Second, failover, redundancy, monitoring, clustering, and other infrastructure management tasks are handled by a virtualization system that runs beneath the operating system level rather than using separate hardware such as load balancers to handle such tasks.

IaaS is easy to spot, because it is typically platform-independent. IaaS consists of a combination of hardware and software resources. IaaS software is low-level code that runs independent of an operating system—called a *hypervisor*—and is responsible for taking inventory of hardware resources and allocating said resources based on demand (see Figure 1). This process is referred to as *resource pooling.* Resource pooling by the hypervisor makes virtualization possible, and virtualization makes *multi-tenant computing* possible—a concept that refers to an infrastructure shared by several organizations with similar interests in regard to security requirements and compliance considerations.

Figure 1. The relationship among VMs, the hypervisor, and the computer



With IaaS, you have the capability to provision processing, storage, networks, and other computing resources, where you can deploy and run arbitrary software such as operating systems and applications. Most use cases for cloud computing follow the same fundamental layering structure you are already used to: a software solution stack or platform is deployed on a network infrastructure, and applications are run on top of the platform. However, virtualization makes the cloud paradigm unique.

Conclusion

In this article, you learned about many of the basic principles of cloud computing as well as the anatomy of IaaS and how it might be used in a real-world situation. The second article in this series will dive into the second major classification of cloud computing: PaaS. In the meantime, check out the Resources section for links to more information on IaaS.

Resources

Learn

- Grace Walker's developerWorks article "Cloud computing fundamentals" provides a good introduction to cloud computing.
- Check out the Cloudscaling's IaaS Buyer's Guide.
- Wikipedia provides good background on cloud computing.
- Check out Eweek.com's survey on companies using IaaS.
- Many companies are providing IaaS today. Two major players include:
  - IBM Smart Business Development and Test on the IBM Cloud
  - AWS (in particular, Amazon Elastic Compute Cloud [Amazon EC2] and Amazon CloudFront)
- Explore developerWorks Cloud computing zone, where you will find valuable community discussions and learn about new technical resources related to the cloud.
- In IBM Smart Business Cloud Computing, get valuable business advise to enhance performance and efficiency in the cloud.
- Read Cloud Computing—A Primer for a basic understanding of cloud computing.
- Follow developerWorks on Twitter.
- Watch developerWorks on-demand demos ranging from product installation and setup demos for beginners to advanced functionality for experienced developers.

Get products and technologies

- ElasticFox is a Firefox extension that allows your to manage your Amazon EC2 resources.
- See the product images available on the IBM Smart Business Development and Test on the IBM Cloud.
- See the product images available on Amazon Elastic Compute Cloud.

Discuss

- Get involved in the developerWorks community. Connect with other developerWorks users while exploring the developer-driven blogs, forums, groups, and wikis.

About the author

Dan Orlando

Dan Orlando is a recognized leader in the enterprise development community. As a long-time consultant, Dan's expertise on Adobe technology platforms is often called upon by industry leaders as well as publications such as IBM developerWorks, Adobe Developer Connection, and Amazon Web Services. Dan can also be found blogging regularly at DanOrlando.com.