Software

DevOps for the Cloud

Achieving agility throughout the application lifecycle

We don't have to tell you that your company is under increasing pressure to respond more quickly to changing business conditions. IT professionals like you, in turn, are under pressure to deliver applications to support that quick response. As you well know, the common term for this responsiveness is agility, and its underlying infrastructure is cloud computing.

While you may treat cloud computing as an exciting technology development, business leaders have far higher expectations for it: they expect nothing less than being able to improve overall business agility by leveraging cloud computing to bring new offerings to market more quickly.

The shift to automated processes operating on a cloud infrastructure requires changes to how applications are provisioned and operated throughout their lifecycle. Developing an integrated strategy to achieve lifecycle agility has given rise to a new operating paradigm commonly referred to as DevOps.

This white paper will examine the shift to DevOps, and why and how integrated cloud management as part of an overall DevOps strategy enables true application lifecycle agility.

The business imperative of agility

The financial crisis that started in 2008 and the ongoing globalization of the world economy have forced change as never before. Companies are facing economic stresses significantly more challenging than previously experienced, resulting in an increasing need to respond – and respond more quickly – to rapidly changing business conditions.

Likewise, the information systems underpinning today's companies also have to be able to transform rapidly to support acquisition integration, new market offerings, joint ventures, and the like. This rapid transformation capability may be summed up in one word: agility.

The move to virtualization has helped improve agility by removing the need to provision hardware.

What is agility?

Agility refers to the ability of information systems to quickly and easily:

- Add application functionality to respond to customer demands
- Move quickly through application lifecycle stages: development, testing, pre-production, production
- Incorporate new functionality or integrate new user populations due to acquisition or mergers
- Change capacity to meet higher or lower demand

Unfortunately, the traditional manual methods of application and infrastructure management are poorly suited for agility and scale. Manual system administration is time-consuming and error-prone, both destructive of agility.

The move to virtualization has helped improve agility by removing the need to provision hardware. Users can request virtual machines, which are served up quickly. Removing the need to install hardware can reduce a portion of the total application provisioning time frame.

However, many organizations have implemented virtualization, only to find that the effect of its automation is limited, with little overall change to application agility. These organizations typically have carried over manual system administration techniques, which are slow and error-prone. So virtualization, while extremely valuable, does not, by itself, achieve the end-to-end business agility required in today's economic environment. Leaving established processes in place while implementing virtualization automates only part of the application lifecycle.

Consequently, the executives responsible for overall business direction find that virtualization is insufficient to achieve their true goal: improving overall business agility by bringing new offerings to market more quickly.

The rise of DevOps and the role of configuration management

In response to the need to quickly configure the software components on virtualized servers, a new discipline and set of tools has come onto the scene.

What is DevOps?

DevOps is a set of practices that is focused on using a new generation of tools to automate the configuration process for both system resources (e.g., the virtual machine operating system and middleware components) as well as application components (e.g., PHP website files). By capturing configuration information in scripts and executing them with an automated scheduler, implemented by a tool that enforces a defined workflow, DevOps solves the problems associated with the manual configuration process. The end result is that server deployment time shrinks from hours or days to seconds or minutes.

Of course, use of scripts to automate configuration management tasks is not new. Individual system administrators have always created scripts to implement commonly executed tasks. The drawback to this approach is that the scripts are limited in scope, highly idiosyncratic, and not shared. Oftentimes, when a system administrator leaves an organization, it is left with a non-working incomprehensible set of scripting files and must rebuild an automation capability from scratch.

By contrast, the new breed of tools brings vastly improved functionality to the configuration automation process and implements the promise of DevOps:

- Standardized tools ensure that all applications within an organization share a common approach to configuration automation.
- Standardized tools mean that the organization is not dependent upon a single individual to execute and maintain configuration automation.
- Tools are designed for more sophisticated use, ensuring they have controls over configuration mechanisms like order of

execution, timing of script execution, and parameter-based execution, which allows sophisticated configuration control flow.
- The extended functionality means the tools can be used throughout the application lifecycle process by development, test, and operations groups. Use of common server configuration tools enforces consistency in application operation.
- Because the tools are widely used across the entire technology industry, their functionality is vastly more capable than scripts created and maintained by a single person.
- Because they are publicly available, configuration scripts can be shared, allowing open source-like community involvement to improve capability across the industry.

Because of these benefits, every organization should be looking at configuration management (aka IT automation) tools. Their use in server configuration can transform an inefficient, mistake-prone process into a streamlined process that ensures speed, consistency, and quality in a critical application management task.

The role of cloud management

Cloud computing is often the driving factor for organizations that implement configuration management. Because cloud computing environments are dynamic, shared resources, it is quickly clear that manual system administration techniques are insufficient. The ability to implement automated, consistent server configuration without limit on the number of servers or how frequently new servers are instantiated is a perfect match for these dynamic environments.

However, it quickly also becomes clear that dynamic, shared environments require more than automated system configuration. Four key areas of functionality are crucial in a cloud environment and configuration management systems do not address these requirements:

Access controls

In a shared environment, it is critical to integrate cloud identity management with the corporate identity management system. If integration is not present, employees who leave the organization may not be removed from the identity management system used by cloud applications, posing the security issue of inappropriate access.

Governance

Associated with access controls is the need to manage resource actions correctly. Without a robust governance capability in place, it is impossible to control user rights with cloud functionality. For example, for cloud applications in development, the organization may have a policy of allowing software engineers to launch and terminate virtual machines, while for production systems only operations personnel may have that right.

Billing

In a shared resource environment, resource costs must be assigned appropriately. Direct chargeback billing to user organizations for consumed resources is crucial to ensure responsible use. In addition, it is also important to track overall costs and measure them against allowed budget, thereby ensuring that user organizations stay within their approved spend authorization levels.

Application management

Application management refers to the initial launch of application virtual machines, their ongoing monitoring, and adjustment of the application to respond to resource failure or user load via failover or auto-scaling. Failover refers to the launching new resources (e.g., virtual machines) in response to the failure of an application resource.

The term auto-scaling refers to the ability of an application to automatically add or remove computing resources in response to changing conditions. A common auto-scaling decision criterion is processor load – when the servers within an application tier experience high processor loads, auto-scaling will add additional servers to the tier.

Implementing automated, consistent server configuration without limit on the number of servers is a perfect match for dynamic environments.

The best practice for an integrated DevOps environment is to use stripped-down templates, containing only the necessary software needed to create a virtual machine.

Likewise, when the tier's servers' processor load drops below a given level, the application terminates virtual machines to bring the number of servers to an appropriate level. Auto-scaling enables organizations to always have sufficient compute resources available while avoiding extra costs associated with low-load virtual machines.

Cloud management systems combine policy, control, and monitoring to ensure that IT organizations operate their cloud environments efficiently and in compliance with organizational practices and processes. They are a critical capability as companies look toward a future of autonomic applications, hybrid cloud deployments, and heterogeneous cloud environments.

Marrying configuration and cloud management

A question naturally arises: how should companies take advantage of both configuration and cloud management, and how should they be integrated with the underlying cloud software platforms that perform resource management – provisioning virtual machines, making network connections, and assigning storage?

While some organizations attempt to use either configuration or cloud management within their infrastructure environment, the most successful cloud users implement an environment that is comprised of three layers (see Figure 1 below).

Resource manager layer

As noted earlier in this white paper, many organizations begin using their cloud computing orchestration software by following a virtualization best practice – creating what are called virtual machine templates, which contain a full software payload consisting of operating system, middleware components, and application software components. They quickly discover that with the rapid pace of change in the many software components that are contained within a virtual machine that keeping templates up-to-date and consistent is a nightmare.

The best practice for an integrated DevOps environment is to use stripped-down templates, containing only the necessary software needed to create a virtual machine. This approach, often referred to as "JeOS" or "Just enough Operating System," frees IT organizations from constantly creating new templates that are difficult to track and quickly are obsolete and difficult to track.

In an integrated environment that uses both configuration and cloud

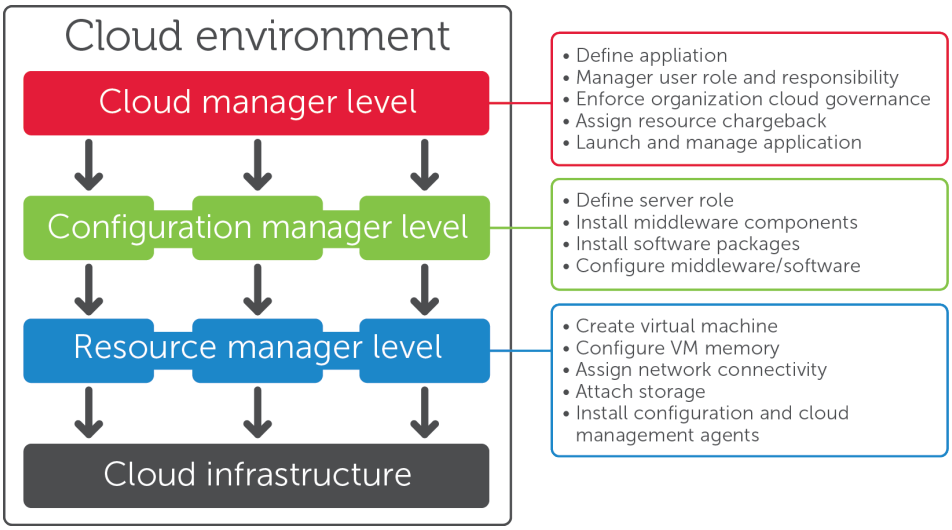

Figure 1. The three layers should be organized in this fashion

management systems, the JeOS methodology performs the following tasks:

- Create virtual machine
- Configure virtual machine memory
- Assign network connectivity
- Attach virtual machine storage
- Install configuration management agent
- Install cloud management agent

The same JeOS methodology is used whether the company has implemented a private cloud product (e.g., vCloud Director, OpenStack, or CloudStack) or is using a public cloud service provider (e.g., Amazon Web Services).

Configuration manager layer

This layer is where configuration management comes into play. Using a tool such as Opscode Chef, Puppet, or CFEngine, JeOS virtual machines are completed by having the configuration management tool of choice install appropriate software packages.

The best practice with respect to use of these tools is to view the responsibility of this layer as configuring individual servers, ensuring that every server used in a given application has exactly the right configuration of software packages and configuration to fulfill its role.

Resource Manager layer duties are carried out by the configuration management server communicating with the virtual machine configuration management agent. The server downloads the appropriate packages onto the virtual machine and the agent then kicks off installation and configuration for each package.

Typically, each virtual machine configuration definition is made up of numerous smaller scripts, which, when assembled, combine to create a virtual machine type. Constructing a virtual machine from a collection of scripts makes it easy to create a new virtual machine type by taking an existing virtual machine definition and modifying it by adding and/or removing individual scripts; the resulting script collection forms the definition of a new virtual machine type.

Using a number of individual scripts to define a virtual machine makes it easy to update an operational VM – the configuration management server connects to the VM agent and commands it to update one or more packages. This fine-grained control enables organizations to quickly roll out changes to individual software packages used by virtual machines without needing to update an entire template and reboot all affected virtual machines.

In summary, the role of the Configuration Manager layer is to ensure that virtual machines within an application are configured properly with all necessary software packages to fulfill their specific application role. The best practice to implement this layer is to use of the new breed of configuration management tools such as Chef or Puppet.

Cloud manager layer

The role of the Cloud Manager layer is to knit together all of an application's virtual machines into a coordinated whole, implementing user management, application governance, billing, and application management. Commonly referred to as cloud management systems, tools like Dell ™ take on responsibility for managing the collection of virtual machines that make up an application, ensuring that their operation provides high availability and compliance with applicable laws and company policies.

Cloud management tools typically control user access – who can access virtual machines and what actions individual users can perform on those VMs. In addition, cloud management tools monitor ongoing performance of all VMs that comprise an application to ensure the application meets necessary availability and user response times. If the cloud management tool identifies unacceptable metrics in these areas, it will raise an

Dell Cloud Manager™ manages the virtual machines that make up an application, ensuring that their operation provides high availability and compliance.

A new generation of tools offers automated, consistent system configuration and administration – this is the promise of cloud computing.

alert so that operations personnel can intervene and correct the problem.

Cloud management tools also translate application resource use into billable costs that will be charged back to application users. Using organization-specific application metrics (e.g., hours of virtual machine operation), cloud management tools assign costs to appropriate organizations, using the identity management functionality described above to assign costs to those organizations.

Finally, cloud management tools enforce governance for applications, allowing organizations to be confident that their applications conform to all necessary company policies and applicable laws and regulations. For example, company policy might be that only operations personnel can access running VMs in production applications, while a broader set of personnel can access running VMs in applications in the development stage.

The overall role of cloud management tools is to coordinate all of the resources that comprise an application and to ensure that they operate as a coherent entity. Typically, an application is defined at the cloud management level as a number of virtual machines, each of which performs a specific role. For each virtual machine, the cloud management tool will call the configuration management tool to install and configure all of the software packages necessary for a VM to perform its role.

Once an application is launched, the cloud management tool will monitor its ongoing performance and take appropriate actions to ensure proper application operation. This monitoring may take the form of notifying operations personnel of individual resource failure. It may also take the form of launching or terminating VMs, adjusting overall resource levels to ensure acceptable application performance.

Conclusion

The promise of DevOps is to operate a cloud environment in such a way that IT organizations can support today's business requirements for agility and speed. The traditional manual methods of system configuration and administration are slow and error-prone. A new generation of tools has come onto the scene that offer automated and consistent system configuration and administration – this is the promise of cloud computing.

Many IT organizations struggle to understand which of these tools should be adopted, and how they should be integrated to support DevOps across the complete application and throughout the entire application lifecycle.

The best DevOps approach is to recognize the appropriate role and advantages of the new tools and to ensure that they are used to support cloud application DevOps. In order to achieve cloud application DevOps, three layers of functionality are necessary (see Figure 2).

Resource manager layer

Provision necessary resources for applications. These are fundamental resources like virtual machines with associated storage and network connections. Products such as vSphere, OpenStack, or CloudStack, typically provide Resource Manager functionality, and are commonly referred to as cloud software.

Configuration manager layer

Configure those resources to fulfill their role within an application. Configuration Manager products install and configure software packages to enable individual VMs to perform their role within the overall application. Products used to provide Configuration Manager functionality, typically called configuration management products, include Chef, Puppet, and CFEngine.

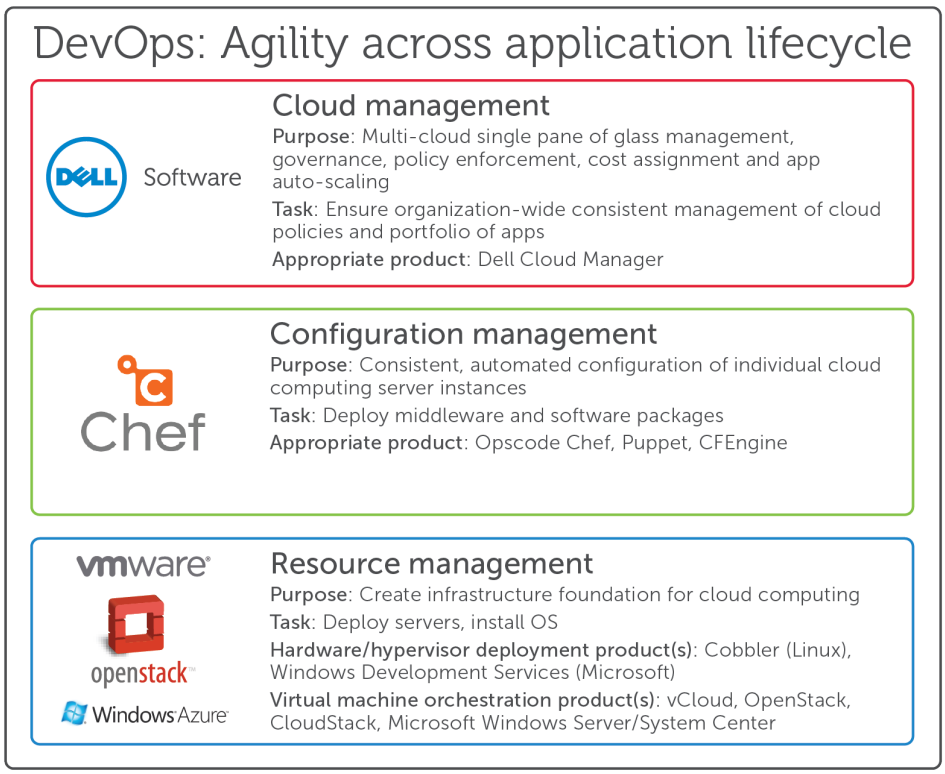

Figure 2. Illustrates the layers and related recommended products

Cloud manager layer
Coordinate the collection of resources to ensure functionality and performance of the entire application. Cloud Manager products ensure that each VM is operating properly and that sufficient resources are available to meet application demand. Cloud Manager products also control user access to application resources and assign costs for those resources to appropriate user organizations. The most common term for Cloud Manager products is cloud management software, like Cloud Manager.

Each of the three layers performs a vital role, and organizations that aspire to the speed and agility of DevOps must implement each layer. However, the layers must be integrated and coordinated to achieve agility throughout the application lifecycle. Cloud Manager supports major cloud software products and cloud service providers, and can integrate with all leading configuration management software. Implementing and integrating Cloud Manager with an appropriate configuration management product and the organization's chosen cloud environment enables IT organizations to achieve the promise of DevOps.

About Cloud Manager

The Enterprise Cloud Management Solution
Cloud Manager is a cloud infrastructure management solution for deploying and managing enterprise-class applications in public, private and hybrid clouds. The Cloud Manager multi-cloud architecture provides enterprises around the world with agility, governance, and choice.

Agility–Easily deploy and manage cloud applications across public and private clouds. Developers can leverage self-service provisioning, deploy applications to the cloud, and automate scaling and cloud-bursting based on system or application triggers.

Dell Cloud Manager supports major cloud software products and cloud service providers, and integrates with leading configuration management software.

Governance–Maintain control over cloud operations through integration with identity management systems, fine-grained access controls, financial tracking, and logging/monitoring of all actions taken within your cloud environment.

Choice–Cloud Manager supports more than 20 leading public and provide cloud platforms, plus leading configuration management tools, and integrates into your existing operations, such as billing and monitoring. Cloud Manager can be delivered as Software as a Service (SaaS), or deployed on-premises.

For More Information

About Dell Software

Dell Software helps customers unlock greater potential through the power of technology—delivering scalable, affordable and simple-to-use solutions that simplify IT and mitigate risk. The Dell Software portfolio addresses five key areas of customer needs: data center and cloud management, information management, mobile workforce management, security and data protection. This software, when combined with Dell hardware and services, drives unmatched efficiency and productivity to accelerate business results. www.dellsoftware.com.

If you have any questions regarding your potential use of this material, contact:

Dell Software
5 Polaris Way
Aliso Viejo, CA 92656
www.dellsoftware.com
Refer to our Web site for regional and international office information.