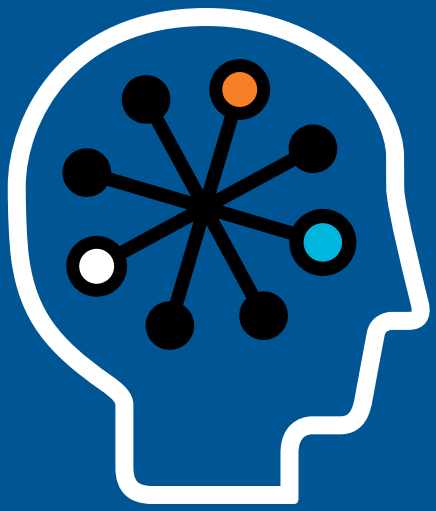

HP Data Center Management Architecture

Adaptive Infrastructure Supply Chain

Introduction

Infrastructure virtualization has provided significant benefits to IT – allowing resource utilization to increase significantly and reducing the cost of many time-consuming steps in the deployment lifecycle. This white paper introduces the next step beyond infrastructure virtualization: Infrastructure provided as a service inventory, allocated on order as in a supply chain.

While virtualization allowed IT to move from deploying application infrastructure in a fully-custom way (also called "design-to-order") to a more nimble way (also called "configure-to-order"), an infrastructure supply chain allows IT to bind a pre-provisioned resource to an application instantly (also called "allocate-to-order").

Fundamental to this shift is the desire to offer resource services from a supplier to a self-service customer on-demand, using the concept of a service inventory – with example suppliers being network services or SAN administration, and customers being application and server provisioning teams. The service inventory is built using a forecasting/planning process. Infrastructure "standards" within an IT shop are critical to making this work, as the service inventory provides only standard resources – for example "small J2EE application server" or "standard VMFS LUN" or "redundant DMZ connection." These standards can be layered in such a way that IT resembles a supply chain of processes to deliver standard infrastructure components.

The benefits of this approach are manifold:

- lower mix, higher volume of distinct processes
- better capacity management
- better quality
- less complexity
- faster TTM for applications
- lower cost

Clearly, these potential benefits will require significant governance changes in IT to bring about. This white paper provides a discussion of how to view IT infrastructure as a supply chain, and what the high-level complications and implications are.

The Old Model – Design-to-Order

Data center infrastructure has typically been custom-integrated and dedicated piece-by-piece to individual applications, based on a tops-down design for each application. No other sensible way to deploy applications has existed in the enterprise, for several reasons:

- Each application requires different platforms, with different OS revisions, patch levels, network topologies, security models, interfaces, performance bottlenecks, etc;
- Disruption of applications by other applications is too costly to justify sharing resources between them.
- The people operating the application are dedicated specialists, who need to isolate their infrastructure domain for the purpose of management and troubleshooting.

This approach to infrastructure deployment is called "design-to-order." The implication of design-to-order infrastructure is that every application commands dedicated infrastructure, maintenance process and lifecycle.

Some IT vendors have actually attempted to automate Design-to-Order deployment processes. Typically, the approach involves a custom template that describes the application's infrastructure requirements, coupled with a control system that, based on a comprehensive model of the entire data center, orchestrates the configuration changes required to constitute the custom, "virtual" servers associated with the application.

What are the problems with this approach? Since every change to the infrastructure is a novel, one-of-a-kind change, it is extremely difficult to automate "design-to-order" IT for several reasons:

- There's no repetition of any given change – difficult to build quality or to pre-test
- The impact of a custom change is unknown without comprehensive analysis. For example, if one mission-critical application could be affected by deploying another application on infrastructure they share, then automating that deployment involves unacceptable risk.
- Lacks determinism – whether or not the change is possible depends on the dynamic state of the entire shared system. If the change is not possible, it probably won't be known with desired lead time to remediate. Unwinding the change may be impossible in a dynamic environment, since all "known states" are novel.
- Forcing a lengthy, complex change review process tends to defeat the TTM benefit of automation in the first place.
- If the approved change process for a custom change requires a series of custom steps, then developing custom automation processes that will be used only once makes little sense.
- Multi-step automations can compound risk if early steps do unexpected things.

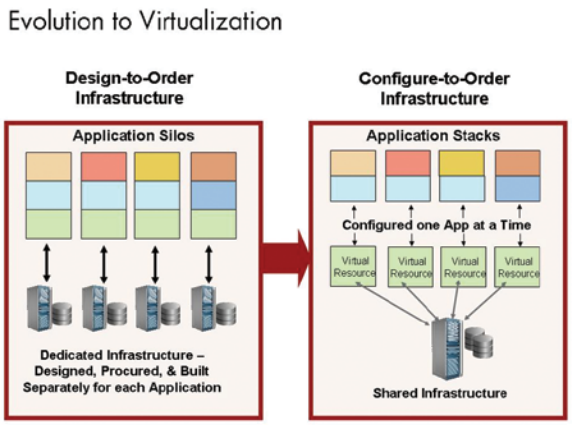

Figure 1. Moving from Design-to-Order to Configure-to-Order Infrastructure

The Virtualization-enabled Model – Configure-to-Order

Virtual machines have liberated server management in multiple ways:

- Allows servers to be shared by many applications, increasing hardware utilization
- Separates ownership of the hardware from ownership of the application, allowing management processes to be consolidated and optimized
- Allows scarce skilled staff to be centralized
- Allows new applications to be deployed without hardware procurement
- Enable shifting workloads between servers to remediate problems

With a virtualized server farm, adding a new workload typically only requires configuring an existing server to host the new application. These benefits have lowered IT costs significantly. However, if the only difference between a design-to-order IT shop and a configure-to-order IT shop is the virtualization, then several problems remain:

- If server configurations are still novel, there is still risk involved with adding or changing workloads
- This risk must be understood and pass a change management process
- Automation will be limited to linear tasks

Many IT shops have established more efficient processes based on internal infrastructure "standards." Typical infrastructure standards might include "small Linux server," "large Windows server," "VMFS LUN," "network distribution switch," and so on. Standard configurations can frequently be coded into standard processes, such that entire configure-to-order processes can be automated. However, the server farm as a whole is still built one application at a time, rather than as an aggregate, and service management must still be customized for each application.

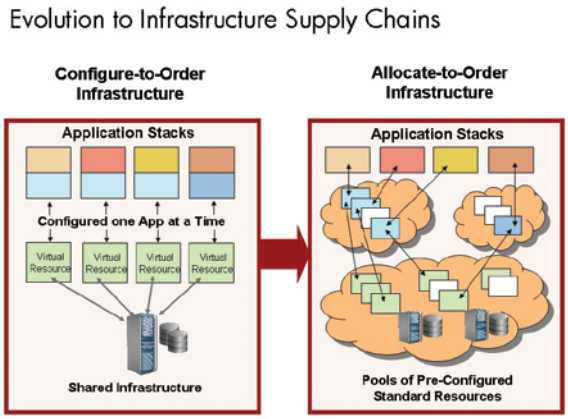

Figure 2. Moving from Configure-to-Order to Allocate-to-Order Infrastructure

The New Model – Allocate-to-Order

The newest model, and the model of this architecture, is "allocate-to-order." With allocate-to-order, a new application is assigned to a pre-existing resource, via a service binding. Allocate-to-order involves a menu of standard, pre-inventoried, well-known resource types. For example, "small Windows server" and "medium Windows server" may be two types of standard resources offered on a menu. When an application owner (the customer of the menu) selects "medium Windows server," he will get exactly the same resource type that has been provided to any other "medium Windows server" customer.

Note: There is a big difference between infrastructure standards versus infrastructure services: infrastructure standards alone represent simply a recipe for building an element, while an infrastructure service actually supplies the element as a service.

With Allocate-to-Order, infrastructure is provided as standard services from menus.

Benefits of Allocate-to-Order (versus Configure-to-Order):

- Infrastructure is fully standardized using a service menu – repetition allows processes to mature;
- Changes to infrastructure are well-known and can be pre-approved
- TTM for new applications can be dramatically reduced
- Capacity management at the whole data center level is greatly enhanced
- Standardized pieces make troubleshooting much less complex

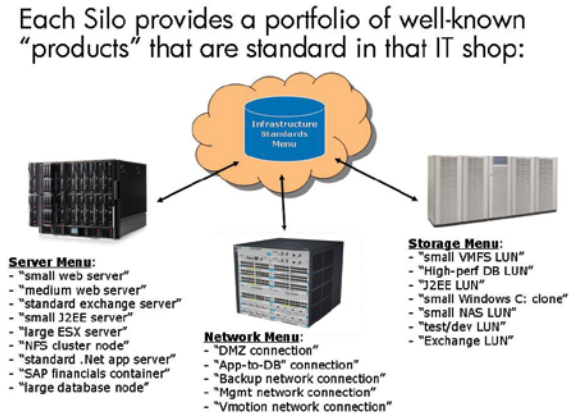

Figure 3. Infrastructure Resource Services Menus

Infrastructure Standards

Much like a supply chain, infrastructure standards can be layered. For example, a standard database server type can be composed of standard LUNs and standard VLAN configurations.

Infrastructure standards are obviously not a new concept. All IT shops have standards -- however, providing standard infrastructure elements using a supply chain concept is new. With a supply chain concept, standard resource offerings are each managed as if they were a product line, rather than simply a recipe.

What does it mean to manage infrastructure as a set of "product line" services?

- Each product line is a mini business – it has customers, suppliers, costs, forecasts
- The entire product line is under change control (not just individual products)
- With adequate quality control, all products are essentially identical
- Thus, processes for producing and managing each product are essentially identical
- Each product line has a lifecycle
- Each product line is managed in the aggregate

This approach can provide significant benefits for IT service management, including:

- Better capacity management – aggregate capacity is managed against a proactive forecast
- Better problem management – all elements and their interactions are well-known and homogeneous
- Better change management – changes are no longer novel, allowing for better understanding and lower risk
- Better quality – higher-volume, homogeneous tasks increase repetition and experience

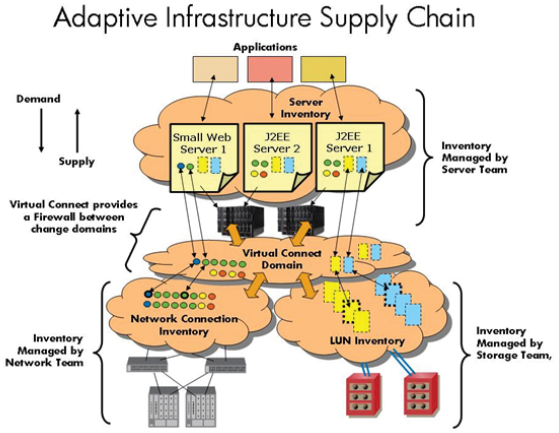

Figure 4. Infrastructure Supply Chain

Adaptive Infrastructure Supply Chain

HP's Data Center Architecture consists of four primary components:

1) Server management domain
2) Network management domain
3) Storage management domain
4) Firewall between these domains, provided by Virtual Connect

This architecture is not about HP technology components; rather, this architecture is about IT and service management people, processes, and technologies. Each domain provides infrastructure as a service, utilizing many tools, some of which HP will provide as part of our Infrastructure supply chain architecture.

The primary interfaces between service providers and consumers in this supply chain are resource inventories. Each provider offers resource services for subscription, typically to another organization in IT. HP's architecture focuses on the inventory of network connections provided by network operations to the server team, the inventory of storage resources provided by the storage team to the server team, and the menu of standard server types offered by the server team to application owners.

Domain Separation

Virtual Connect

Virtual Connect plays a key role in Infrastructure supply chains. By virtualizing server I/O, Virtual Connect hides server changes from the network and SAN. Separation of these domains is critical to automation and efficiency, for two reasons – reduced complexity and increased volume for any specific task.

Complexity is a direct result of:

- Number of subsystem interactions
- Number of organizational/people interactions
- Number of potential effects of any change
- Number of people who must approve or be consulted for any change

Whenever server changes don't impact the network, both functions can increase their productivity by separating their operations into smaller parts and simplifying those parts. Similar with the network – if network changes don't affect the server team, both teams can increase productivity. Modularity of process is central to the efficiency of any supply chain.

Resource Menus

The key to achieving process separation is to establish the right interface between supply chain processes. The interface for infrastructure supply chains is the resource inventory. An inventory is an agreement between a supplier and a consumer of capacity, that combines the two in space and time, yet provides a buffer that allows them to operate asynchronously. Beyond the basic SLA, the consumer does not care how the supplier provides the inventory, and supplier does not care how the consumer uses it. This allows both domains to change and evolve with minimum complexity.

Imagine the complexity behind the supply chain for your Microwave oven – starting from raw materials, progressing through the huge diversity of processes and organizations that add value and provide piece parts to one another. Most of the suppliers in the chain are servicing a widely diverse set of customers and markets -- only the very last organizations in the chain are focused on producing microwave ovens. Obviously, the volume from this diversity of uses (beyond just microwave ovens) is paying for all the energy that transforms sand, metal ore and plastics into the bag of parts that you consume as a single oven. But nobody opened a "ticket" to scoop sand off the beach when you called the dealer to ask about microwave ovens!

Two natural results of supply chains emerge:

1) Specialists in the chain feed off of high volume, and thus serve multiple customers.
2) Consumers choose standard offerings, even if they are not a perfect fit for the requirements.

For IT to achieve this efficiency, two practices must become common:

1) Suppliers of services must get better at recognizing their customers' usage patterns.
2) Customers must settle for "what's on the menu."

By following these practices, the supply chain of IT will optimize itself over time.

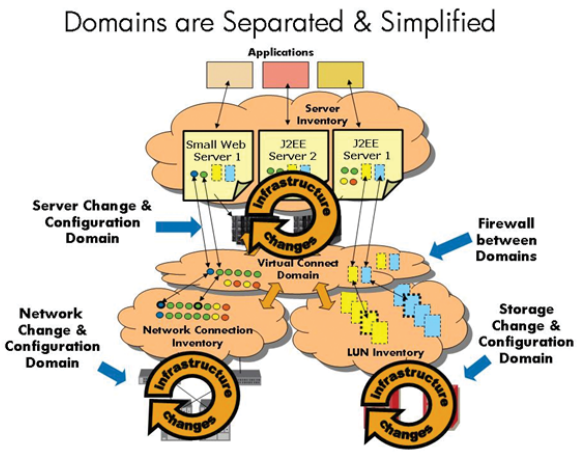

Figure 5. Separate Change Domains

Virtualizing the Network

As a prime example of domain separation in infrastructure, network virtualization is one of the most fundamental challenges for a supply chain approach.

Why do networks present such a big challenge to the resource inventory concept?

- The network is a single shared complex system, not a discrete set of pieces.
- Cost-effective networks are always over-subscribed, disabling the intuitively-pleasing approach of "carving bandwidth" statically throughout the network for each endpoint.
- Cost-effective networks also tend to be switched – not meshed – which also disables the concept of dedicated, carved bandwidth.
- Laws of physics – which dictate what capacity and connectivity exists between various points in the physically-cabled topology – cannot be "virtualized" the same way space & time can be divided and multiplexed in other domains like storage space or CPU cycles. Any network that offered unlimited capacity between all points would be grossly unaffordable.
- Network policies are not resource-oriented, but rather application-oriented. For example, ACLs are application-sensitive polices that pertain to the usage pattern of a server, not the properties of the server itself. This prevents network configuration of servers from being accomplished asynchronously from application deployment.

The remedy is to present the network as a pre-planned inventory of physics-sensitive and application-sensitive connection subscriptions. By dictating what types of connections can occur where, the laws of physics behind the topology are manifest by the inventory itself. For example, the available inventory can limit

how many connections of which type are available in a particular set of racks, and the inventory can be tuned over time in response to actual usage. Much like choosing seats on a flight, the inventory of options reflects actual capacity.

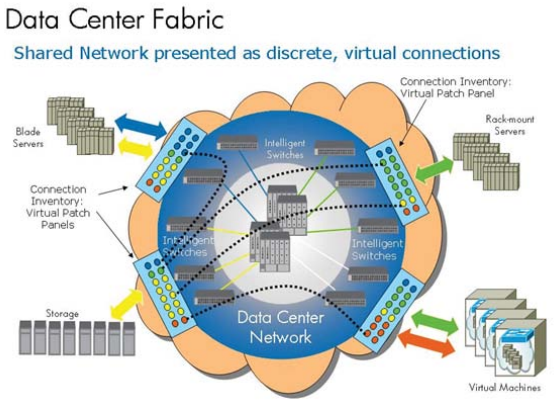

Figure 6. Network Virtualization

Application Lifecycle with Infrastructure Supply Chains

The advent of infrastructure supply chains has only a minor impact on application lifecycle. The traditional application lifecycle consists of a six stages – requirements gathering, designing, building, deployment, operations, and optimization.

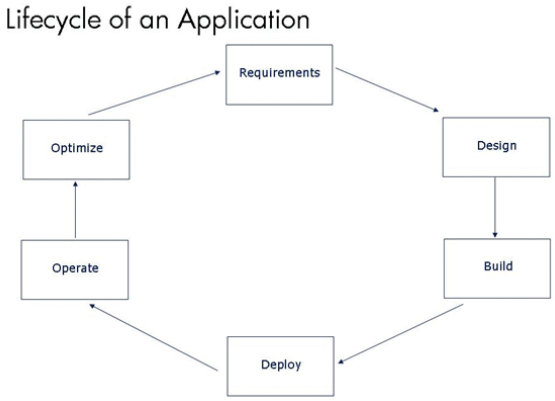

Figure 7. Application Lifecycle

With infrastructure supply chains, the primary change to the application lifecycle is to replace "design" of infrastructure with "choose from a menu" of those components that are offered as a service. For example, if requirements for an application dictate a 4-CPU Linux server with 4 GBytes of memory, the design stage would involve a search through the IT server standards for the best server menu option that offers at least four CPU cores and 4 GBytes of memory. If the "small Linux server" has enough CPU cores but not enough memory, then the designer must opt for a larger server option on the menu. The designer may be forced to use an 8-core server in order to get 4 GBytes of memory. While this may seem sub-optimal for this particular deployment, there may be hundreds of other servers for which the current menu option of 4-core / 2GByte is perfect. The key is not to manage IT infrastructure one application at a time, but in the aggregate. By using a well-known service option, this designer allows the supplier to do something they know how to do very well.

There are many ways to remediate the sub-optimal menu option problem over time:

1) If the ROI justifies it, offer a new menu option with 4 cores and 4 GBytes of memory
2) Develop a "thin provisioning" approach, which actually allocates a smaller amount of capacity initially than the menu specifies, and provides the full capacity only if it is demanded
3) Choose an alternative application derivation that makes more optimal use of the existing service menu

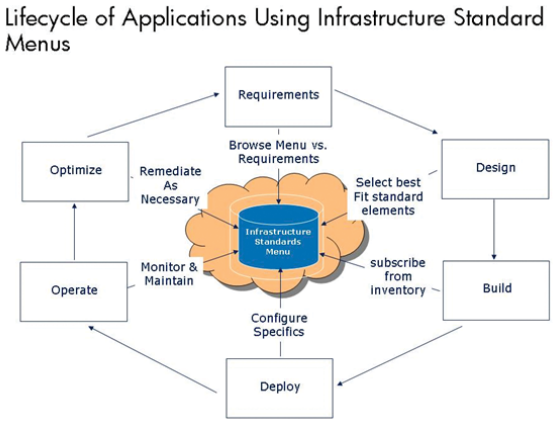

Figure 8. Application Lifecycle with Infrastructure Supply Chains

Summary

As management processes evolve, the natural evolution of infrastructure deployment will move towards a supply chain service model. The primary concepts behind Infrastructure supply chains is that of rendering infrastructure as a service, based on home-grown standards that are inventoried in high volume.

HP is committed to establishing a service-centric management toolset, with significant co-development between server, networking, and storage business units. Stay tuned for more news and information on this topic from HP.

For more information

To learn more about HP ProCurve Networking, visit:
www.hp.com/go/procurve

Technology for better business outcomes

To learn more, visit www.hp.com/go/procurve

4AA2-5725ENW Rev. 1, June 2009

ProCurve Networking