Event ID: 286699
Part 2: How to efficiently deploy and virtualize your application environment

Elliot Kass: Good morning, good afternoon or good evening, depending on where you are in the world, and welcome to today's webcast on how to efficiently deploy and virtualize your application environment. This is the second part of a five-part series on small-scale IT projects for big-time business results brought to you by InformationWeek, IBM and UBM TechWeb. I'm Elliot Kass, and I'll be your moderator today.

To ensure that this is as interactive an event as possible, we're going to make a few announcements before we begin.

At this time we suggest that you disable your popup blockers. The webcast is designed to be interactive and makes use of popups or widgets to help you interact with the presenters. You can launch these widgets from the docking tray at the bottom of your console. Some of them will open in a separate browser window. You can resize them by dragging the lower right corner, moving them by dragging the top bar and minimize them by clicking on the blue button in the upper right corner.

During the webcast you can ask questions at any time. Just type your question into the Ask a Question text area and then click on the Submit button.

The slides will advance automatically throughout the event. You can download a copy of the slides by clicking on the Download Slide icon located below the presentation window.

Towards the end of the program we will ask you for your feedback in a separate popup form.

If you are experiencing any technical problems, please visit our webcast help guide by clicking on the Help icon on the far right of the docking tray at the bottom of your console. In addition, you can contact our technical support line, the number for which is located in the webcast help guide.

Now let's get going with today's presentation on efficiently deploying and virtualizing your application environment.

Our speaker today will be Marc Haberkorn, who's a product manager with WebSphere CloudBurst Appliance at IBM. Joining Marc during the Q&A portion of our webcast will be Yakura Coffee, who is a Software Development Community Manager at IBM.

By the way, in case you're slightly confused because you're seeing Dustin Amrhein in the slide there as today's presenter, Dustin was unfortunately delayed due to a flight that was delayed by the weather on the East Coast, so Marc and Yakura have kindly agreed to step in to fill in for him today.

So, now, we're going to begin by kind of gauging where our audience is at with the first of our three polling questions.

On your screen in front of you now you should see the first question, what's the greatest imperative for your IT organization right now? We'd like you to select one of the following: cutting costs; reducing the complexity of our systems and operations; improving our response and deployment times; freeing more of our budget for new systems and applications; all of the above; or none of the above, we have other priorities. So, again, the choices are: cutting costs; reducing the complexity of our systems and operations; improving our response and deployment times; freeing more of our budget for new systems and applications; all of the above; or none of the above, we have other priorities.

And now we're going to wait for a moment to have the responses come in. Hopefully everybody is participating in the poll here. And here we go. So, by far, far and away all of you agree that all of the above are imperatives for your IT organization right now. Very few of you thought that none of them were important priorities, and then the rest of you singled out different ones, with reducing complexity being the dominant one there, but really by far you all agreed that all of these were important issues for you.

And with that as kind of a background here, or a context, I'm going to turn this over to Marc now to begin today's presentation.

Marc Haberkorn: Thank you.

I want to talk to you a little bit about a couple of products in the WebSphere portfolio with respect to virtualization in both the application infrastructure virtualization and server virtualization spaces. Again, my name is Marc Haberkorn. I'm the Product Manager for WebSphere CloudBurst Appliance, WebSphere Application Server Hypervisor Edition and WebSphere Virtual Enterprise offerings.

So, take you ahead to Slide 6, and I want to make a couple of points here. The primary point I want to make here is that if we're looking at removing cost, which I know that wasn't the top item on the previous slide, I suspect that some of the 50 percent of you who selected all of the above would have chosen cost as your primary driver had you been required to prioritize those four -- the most common request I see from customers I visit is usually they're looking to reduce their cost to comply with budget cuts that have been enforced upon them based on the recent economic conditions we're all operating in -- so one of the concerns, then, is, if we're looking at reduce costs and make our budgets sort of more whole, we've already cut our resources to the bone. We're not really looking to cut more resource, cut more spending. We're looking to be more efficient with the spending we've already cut.

So one of the things to look at is that you're spending $0.70 on $1.00, based on a survey that we've done, just on maintaining your current IT infrastructures versus adding new capabilities. Quite a few of you on the previous poll said you were looking to add new capability, so this should resonate with you, as well. If you're spending $0.70 on $1.00, 70 percent of your budget, on not adding new capabilities, that seems like an area you should target some gains for efficiency. So the question then is what gains in efficiency can be made?

And so let's look at Slide 7. Basically there's a loop, or sort of a feedback loop that emerges, and you want to break this feedback loop. And the way that that works is when you want to get a new application environment up and running there's a lead time associated with that. It's usually in the order of months. Two to four months is the average we see. I've seen anywhere from two weeks up to upwards of a half of a year. And that's because you have a lot of serial processes that are often manual in nature. You have approval processes before you can even get started. Then you go through a procurement exercise. You wait for shipment. You install the hardware in your racks. You procure licenses for all the stuff you need to install on top of the hardware. You install your operating systems. That's one team. You install your middleware. That's another team. You install your application. That's another team. There's generally some configuration that happens. And at the end of the day or the end of four months, not a day, you have an application environment up and running.

Because these processes are manual in nature, you have sort of this tendency for bugs to be introduced, and this is particularly evident when you're moving between stages of application management, so between dev/test and QA into production. You're effecting change in your environment. You're trying to duplicate an environment. And you have a situation where you're introducing bugs. These bugs are difficult to detect, because they're not code bugs in your application. They're someone missed step 200 out of 225 on your standard SP for your golden topology, and that's tough to find. So it creates this perception, and it's reality, that it's expensive to set up an environment. So once you get an environment up and running, you're not going to have it pried out of your cold, dead fingers basically is what it amounts to. People hold onto these things just in case they can be used in the future, and they sort of hoard this resource. And that requires future environments, then, closing the loop, to go back to this two- to four-month lead process -- lead time process, in order to get a new environment. So you need to break this cycle.

And what we've prescribed for that is on Slide 8, and that is the WebSphere CloudBurst Appliance and the -- the IBM Hypervisor Edition products, to actually expand that beyond just WebSphere. WebSphere CloudBurst Appliance is a secure cloud management appliance. It's a 1U device. It basically allows you to set up patterns of your infrastructure based on one click. So you have a multiserver, multiproduct topology codified in a pattern. You basically make that available to a set of folks in your team based on their roles. And they can come to the appliance at some time in the future and request availability in a self-service manner for that pattern.

What that delivers is reduced setup time, so you're clicking one button and getting an entire multiserver, multiproduct environment connected together. You're codifying your infrastructure for reduced risk, so that whole idea of when you're moving from one environment to the next, where you were injecting manual effort and therefore risk for error, you don't have that error anymore, because computers don't think. They execute the same steps each time. You have your entire environment codified in a pattern. It will be the same when I deploy it versus when you deploy it versus when someone else on our team deploys it. You have simplified maintenance and management. We'll talk more about that as we go forward.

And it dispenses a set of virtual image products we call Hypervisor Edition images, and these are basically -- I'm going to use a chemistry analogy of atoms and molecules. If the patterns we talked about are molecules, these Hypervisor Edition images are the atoms that you use to construct those molecules. You have a set of Hypervisor Edition images available, and I'll talk about that list as we go forward in this presentation, and you can basically combine these Hypervisor Edition images together to form patterns.

So, the Hypervisor Edition products listed on Slide 9, these are basically shipped ready to run on Hypervisor, which supports several Hypervisors. We have PowerVM, z/VM, VMware ESX capability. These are all OVF images. You don't have to do any installation to get these to run. They're basically just freeze-dried images that you simply place on a Hypervisor and click Start and it runs. We have WebSphere Application Server Hypervisor Edition. We have Process Server in this form, WebSphere Business Monitor, WebSphere Message Broker, WebSphere Portal, DB2, and we have a set of other images planned in the future, so watch this space for future announcements.

So, a little more detail on what the WebSphere CloudBurst Appliance is. So, you have on this box a set of artifacts, right? You have the set of users and groups, which represent your teams in your environment, your permissions, your logins and passwords and sort of what permissions are allocated to those various accounts. You have a palette of virtual images, these Hypervisor Edition images we talked about. You have a set of patterns and scripts that are used to deploy and manage those Hypervisor Edition images in the context of patterns.

So you bring your own enterprise cloud. So you have a set of hardware that's presumably underutilized. So you have, again, x86, Power or z hardware. You connect that to the box. You provide a storage device for all the boxes in your cloud to share. You provide network information like IP addresses, subnets. And basically then you have -- you've marshalled a cloud. You have the CloudBurst Appliance managing a set of resources in the cloud. This comprises a set of what we call virtual systems. You can do a couple of things to customize the way that you use these systems in the cloud, and we'll talk more about that on a future slide.

So the next slide is the life cycle in the on-premise cloud. You basically create your custom WebSphere environments. You have a set of virtual images that you're able to extend and capture. So this notion of modifying the operating system, for example, that you're deploying, so things like fix packs of the operating system, you have virus scanners, you have monitoring agents, things that you need to place on the operating system itself, you can do through an extend and capture mechanism. You can then create patterns of these images, once you have effected your customizations, and arrange them in custom ways. So if you have two database nodes, a couple of app server nodes, a portal, and, for example, some web proxies out front, you can codify that into a pattern and place it on a palette that people can use in the future to deploy in a self-service manner.

Then, of course, you're enabling these folks to deploy these patterns you've created to a private cloud. They can, upon dispensing of their patterns, provide some parameters like the IP addresses that should be used. For example, if part of the pattern needs to be in the DMZ and part of it can be behind the DMZ, behind the firewall, you can effect that through the deployment process.

And then you have the manage piece, where these things are running in the cloud and you need to manage the virtual systems. So you're monitoring your resource usage, and you can do that through our UI. You can start, stop and remove the virtual systems, so when you're done using something or you've done an analysis on your environment and said that we have too high of a percentage of our utilization taken up by a specific type of workload and you want to remove some of those to get some resource back, we provide that capability.

You can create snapshots of virtual systems, and this is actually done automatically before you apply fixes and service level upgrades. So, and that's the next point, you can use the WebSphere CloudBurst Appliance to apply either emergency fixes that you get from IBM Level 3 support or applying fix packs and that sort of thing.

And, finally, we allow you to manage your software license use. So, in a shared environment where you have multiple people using a shared set of licenses, you have a situation where you need to make sure your group in aggregate doesn't put you over your license entitlement. And of course no single person in that group is going to know what your license position is at any point in time. So it's incumbent upon this appliance to allow you to keep track of what your usage is versus your entitlement position, and we do support that.

The next slide talks a little bit about the catalog, so one of the pieces that was on the appliance that you had seen earlier. This is a set of images, basically, that are provided by IBM. We have the Hypervisor Edition images. So you have a set of these images loaded on your box in a catalog. Any customizations you've made by using our extend/capture mechanism to modify the operating system are also reflected on this catalog. And then you have parts from images representing node types. So within one image, for example, the WebSphere Application Server Hypervisor Edition image, you'll have multiple parts. You'll have deployment managers, you'll have custom nodes, you'll have job managers, that kind of stuff, that you can deploy at various points in your pattern.

You have a set of user-supplied script packages, then, so, of course, we give you general purpose software that you've found many creative ways to use, and it's impossible for us to reflect all those possible use cases in our UI. So we allow you some freedom, then, to provide script packages to do things like installing applications, tying into external database services, for example. All those sorts of things that you need to do in a custom way we give you sort of this catch-all mechanism through a script package to do whatever you need to do beyond what we can give you.

Then you have patterns, and these are also on the box. We give you a couple of patterns as samples out of the box, but we also give you a pattern editor. It's a simple drag-and-drop mechanism where you take palettes off the catalog that we talked about on the previous slide, you basically drag them onto the canvas, you connect these pieces together, you specify a few parameters, and that's the way you can build patterns. And these patterns can then be shared across teams based on things like user groups and permissions.

So if I create a custom pattern for my development and test team to test a specific application, I can provide that pattern exclusively to my development and test team for that application. And I can do a couple of things by using that mechanism. First, I guarantee that everyone on that team has access only to certified patterns, meaning that there's no risk that we're using different versions of patterns. And you also can sort of simplify the list of options for each person so they're not wading through lists of -- long lists of patterns that are irrelevant to their particular need.

The next slide talks about some of the opportunities for customization of these patterns. So we mentioned the extend/capture, and I'll give you some details how this works. We basically give you what we call a sandbox. You can deploy a single image into the sandbox. You can then make changes to that image in the order of operating system tuning, installing monitoring agents, installing database drivers, installing virus scanners and firewalls, sort of things that are needed on that image and may be required on all images in your datacenter. For example if you have an operating system team who has a golden image that they use for all of their operating systems and no one's allowed on your network without using that operating system image, then you can reflect that golden image in your base Hypervisor Edition image using this extend/capture mechanism. And then everyone who creates an image or a pattern in the WebSphere CloudBurst Appliance environment will be required to extend that base golden image you've provided.

The second piece is within the pattern. So we have these arrangements of multiple images in a pattern. You can see -- you may not be able to see it too well -- at the bottom you have deployment manager, custom nodes, IBM HTTP servers. These are sort of the various node types that are available in the WebSphere Application Server Hypervisor Edition. And you can see that we've arranged them with a certain number of each node. You have, I believe it's four custom nodes, you have two IBM HTTP servers. You can change things like the number of nodes, the versions of these individual products that are used at each point. There's actually, if you look closely you can see a lock. It's a yellow icon. You can lock individual components of patterns such that future people who deploy them have no capability of modifying certain locked parameters. And what that allows you to do is guarantee consistency from one use to the next.

The third thing you can do for customization is script packages and parameters, and I mentioned this previously as a catchall mechanism. It allows you to do basically anything you can do with a command line. We give you a command line. You specify what you want executed. And you can do anything you can do in a shell script, anything you can do in a Jacl script, a wsadmin script. Anything you would like to run you have capability to do here. The goal, of course, is to minimize the amount of work you have to do here. So we do support your existing scripts that you have for installing applications. Of course, we've delivered quite a bit of value in tying clusters together, connecting nodes of various sorts to one another just through the mechanism of pattern management. This is the catchall mechanism that if you have custom needs you can enforce those or effect those through this mechanism.

The next slide emphasizes the point that you're bringing your own cloud, and it talks about sort of the various artifacts you bring to the table when you're setting up a cloud. So you'll set up a set of hypervisors. These are one of the three supported types of hypervisors: VMware ESX, PowerVM, z/VM. You give us a list of those, simply IP addresses and logins and passwords, and then we'll connect to them and basically marshall a cloud based on those hypervisors. You give us a set of IP addresses that we can use when we deploy individual nodes and we'll map those IP addresses either randomly, if it doesn't matter, or we actually specify -- allow you to specify policies by which IPs are mapped so that you can give things that are in the DMZ you can give externally facing IPs and things like that.

And then you have storage, which needs to be connected to each of the hypervisors in the cloud. This is not storage for the box itself, for the CloudBurst Appliance box. It's actually storage for the individual nodes that are managed by CloudBurst. And this can be either local storage, local to each machine, or it can be a shared SAN. And this is for storing things like the images that are deployed out to each box and that sort of thing.

The next slide just mentions that -- and we're on Slide 16, for those of you who are following along -- this is an automated deployment process. So the way this works is a user logs into the appliance. They have some credentials that have been given to them by an administrator. Based on those credentials they are presented with a list of environments that have been made available to them. They select one of those environments to be made available. They specify when they want it to be made available, and optionally an end time that it can be removed. And then the CloudBurst Appliance takes over.

It chooses a set of hypervisors in the cloud. We have some intelligent placement mechanisms that will basically determine where it's optimal to place these environments in the cloud. We'll create the virtual machines there. We'll inject the IP addresses. We'll start the VMs and the workloads. It can be WAS, it can be whatever Hypervisor Edition image you're running. And then we run the scripts that you've provided us. So after we've started the WAS nodes and connected the clusters together and set up all the cells and that kind of thing we can run your scripts to do any sort of last-minute cleanup.

The next slide talks about maintaining your deployed virtual systems. So then the question is, once these are up and running in the cloud, how do I keep them current on maintenance without redeploying them? And the answer is that you can use the WebSphere CloudBurst Appliance to do this. You can actually schedule maintenance, which is a nice feature, so that if you have a maintenance window in the middle of the night you can sleep through it and let the CloudBurst do your work.

You can apply emergency fixes through CloudBurst Appliance. So you simply upload an IBM PAK file and you can execute that through WebSphere CloudBurst Appliance. You can also provide -- roll out service level updates. So you can upload a new IBM Hypervisor Edition image into the catalog and then use that to upgrade either existing images that are running in the cloud or use it, obviously, for new deployments that will be on the current level.

The next slide talks about managing software licenses in the cloud, and I mentioned this earlier as a key requirement when you're setting up shared environments like cloud computing. We basically allow you to, or require you to enter in the number of PVUs you've entitled, or you've purchased, for the various products that we have in our portfolio or our palette. So let's say you have 2,000 PVUs of WebSphere Application Server Hypervisor Edition. So you enter in 2,000 PVUs in here. And then you can select how we enforce that. You can say don't enforce it; I'll handle it myself. You can say warn me, but allow us to go over our entitlement if we need to. Or you can say restrict me, and basically if you're reaching your entitlement level and you don't have any room left in your entitlement position we won't let you go over your entitlement, we'll stop you. So that's your choice. You can pick what level of control you like. You can also specify a threshold, say 90 percent, and we'll warn you when you reach that threshold that maybe it's time to go add more PVUs or look at what's deployed and perhaps reclaim some of that resource.

So, bringing it all together, you have basically this catalog of images and patterns. You have a set of users and teams and groups who are allowed to use these various pieces. You have people who are responsible for setting up the golden topologies in terms of operating systems. You have your administrator. You have your developers and testers. Right? These people can all work in concert with WebSphere CloudBurst Appliance based on their various roles, and this picture sort of depicts how that plays out.

As far as datacenter integration, we actually have some capability to integrate with Tivoli products, like the Tivoli Service Automation Manager, and basically they can coordinate management of a set of resources. We also have integration with the Rational Automation Framework for WebSphere. And there's basically a script package that comes in the box, out of the box, WebSphere CloudBurst Appliance, that allows you to tie into a set of artifacts you may have in Rational Automation Framework. For example, you may have configuration databases that are things you've had around for quite a while and you don't want to have to recreate those in the format of WebSphere CloudBurst Appliance patterns. So what you can do is basically sublet, if you will, or subcontract out to RAFW and ask it to population the configuration on a deployed pattern from WebSphere CloudBurst Appliance. That way you can leverage your existing artifacts without duplicating them.

We have a reference to mention. The Haddon Hill Group is one of IBM's business partners who have used WebSphere CloudBurst Appliance in several engagements now to effect a more cost-effective method of provisioning into virtual cloud environments. They've seen quite compelling results in terms of ROI -- savings in the millions. They're citing 100 times faster time to market in terms of deployments and getting new environments up and running. They've reduced the order of one to two months down to hours, so that's a compelling game changer for them.

I want to shift gears a little bit now that we've gone through WebSphere CloudBurst Appliance and talk about another offering in our portfolio called WebSphere Virtual Enterprise. So, WebSphere Virtual Enterprise, I want to talk about two forms of virtualization. One of those is basically the end user client virtualization, which would be things like desktop virtualization and application virtualization. We're not playing there. IBM's products in this space are concerned with virtualization in the datacenter. And, although it doesn't show up on this slide, the upper right-hand quadrant should say application infrastructure virtualization. Some sort of a formatting issue with this tool.

The server virtualization space is, of course, what you know to be things like VMware ESX, PowerVM, z/VM, your server -- your hypervisors, basically. You also have storage virtualization, using [inaudible] and things like that, network virtualization. But we also have this application infrastructure virtualization, and we'll talk about what that delivers as we go through this presentation. Oh, there, the application infrastructure virtualization showed up.

So, introducing WebSphere Virtual Enterprise. It does a couple of things for you. It can lower your operational and energy costs. So it improves utilization, and it does this basically by allowing you to map workloads to logical pools of resource instead of mapping them statically to particular machines. So, stated differently, you may have a set of 10 servers, and today they're siloed over a set of applications. Well, you can correct that situation and improve that situation by creating a logical node group in WebSphere Virtual Enterprise and then deploying workloads into that node group, and WebSphere Virtual Enterprise will intelligently move those workloads around inside that arrangement of nodes intelligently, based on a couple of things.

So one of them is observe demand, and this gets us into the next piece, where we actually do traffic shaping and flow control. So we're looking at the demand coming in. We can both affect where placement of applications in the back end is and we can also affect routing and affect flow control of incoming work. And we look at things like priority as configured for different applications. We look at service policy goals and things like that in making those decisions.

The notion of spreading out these workloads over a logical resource pool also gives you benefits in availability, because you have, instead of all of your eggs in one basket, so to speak, any of those nodes in that given node group are capable of running any of the workloads in that node group. So if you lose a server you simply move the resources that were working or the workloads that were running on that server onto other servers. It's a sort of an insurance policy.

And one of the other things we offer is better management of health, so we can do preemptive things around looking at JVM age, heap size, sort of indicators that could sort of forecast or indicate that in the future you'd have an outage. We can take preemptive action to sort of subvert those outages.

And, finally, the last piece that we like to talk about with WebSphere Virtual Enterprise is the management of upgrades, the management of application editions. So you can have basically two versions of an application running side by side whenever you roll out your new application upgrade, and you can quiesce your old version and move all new work to your new version. So what that allows you to do is effect an upgrade of an application without taking an outage. So you've deployed Version 2 of your application, and none of your customers who are using that application incurred an outage, regardless of whether they were running on Version 1 or Version 2.

So, some of the value scenarios, and I mentioned some of these in the previous slide, but you can save money through server consolidation. I mentioned on the previous slide that typically people are allocating their resources in siloes today, and instead you can sort of logically pool your resources together and consolidate servers in that way. What this allows you to do is instead of provisioning for peak utilization of each individual workload you can sort of provision for average usage and then give a little extra that's shared across the whole pool, so instead of having enough extra in each silo, you're only having to provision extra resource across the pool, and that gives you some savings in terms of hardware cost.

You can also improve your environment manageability. So you can basically enable your middleware operations team to run the environment easier. You get better visualization of what's going on. We have some charting mechanisms to give you a view of various metrics in your environment. You can handle version upgrades, as I said on the previous slide, much more easily, and that sort of thing is important to the operations team.

And, as I mentioned, you can improve your availability through not only the ability for every node to execute all workloads in a node group, but also through the health management capability that allows preemptive addressing of possible problems, like, again, JVM lifetime, heap size limits, that sort of thing.

So, to sum this up, what is WebSphere Virtual Enterprise? It provides operations optimization features to your environment. There's dynamic operations, which are basically intelligent placement of workloads in your environment. You have server management, you have health management, and health management, again, is this preemptive actions that we allow you to take in response to certain observed situations, and runtime operations monitoring, so we give you charting capabilities where you can look and see certain metrics as to the management of your systems, both response times, concurrent usage requests, that sorts of thing.

So, standard topology of WebSphere Virtual Enterprise looks something like this. You have, obviously, a firewall outside. You have a web server tier. So Port 80's opened up through that firewall. You have a web server tier that has a firewall behind it, so that's your DMZ. And then behind the second firewall you have two tiers. You have the ODR tier with an on-demand router and you have your application server tier. And what the on-demand router does -- we're going to look at this a little more in depth on the next slide here -- what the on-demand router does is takes care of a couple of things like classification of incoming workloads, prioritization and flow control, so sort of like a traffic cop, and then routing and load balancing.

So if you look at this picture -- and I'm going to walk you through this, because this is a pretty busy chart -- on the left-hand side you have three colors, green, blue and pink -- stock trading, some management and you have financial advice, -- account management and financial advice. So you have these requests coming in. And it may be more important to prioritize stock trading higher than financial advice, because it's more real time, more time sensitive. So let's say we have a policy that says stock trading gets the highest priority, we're going to let them through first.

So what you do is you first have to classify incoming requests into these various queues, so we've sort of sorted them out. And then you have a mechanism inside that handles prioritization and flow control. So he says, okay, I have some queues of each of these types of requests. How should I map those to the available resource on the back end? So he'll have basically the ability to throttle incoming requests, so this can actually handle denial of service attacks and things like that, so it doesn't inundate your back end.

So then, after it handles its flow control, it basically goes back to the routing and load balancing piece. So this piece has sensors on each of the back-end app server nodes in the second tier, the app server tier, and can tell which nodes are overworked and which ones have extra capacity, and it can route the work intelligently that way based on where applications are running, what the available resource is and that sort of thing. And, as I mentioned earlier, with the placement technology we can actually spin up additional instances of each of these applications in our app server tier. So if it becomes apparent that my very important stock trading application is reaching its limits, we might need to stand up additional instances of that application.

So these autonomic managers that I talked through are specified in detail on this slide. And I already went through a couple of these things, so it may not be required to go through them again, but the autonomic request flow manager, that's sort of the thing that handles the flow of requests and the order of requests that I mentioned in the ODR. Dynamic workload management sort of handles how these requests are load balanced across all your available servers. Your application placement controller decides how many instances of each application need to be started and where they should be started. And your health controller decides how to enforce your health policies, so if you say -- or it does enforce your health policies. If you say, "I don't want any of my JVMs to live beyond a week, or a day," then it'll keep track of the lifetime of a JVM and it will restart the JVM whenever it reaches its limit. Similarly, if you specify a maximum heap size, it can enforce those limits, as well.

So, if you look at middleware virtualization, typically, or traditionally, your server topology is fixed. We talked about that before. WVE changes this paradigm. It treats the cell as a resource pool that's sort of more logical than it is static. You can define which nodes in a given environment are capable of doing various things. For example, you may have some nodes that are on a certain subnet and other nodes that are not, which means not all nodes have connectivity to the same databases, the same servers. You may have certain hardware that has SSL accelerator cards in it, or something like that. So you can specify which nodes are unique for certain capabilities, and this can aid in the load balancing effort in terms of making sure that workloads get to appropriate servers that can actually handle those workloads.

The notion of a dynamic cluster is basically the logical entity that encapsulates an application. So you install applications into a dynamic cluster and you map a dynamic cluster onto a node group. You can have multiple dynamic clusters in a node group. Basically what happens is a dynamic cluster will have a certain set of nodes it can run on, so it'll have a minimum of two nodes and a maximum of 10 nodes, for example, and that dynamic cluster will scale according to the needs of that application based on the observed incoming demand for that application, which, again, our ODR node understands that demand, and the available resource on the back end to serve it. So that's how dynamic clusters work.

We do offer first class support for non-WebSphere platforms. So, of course, we understand the WebSphere platform quite a bit better than the others, and we have control points there, so we can do a little bit better -- or quite a bit better, actually. We have a white box knowledge of what's running in WebSphere. But we have black box knowledge of the other app servers that are out there, as well, and we can do some things for them. So we can do some automated life cycle management for things like WebLogic, JBoss, Tomcat. And you can read through this slide if you have specific questions on that support.

So, combining your virtualization approaches, one of the things that we've seen is that when you combine this application infrastructure virtualization approach with the server virtualization approach you can get quite a bit of benefit. We've actually seen basically a 200 percent benefit by combining the two together in certain surveys we've done. And what I mean by that is lower utilization, lower response time, higher throughput. And the reason for this is if you're relying on your server virtualization layer to move applications around or make fine-grain changes, well, they can't make fine-grain changes. They're moving entire virtual machines. What we do is more fine-grain tweaks. We can move individual JVMs. We can actually move requests from one machine to another machine intelligently. And, as a result, because we have this more fine-grained insight, or this more context, we can make better decisions and get better results. And that's the message of combining server virtualization and WebSphere Virtual Enterprise. It results in better results.

And then the last thing I'll say about this space is that the WebSphere Virtual Enterprise capability is actually able to be used in the context of WebSphere CloudBurst Appliance through something we call the Intelligent Management Pack. It's an optional add-on that you can use in conjunction with WebSphere Application Server Hypervisor Edition, and it allows you to basically deploy WebSphere Virtual Enterprise capability in a CloudBurst Appliance-managed environment, so you're combining your server virtualization and your application infrastructure virtualization in this way. And that is available today.

There are some resources available. I recommend if you have specific questions about these offerings, especially CloudBurst, we actually have some demoes on YouTube that can show you what the UI looks like in more detail and more clear detail as to how you interact with it and that sort of thing. If you go to YouTube.com/WebSphereClouds, and the link's on here, you can look at some of those five- to 10-minute demoes. They're nice bite-sized things and they're pretty useful.

I'll leave you with a mention of Impact. Of course, we have Impact2011 coming up in April, and a lot of us will be out there, so feel free to reach out and we can have some discussions either before or during or after Impact hours. So feel free to reach out. I think my email address is made available here. It's hmarc@us.ibm.com. Feel free to reach out to me and we can talk shop at Impact.

So with that I'll hand it over to Elliot to execute our next poll.

Elliot Kass: Thanks, Marc, and, really, thank you for sharing those insights and helping us all get a better sense of how we can deploy our applications more efficiently.

Now we're going to take a moment to gauge what our audience is thinking. So this is the second of our polling questions. At your company, what's the average lead time to deploy a new application environment? We'd like you to choose one of the following, and the choices are: one to three months; two to four months; three to five months; and greater than five months. So please choose the response that most closely matches your reality. And, again, the question is, at your company what's the average lead time to deploy a new application environment? One to three months; two to four months; three to five months; or greater than five months.

And we're going to wait a moment for the results to come in. And here we go. So, it definitely looks like some accelerated development cycle is in order here. The predominant answer by far, more than half of you are taking more than five months to roll out a new application or a new application environment. And it's worth noting that the second strongest answer, at a little -- around a fifth of you is three to five months. Only, what is it, 13.5 percent of you are operating between one and three months at this point.

So we'd like to move on to a second polling question here. This will be the last of our polling questions, by the way. At your company, how much of each IT dollar spent goes towards maintaining the current IT infrastructure versus adding new capabilities? So, how much of your IT budget or each dollar that you're spending on IT is going towards maintenance as opposed to new applications or new capabilities? Please select one of the following: $0.90 or more going towards maintenance; $0.80 to $0.90 going towards maintenance; $0.70 to $0.80 going toward maintenance; $0.60 to $0.70 going towards maintenance; and less than $0.60 going towards maintenance, meaning the other $0.40 or more are going to new applications or new capabilities.

And, again, we're going to just wait for a moment. And here are the responses. Okay, so, this is pretty interesting. The plurality of you, the largest number of you, at a little over a third, 36.3 percent, are spending between 70 and 80 percent on maintenance, with the remainder going to new capabilities. That's kind of in line with industry averages. But a very substantial number of you, just under one third, at 31.8 percent, are spending less than 60 percent on maintenance, with the remainder available for new applications and capabilities, which I think is kind of a happy surprise, from my point of view, and definitely shows some progress out there in terms of reversing that 80/20 split and freeing up new money for new capabilities.

So, we're going to move on now to the Q&A portion of our webcast. But before we do, we'd like all of you to fill out this feedback form that is now open on your screen. The form's very important from our point of view. We take your feedback very seriously, and we use it to improve these sorts of broadcasts. So if you could all take a couple of moments to fill out the form, we'd very much appreciate it.

And now we're going to move on to our Q&A. So, Marc, with the assistance of Yakura Coffee, will be trying to respond to your questions.

And the first question has to do with the security or VPN capability that's associated with the WebSphere products that you were talking about, Marc. So maybe you want to take a few minutes and -- actually, I think the question is suited for both of you. So, Yakura or Marc, if you want to take a stab at this, please go ahead.

Marc Haberkorn: Yakura, do you want to try one?

Yakura Coffee: Sure. In the aspects of security, or VPN, if you're talking about security into the WebSphere CloudBurst Appliance itself, the appliance itself is set up where an administrator can lock down the specific users of the CloudBurst Appliance. So if there's a part or a logical organizational structure, maybe development and test, you can logically separate patterns in virtual systems so only a specific group of people, even controlled by an LDAP server in the organization, can use the development patterns in the development virtual systems, and also vice versa with the testing infrastructure that way.

Elliot Kass: Great. So, here's another question. This question comes from Ron, by the way. Can you run multiple virtual environments on the same group of hardware servers at the same time? And specifically he's talking about parallel universes running at the same time but that are running independently of each other.

Yakura Coffee: Yes, I can take that one also. Yes, you can do that. CloudBurst manages a defined set of IP addresses, hypervisors and storage. So CloudBurst dispenses patterns of virtualized environments onto your infrastructure layer. So, for example, if you're running a VMware ESX server farm that's defined in CloudBurst, you may deploy as many complex virtual environments that you like, as long as you don't exceed the various thresholds in that environment that CloudBurst will track for you, so, for example, the IP ranges don't be exceeded, storage allocations and the PVU. So your software licensing thresholds, the Hypervisor Edition images dispensed by CloudBurst should not be exceeded. So you can definitely run parallel universes as long as you don't exceed those thresholds.

Elliot Kass: Great. Here's a new question from Michael. He says he may have missed it earlier when you were talking, Marc, but can CloudBurst work with multiple hypervisors such as IBM and VMware?

Marc Haberkorn: Yes. WebSphere CloudBurst Appliance supports VMware ESX, IBM's PowerVM and IBM's z/VM. And you can use a single box to do multiple of those. So if you have a situation where you use VMware ESX for dev/test and you use PowerVM for production and QA, you can use the same pattern across all three.

Elliot Kass: Okay. This question, I guess, gets at the financial side of this. What kind of initial investment does hypervisor require, and how does IBM charge for the software? Do you do this by the seat? Specifically, the questioner is saying that there are around 15 developers in his shop, so he wants to know how many seats would they require?

Marc Haberkorn: We license our software on a PVU model, which is basically a capacity model. It essentially assigns different weights to different technologies. So it's basically a per core model, and different cores require different numbers of PVUs. If you search for PVU table on Google you'll probably find it as your first hit. But we basically will license, it's a perpetual license, you pay so much per PVU for each of the products, and then you have the option to renew maintenance each year after the first year [inaudible].

Elliot Kass: Okay. Now, this question actually may be something that you spoke to earlier, Marc, but I guess we need to provide some clarification. Do you have to run the CloudBurst Appliance in order to use the hypervisor software, or can these be used independently of one another?

Marc Haberkorn: We allow the WebSphere Application Server Hypervisor Edition images for VMware ESX to be used outside of CloudBurst, but the rest of the images require WebSphere CloudBurst Appliance.

Elliot Kass: Okay, thanks. So, this questioner says, and I'm sure he's speaking for a great many people out there, we can't purchase any new software without first demonstrating the ROI. Do you have any suggestions about how to do that for the WebSphere environment?

Marc Haberkorn: Absolutely. So, we actually have a couple of things we can do there. We have POTs and POCs to allow you to actually try out the box before you purchase. But beyond that we have a free service we call business value assessment, and we actually do an interview with you to come to mutually agreeable terms in terms of what certain cost parameters of your environment are, what your labor rate is, how much labor you need for certain activities, how many -- what your hardware utilization rates are, all that kind of stuff, and we'll put that into a model and basically give you a report that you can use, which is actually pretty helpful in getting buy-in from your finance types.

Elliot Kass: Great. I think we have time for one more question, and this one I think is directed at Yakura. Can you explain how the WebSphere patterns work and whether you need CloudBurst or hypervisor or both of them to create those patterns?

Yakura Coffee: Sure. Patterns represent meaningful WebSphere configurations that are also deployable via WebSphere CloudBurst to a set of hypervisors within a private cloud. So WebSphere CloudBurst represents the WebSphere middleware environment in a pattern. These patterns are essentially a representation of a WebSphere Application Server topology with desired customization. So using WebSphere CloudBurst pattern editor that's inside WebSphere CloudBurst you can interact with a graphically -- interface to create your WAS middleware environment, to include topology definitions, custom settings and end user applications.

Elliot Kass: Great. Thank you.

And, really, thanks to both of you, Marc and Yakura. We know you had to step in at the last minute for Dustin, but with little time to prepare you've both done really a terrific job in helping to explain and clarify these things for our audience, and we appreciate it.

So, this pretty much concludes today's webcast on efficiently deploying and virtualizing your application environment, brought to you by InformationWeek, IBM and UBM TechWeb.

For more information, please visit any of the resource links that are available to you. You should see them on the screen in front of you.

Within the next 24 hours you'll receive a follow-up email and details about a link to today's presentation. You'll also be able to download it on demand by visiting NetSeminar.com.

This webcast is copyrighted 2010 by United Business Media. The presentation materials are owned by or copyrighted by InformationWeek and IBM, who are solely responsible for their content. Individual speakers are solely responsible for their content and their opinions.

I'm Elliot Kass. On behalf of our guests, Marc Haberkorn and Yakura Coffee, thank you all for your time, and have a great day.