

Staffing for Converged Infrastructure, Part 1: Taking Charge of the Changes

February 21 2013 | By [Robyn Weisman](#) | in [Converged Infrastructure](#)

My college roommate is a rare breed, a female network-monitoring engineer. She works at a large educational institution and thought my post on [why repairing IT services takes so long](#) was “right on!”

So because she’s actually in the IT field, I asked her, how would you go about restructuring your staff to handle a more dynamic infrastructure? I posed it in a general fashion because I knew her organization wasn’t using Cisco UCS, but given the way infrastructures are changing across the IT industry, I still thought it was a valid question.



She sighed and said that’s a “permanent problem” because the silos within her organization are still so separated. She said that ideally you would find people that can understand the complexities of an entire system, but even with those people on-board, getting access to those people is almost impossible – and even if she had that access, figuring out how to communicate with them is a whole ‘nother problem.

Well, roomie, this series of posts is for you! Recently Zenoss spoke to several IT leaders in different verticals to find out how they’re handling their staffing as they migrate to converged infrastructures.

For this first post in the series, Zenoss talked to a senior director of applications at a large SaaS company. I’m focusing solely on him today because he has taken it upon himself to examine his company’s business model and make changes that would best work under these emerging conditions.

1) Sifting and Recombining

His original applications team was responsible for the ownership, design, and deployment of applications, but this method was starting to cause problems. When the ownership and design functions were lumped together, one of the two realms would end up monopolize most of the team’s time, “and something would get starved.” And that was assuming the deployment part of the equation didn’t suck time away from the other two:

...the actual deployment, literally putting the code on the servers and reconfiguring the servers...inevitably became the priority. So everything else didn’t get done.

After he was able to tease out these three aspects of the development process, he then recombined these teams back into a larger application team:

I took the NOC and the monitoring team and [the ownership, design, and deployment] teams and said everything that has to do with the application is one giant organization and we have functions within it that we’re trying to scale.

He had the insight into his team to delineate the primary stressors, sift out the three elements of the development process, and reassign employees to one of those three areas. I’m guessing that narrowing the job responsibilities in this way enabled members of his team to better focus on their respective responsibilities while his “recombining” of these teams back into one greater team meant that each sub-department understood that ultimately the sum of their actions would be greater than the parts.

And such an approach seems a more satisfying one than the one my college roommate has to use, which is to hide her tools from “everyone” because they aren’t motivated to understand them.

2) Reorienting Toward Business Goals

Listening to more of what this senior director of applications had to say, I learned that the next big organizational shift in his staff’s restructuring would be to reorient his overall team to its line of business (LOB):

My application team will be oriented to the engineering team, oriented to product owners, [and] oriented to marketing teams, which [will] ultimately support the customer better.

Trending Posts

[See Under the Hood: Watch a Live Demo of Private Cloud Service Orchestration with Cisco UCS \(81\)](#)

[Top 10 Presentations on Cloud Computing \(59\)](#)

[Orchestrating Cloud Operations – A Practical Example \(32\)](#)

[The Price Tag Question \[Converged Infrastructure Learning Curve Part 6\] \(28\)](#)

[Free ZenPack Development Training Webcasts in March \(25\)](#)

Zenoss on Twitter

Take 5 minutes to help shape the future of #opensource! [bit.ly/WB088t](#) #FutureOSS
7 hours ago

What makes #opensource software attractive to you? Better quality? Lower maintenance cost? Tell us! [bit.ly/WqU7Lr](#) 4 days ago

“To have an efficient operations team, you need one tool not six” via [@kentwithat](#) [@jdooley_clt](#) [bit.ly/WprLkN](#) #VCE #Zenoss 5 days ago

Follow @zenoss

Find us on Facebook

This shift puts the focus outward toward the product and the market it is serving, which lead to cooperation between the different application teams:

We collaborate about the architecture and design [of applications]. My development team has an idea [of] something we're going to sell, and they work with my design team to decide how it would work — how they would use load balancers, what type of database requirements, [and so forth] — almost like a pre-sales system engineer role. That design team will capture in a document form, 'Hey, is this what you're building? Is this what you're thinking?' And [they'll] get sign-off in an informal partnership role, so that by the time the engineers finish building [it], the environment is already commissioned. That means servers are bought, racked, NetScaler configurations are built.

At first glance, his decision seems self-evident. If you were part of a business, wouldn't you automatically set things up so that they would benefit your business objectives?

But then I think back to college when, out of curiosity I asked to see my roommate's homework from some master's level electrical engineering class. That stuff may have been easy for her, but it requires a lot of concentration. I can see why people doing this type of work could easily get tunnel vision – and fail to consider the larger goals if they lacked a culture where those goals mattered.

3) DevOps or Work?

This idea of focusing on the business problems, rather than the technology problems, seems comparable to **DevOps**. Although the name DevOps sounds like an IT-style version of a **Reese's Peanut Butter Cup**, it's really about aligning IT teams with business goals – exactly what he is doing with his team.

And for his part, he said that his group is about as DevOps as it gets, since he can't think of an alternative description. But he said the labels are irrelevant because it all boils down to "work:"

Ultimately we're responsible for the running of the applications and people are building the applications, and so we have to partner with them to make sure that things scale. In our world, because of SaaS, the only thing that we really care about is product availability, that there's always 99.9% availability to my customers. Everything that happens behind that is just the work.

I give this senior director of applications a lot of credit. It can't be easy to break your team down to the molecular level and then consolidate it back to a single organism with a meaningful purpose. He probably had to deal with a lot of trial and error, and most businesses are reluctant (at best) to allow for mistakes. The leadership qualities he has shown should bode well for this SaaS' future, and the changes he has implemented could be a great template for your NOC.

Meanwhile, I am hoping my college roommate forwards this post over to her boss. It's one thing helping me pass my engineering requirement back in college, but she shouldn't have to tackle all the networking responsibilities in her department without at least a little cooperation on the part of her colleagues, you know?

In my **next post**, I'll look at how a couple of financial firms are going about restructuring their IT staffs...

Like Send

You, Gita Khadiri and 3,3 others like this.



Circle us on Google+

+725 Recommend this on Google

Tweet 12 Like 1 1 Share 3



Robyn Weisman has written about technology for over a decade, specializing on IT-related topics, such as server and storage virtualization, data storage and cloud computing. Her email address is robyn@robynweisman.com.



Netcool
**Beyond Netcool:
 The Open Alternative**
 WATCH THE WEBINAR »

Tags: [Cisco UCS](#), [Converged Infrastructure](#), [Devops](#), [monitoring](#), [NOC](#), [Saas](#), [Staffing](#)