

Tack Driving, IT Style.



Don MacVittie, 2013-10-04

Driving tacks in carpentry is just that – putting small nails in to hold something (normally carpet) in place. It is a task that is easily accomplished by the most novice of carpenters, and heck, even a non-carpenter like myself can get it right most of the time. A repeated process of nice, clean, repeatable steps.

And it relies on an entire set of industries, without which the task would not be composed of clean, repeatable steps.

Tacks are cranked out at a factory, carpet at an entirely different factory. The tools that drive the tacks? Yeah, completely different set of factories, and have a huge impact on the quality of the experience. A ball-peen hammer can certainly be used to drive tacks, but is just not in the same class as a state-of-the-art brad nailer. But then the brad nailer needs to be maintained, and the ball-peen will work covered in rust, soaking wet, and with the handle cracked.

IT has followed this same type of course, from simple, reliable, and slower, to complex, generally reliable, and very fast, in so many different areas that it's nearly impossible to list them all. For IT, adding "portable" in there makes it more chronologically appropriate also.

Early in my career, a smart old salt told me that the best inventions in the history of man were because *you* needed it. He praised things like the computer for the forward thinking, but maintained that it was people using it who invented the tools that improved it time after time. While I don't 100% agree with him, corporations have come up with some pretty astounding inventions in the interest of R&D, there is a nugget of truth to that phrase.

And IT follows it perhaps more than other industries, simply because the users of IT products are IT specialists themselves. We utilize tools that make our lives easier and offer more capabilities, but for any given IT product, many of you could recreate it if necessary. That means the users are *very* savvy about the product, and have the ability to offer feedback more directly than a consumer market would.

But a funny thing happened on the way to the massive datacenter... We lost track of how many bells and whistles we knew about and/or had at our disposal. I've said this before, and I'll say it again, you have gear capable of doing much more than you're using it for sitting in your DC.

And that means potential over-expenditures when it comes time to implement new features/technologies. Knowledge is indeed power when it comes to IT products. In a highly competitive marketplace, features are added to products as a competitive advantage, but knowing what's sitting around and utilizing it effectively is your part of that. If a product you already own can solve your problem, don't buy a new product and increase maintenance costs. Even if the product requires an upgrade or extra licensing to achieve your goal, it's worth the effort if there is not yet another box in your DC.

Of course there are times when the new functionality can't be had in your datacenter, or when the implementations in devices you already own are not suitable to your use, but honestly, are you even asking those questions? Most IT shops aren't. They have a device that does X, so they use it for X, and don't even consider it when Y comes up as a requirement unless there is an advocate for the product inside the datacenter.

I get it, there are never enough hours in the day, I've been there. But there are never enough hours in the budget either. Knowing what you have and how to get the most out of it is guaranteed to save those budget dollars in the long run.

I know when I worked in datacenters there was a fair amount of the "Oh that does X" mentality. And yeah, when I got "that does X" from someone who'd been there longer than me, I participated. New stuff, I learned all about. Gear that had been there a while, even if upgraded repeatedly, I didn't generally do the same with. Assign it. Get your team to learn about all the gear in their purview at a deeper level. It'll save you not only money, but maybe a ton of time.

Of course, I'm one of those people who, if the source code for a library is available, I add that to my project instead of the library. Easier to debug, but I *like* knowing the toolset at hand too.

Keep kicking it, look around, make sure you have at least an idea of capabilities.

And for those of you who share one of my many hobbies... Yes, this applies to that kind of tack driving too.

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com

©2016 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at f5.com. Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5. CS04-00015 0113