Programmable, Automated Infrastructure: SaaS Success Raised by the Power of 2

This article was originally published on TechTarget (searchsolidstatestorage.techtarget.com) and authored by Bill Laberis, Partner, Gillin & Laberis – Original source

IT leaders, developers and architects are acutely aware of the growing business requirement for continuous development of new solutions and IT services, and continuous improvement to existing ones. There is no shortage of advice on how to do so—some of it actually worth listening to.

But much of this advice suggests more deeply integrating development into infrastructure management, or even pushing it out of core IT into development. This is beyond the infrastructure capabilities of most organizations. Don't fret. There actually is a lot you can do to foster far more agile (i.e., speedy and flexible) development, and more easily create new features and then deploy them in-market faster than ever.

Two keys are automation and programmability, which in fact are readily available to be pressed into service. Then layer in a measure of all-flash array, which not only enables much faster data transfer rates, but also is widely considered as the underpinning or foundation for the next generation of high-performance business applications.

Let's look at automation first, in the context of the development and test (Dev/Test) environment. It turns out that Dev/Test is deservedly one of the more popular and successful use cases for OpenStack, the open source platform that does wonders to enable infrastructure as a service.

With OpenStack, developers can leverage various automation tools to support continuous integration and service improvement. This translates into more differentiated product offerings, which in turn can translate into real competitive advantage in the market. But one caveat: To really unleash the potential of OpenStack in Dev/Test, you have to deploy the right OpenStack storage. And today, that likely means an all-flash array. AFA has proven particularly useful and robust in continuous integration environments, where new work is often integrated on a daily basis. An automated build can

verify the integrity of these improvements and catch errors before they are part of a solution going out the door. Further, it is important to automate as many processes as you can, including build, promotion, release and follow-up monitoring.

Here are a few features to look for in that all-important underlying storage platform for Dev/Test:

- Ability to run Dev/Test and production from the same system, standardizing assets between development and operations
- Flexible customization of application volume types
- Ability to quickly clone an environment with little to no impact on storage capacity or running applications

Walking hand-in-hand with automation is programmability. A good way to think about programmability is to consider for a moment just how much time developers spend tweaking, adjusting, verifying and otherwise babysitting code. And of course, we know how much they love that kind of work— about as much as they enjoy a root canal.

This is where programmability shines, as it does in general by eliminating the need for those tedious manual processes that bog down real development work. As SearchCloudComputing notes, programmability uses higher-level descriptive language(s) to automate historically manual processes.[1]

The benefits are obvious. For one, it follows that a happy developer is a productive developer who feels he or she is contributing something meaningful to the organization and its services. The tedium and counterproductive work associated with manual operations tasks does not make for happy developers. Also, programmability results in developers NOT having to rely (and wait) on system administrators to manage matters in a Dev/Ops environment. Instead, developers can save several steps (that is, time and money) by conducting their own quality assurance or mock deployment.

Programmability and automation: They are the new power couple in the developer's world.

■ NetApp®

1 "Infrastructure as Code definition," SearchCloudComputing