Is PaaS Just Outsourced Application Server Platforms?

Lori MacVittie, 2010-29-04

There's a growing focus on PaaS (Platform as a Service), particularly as Microsoft has been rolling out Azure and VMware continues to push forward with its SpringSource acquisition. Amazon, though generally labeled as IaaS (Infrastructure as a Service) is also a "player" with its SimpleDB and SQS (Simple Queue Service) and more recently, its SNS (Simple Notification Service). But there's also Force.com, the SaaS (Software as a Service) giant Salesforce.com's incarnation of a "platform" as well as Google's App Engine. As is the case with "cloud" in general, the definition of PaaS is varied and depends entirely on to whom you're speaking at the moment.

What's interesting about SpringSource and Azure and many other PaaS offerings is that as far as the customer is concerned they're very much like an application server platform. The biggest difference being, of course, that the customer need not concern themselves with the underlying management and scalability. The application however, is still the customer's problem.

That's not that dissimilar from what enterprise-class organizations build out in their own data centers using traditional application server platforms like .NET and JavaEE. The application server platform is, well, a platform, in which multiple applications are deployed in their own cozy little isolated containers. You might even recall that JavaEE containers are called, yeah, "virtual machines." And even though Force.com and Google App Engine are proprietary platforms (and generally unavailable for deployment elsewhere) they still bear many of the characteristic marks of an application server platform.