SaaSGrid℠

The SaaS Application Server

[Q&A]

Q: What's the easiest way to understand SaaSGrid?

SaaSGrid is a distributed SaaS Application Server that eliminates the difficulties of building and delivering Software as a Service (SaaS). Conceptually, SaaSGrid parallels the operating system or the modern day application server; it defines a concrete layer that "sits below" your application and acts as a host for your application. As a hosting layer, SaaSGrid manages network resources while providing an environment that allows your application to function as a SaaS offering even if it wasn't written as one. Furthermore, very much like your operating system might provide you a control panel, SaaSGrid provides a variety of tools via the web to interact with your deployed application.

Q: Is SaaSGrid just a technology solution?

Absolutely not! Although SaaSGrid performs impressive technical magic to host relatively standard web applications as SaaS offerings, it has significant strengths in helping you manage your SaaS business. Ranging from the ability to link to merchant accounts, accept credit cards, and define complex pricing structures to providing tools that focus on customer management and business analytics, SaaSGrid helps ensure that you have what you need to understand and react to your market and service your customers.

Q: Is SaaSGrid a virtualization technology?

SaaSGrid is not a virtualization technology that uses machine images to perform it's magic. Instead, SaaSGrid relies on a unique architecture to endow your application with virtualized functionality such as multi-tenancy. Virtualization normally helps solve certain types of scale and deployment issues, but is not a means for solving SaaS problems and does not provide any sort of SaaS delivery mechanics.

Q: Is SaaSGrid really a grid?

Yes. SaaSGrid uses a distributed grid architecture to host SaaS applications in a decentralized fashion, allowing for failure tolerance, easier scalability, and high availability.

Q: Does my application have to adhere to specific user interface guidelines, or is my user interface limited in any way?

SaaSGrid is not a virtualization technology that uses machine images to perform it's magic. Instead, SaaSGrid relies on a unique architecture to endow your application with virtualized functionality such as multi-tenancy. Virtualization normally helps solve certain types of scale and deployment issues, but is not a means for solving SaaS problems and does not provide any sort of SaaS delivery mechanics.

Q: Does Apprenda host SaaSGrid?

No. Apprenda's core competency is in building powerful SaaS enablement technology. SaaSGrid *is* a SaaS Application Server, and companies can either license SaaSGrid to create their own private cloud, or work with one of Apprenda's hosting partners who manage their own SaaSGrid cloud. This gives organizations the power to write a SaaSGrid application and have multiple options for deployment.

Q: Can I test SaaSGrid?

Yes! You can sign up for a SaaSGrid "sandbox" account, which gives you access to a non-production deployment of SaaSGrid hosted by Apprenda. Here you can deploy sample applications and utilize the SaaSGrid management portals.

Q: Is it appropriate to compare SaaSGrid to services like Amazon EC2?

No. SaaSGrid is not virtualization technology so a comparison to Amazon's EC2 is not appropriate. Services like EC2 provide a Resource as a Service offering that is closer to "bare metal." SaaSGrid provides higher level functions that provide targeted SaaS value. As such, EC2 can be leveraged via a private SaaSGrid cloud as part of the underlying compute resources that SaaSGrid manages.

Q: Is it appropriate to compare SaaSGrid to Windows Azure?

No. While there are parts of SaaSGrid that cross over into what Windows Azure provides, SaaSGrid is focused on the higher level functions that provide targeted SaaS value. SaaSGrid is meant for B2B SaaS solutions, not just raw general purpose hosting. As such, various Azure services can be leveraged as value adds to a SaaSGrid deployed application.

Q: Do I need to learn a new programming language to build SaaSGrid applications?

No. SaaSGrid lets you utilize Microsoft .NET languages to write code and provides a slim, simple yet very powerful API layer that allows the application to interact with SaaSGrid in certain explicit ways. This also means that existing .NET based applications can be extended and deployed as fully commercialized multitenant SaaS offerings.

Q: I want to port and application the SaaS model via SaaSGrid. How long will it take?

It depends. SaaSGrid provides value two ways: explicitly through the application interacting with the SaaSGrid API and implicitly as part of deployment. The implicit portion (things like multi-tenancy and provisioning) require no effort on your part and are available out of the box. Therefore, the amount of time it takes to port is highly dependent on the size of your application and the richness of the explicit interactions you plan on taking. Given that you would have to port your application to SaaS anyway, SaaSGrid provides the fastest path to SaaS since the port requirements are fairly minimal.

Q: Can I integrate external systems with my SaaSGrid application?

Yes. SaaSGrid applications are built using rich web services. These web services can be exposed by SaaSGrid as a web API, allowing for integrations with external systems.