

[A Cloud PSA: Cloud Needs BCP Too](#)

Everyone is looking to the cloud for solutions to many problems. This could be for centralizing applications and data, ensuring availability regardless of location, enabling scale-out application infrastructure, or many other reasons. One of the reasons is to use the cloud as a BCP/DR (Business Continuity Planning/Disaster Recovery) target, which is also a great use-case for the public cloud.

Among all of the right reasons to use the cloud as a primary source for storing applications and data, are a number of challenges that are often overlooked. One of them being availability.

Cloud is Not Always About Availability

Public cloud infrastructure is designed and built to be massive, and scalable. Availability is a key part of the core of cloud infrastructure also, but we have to remember that despite all of the checks and balances that are in place when a public cloud infrastructure is built, there are real, unavoidable reasons which will cause a failure.

When I speak with service providers and cloud providers, they are clear on one thing. All the nines in the world won't get you to 100% availability unless you design your application for failure.

Wait, what? But what about my 5 nines (99.999%) uptime guarantee? Read a little more closely and you'll see that the result of an outage that violates the 99.999% uptime will result in only one thing, which is a refund. If AWS has an availability region go down, you will simply get an email apology and a pro-rated rebate on the lost hours of compute time.

Take the recent Azure outage (<http://www.forbes.com/sites/benkepes/2014/11/20/microsoft-delivers-a-post-mortem-the-reasons-behind-the-global-azure-alyipse/>) and we have another clear example. Despite all of the work done to ensure the highest availability, things happen.

Design for Failure, Always

I'm a big proponent for TDD ([Test Driven Development](#)) and I extend that into TDI ([Test Driven Infrastructure](#)) in order to apply the same practice. If you don't know how things will behave when they fail during UAT builds, you will certainly not be in a good position when it goes down in production.

Even the might Amazon Web Services has had challenges around availability. I've received emails on relatively short notice that my instances are going to be shut down for maintenance. The assumption at the AWS operations center is that you and I have done our due diligence and built a resilient application infrastructure.

Cloud is a Methodology more than a Technology

As great as public and private cloud infrastructure can be, the focus on what cloud can provide is a methodology for building application and infrastructure services. Moving to a cloud infrastructure one way to enable building more resilient application infrastructure. Just the same way that a spare tire can protect you from a flat tire if one happens, remember that there is only one spare tire. What happens when you have two flat tires?

This is really just a public service announcement for everyone who thinks that the move to the cloud is the answer to availability. It can be, but don't depend that creating a single instance in a public cloud will be the answer to keep your application available. Plan for the worst and hope for the best, not the other way around.