SSBG IT Solutions

Ph: (8621) 5404-3999 Fax: (8621) 5404-7759
Email: Sales1@SSBG.com.cn www.SSBG.com.cn

Lessons from high-profile IT failures

 By Jonathan Hassell

Original Article Post - http://www.computerworld.com/article/3114125/backup-recovery/lessons-from-high-profile-it-failures.html?upd=1472801735530 (August 31st 2016)

Delta passengers wait to check in following a system-wide computer breakdown, at Newark International Airport in Newark, N.J. on August 8.
Credit: REUTERS / Joseph Ax

This summer's outages at Delta and Southwest have much to teach all of us in IT

Computerworld | Aug 31, 2016 3:00 AM PT

It has not been a good few months for the health and consistency of airline information technology. Two huge outages within a couple of weeks of each other -- caused by simple component failures -- resulted in massive passenger disruptions and cost two U.S. airlines millions of dollars in lost revenue and customer compensation.

These events, while of course most painful for those who experienced them, present quite a few opportunities for learning and improving our own processes, and that's what I'd like to explore in this piece.

First, a little background. What ended up being a faulty router took down the entire Southwest Airlines operation for a day on July 21 and caused rippling effects for several days after the original outage. (A

fact that might surprise you is that Southwest is by a wide margin the largest domestic carrier of passengers in the United States.) The Dallas Morning News reported the fallout.

"The outage occurred early Wednesday afternoon after a network router failed and the backup systems failed to kick in," the newspaper reported. "Although the outage was fixed about 12 hours later, the scale of the disruption wreaked havoc on Southwest's operations for the next several days as the Dallas-based carrier worked to get planes, crews and passengers where they were supposed to be." In total, the airline said it canceled about 2,300 flights, or around 11% of the total it would have otherwise operated in that time frame.

(The line "backup systems failed to kick in" will ring familiar to Amazon Web Services customers, as a similar failure of backup systems took out much of Amazon's cloud hosting operation back in 2011 and really kickstarted the process of using Amazon's availability zones and other fault-tolerance features among cloud customers.)

Then, a couple of weeks later, on August 8, it was Delta Air Lines' turn at the Wheel of IT Outages, with hundreds of canceled and delayed flights during the middle of its demanding summer travel season, all due to an electrical component failure.The Wall Street Journal reported the story thusly: "An electric problem at its Atlanta headquarters occurred at 2:30 a.m. ET and the airline was forced to hold hundreds of departing planes on the ground starting at 5 a.m., according to Ed Bastian, the chief executive, who apologized to customers on a video. The technical problems likely will cost Delta millions of dollars in lost revenue and damage its hard-won reputation as the most reliable of the major U.S.-based international carriers, having canceled just a handful of flights in the most recent quarter."

Apparently the underlying technology issue at fault in the Delta outage was a switchbox -- essentially a giant fuse box that routes power into and out of a facility -- that failed at Delta's headquarters, according to Georgia Power, the public utility that supplies electricity to the location in question. What is not clear is why an outage that occurred at 2:30 a.m. was not able to resolve in time for flights to begin being dispatched at 5:00 a.m., nor why the cascading delays from the 5:00 a.m. cancellations could not have been less severe, or why they could not have been rectified more quickly.

What does all of this mean?

The Delta and the Southwest outages show how a single IT failure at the wrong place at the wrong time -- still, even after all of these years of planning and talk of the importance of disaster recovery -- can quickly cost millions, even in the course of just hours.

We have had decades of high-availability options: Different methodologies to either scale up with beefier redundant hardware or scale out with more cheap commodity boxes on hot standby and in clusters, failover options for both Windows and Linux that move operations across geographies in a matter of milliseconds, and now even infrastructure as a service possibilities where you just run backup operations in someone else's data center when you need to.

These options have all come down in cost, too. Where you used to need budget allowances in the millions to build any sort of failover capacity, now failover can honestly be as simple as purchasing a few hours' worth of runtime services with a credit card. (That is certainly too simplistic for a billion-dollar airline, but most of us do not run billion-dollar airline operations.)

An important tenet that I think many folks miss when they start planning for business continuity and their infrastructure: If your business depends on IT in any sense for its normal operation, then you

should be planning for high (and ideally continuous) availability. Many people think that disaster recovery and high availability are the same thing, are mutual goals, and have equal value for organizations. I think in this day and age, that is a mistaken impression.

Disaster recovery is the mode you enter when your technology operation has gone down in some way, whether it's a technical failure or Mother Nature dropping a hurricane on top of your data center. It describes the processes and procedures (and the requisite infrastructure) you would need to pick up the pieces and get back to a normal operation. The whole time your disaster recovery plan is being carried out, however, the implication is that your IT assets are not online, meaning your business is not serving customers and is not moving forward.

High availability, on the other hand, essentially plans for some downtime by adding compensating hardware and software that is always online, ready to be "failed over" to. While perhaps the full load of your system cannot be carried forward, operations essential to your business can indeed continue to run because your technology is available -- perhaps degraded, perhaps not, but at the very least, available.

That is not to say it does not make sense to plan for disaster recovery. After all, there are some situations that, despite all of your planning and anticipation and careful thought, you will not be able to foresee. But I think your best bet is to try to eliminate hard downtime, not plan how to get back from it.

Takeaways for targeting high availability

Why is high availability the most important goal? It is almost always less disruption for your operation to operate on a reduced or shed-load basis than it is to have your technology totally offline. Perhaps you cannot serve every customer in a normal way during an outage, but having at least the capacity to get critical functions executing completely may allow you to revert to manual or alternate processes on other fronts to move the business along overall.

In the case of Delta Air Lines, this most recent outage prevented some flight plans from being filed with the FAA, which essentially meant planes could not depart even if all other processes worked automatically or manually. If there were another site or location where flight plan submission software had been available, even at reduced capacity, then critical flights with the most passengers or flights that would get the right equipment to the right locations could have been dispatched, and many fewer customers would have been negatively impacted by the outage.

Takeaway: Find out what "gating" issues you have -- critical points in your process that must happen or the whole deal falls over -- and ensure you have secondary and tertiary backups for those points, as well as a plan to get them online in the event the primary fails.

When planning for high availability, geographic redundancy is also a concern. The only conclusion we can safely draw from the Delta outage is that all of the airline's critical infrastructure depended on some piece of hardware or software that ran inside the Delta headquarters building in Atlanta affected by the outage. Even if Delta's passenger service system or operations software was hosted elsewhere, as news reports have indicated, it is clear that some application running in a server closet or some piece of hardware in that building's data center was in fact a single point of failure for the airline.

One of the most basic lessons of planning for availability is that any critical points of failure need duplicate copies or versions available, most preferably in a location far away from the other operating node, so that local failures like electricity problems or weather disruptions will not affect the operation of the backup node.

Ph: (8621) 5404-3999 Fax: (8621) 5404-7759
Email: Sales1@SSBG.com.cn www.SSBG.com.cn

Takeaway: As far as disaster planning has come, you still cannot ignore having backup systems in a place far away from your primary systems.

Finally, testing your failover protocols is critical, because they will not always work when you want them to work. Both the Southwest and the Delta outages shared this in common: Backups and failovers didn't switch on when they needed to work.

The New York Times reported, "In Southwest's case, a backup system was in place, but the airline said that system was not triggered as it should have been when the router failed. And Delta said on Monday that it was investigating why some of its own critical operations had not switched over to backup systems." All of the investment in disaster recovery and high availability in the world will not help if those investments do not activate when needed.

Takeaway: Test your failover rigorously and consistently. Choose off-peak times for most tests, but also stress-test your plan by failing over during a slow, yet normal, day, and note any failures that happen for further investigating and remediation.

Would the cloud have helped?

Cloud fanatics have come out of the woodwork after these outages. "See what happens when you leave things on premises!" they say. "This would never have happened if you were using Amazon Web Services or Microsoft Azure," they say. But is this actually the case?

Yes and no. The cloud is a multilayered concept. When one says cloud, is one speaking of virtual machines running elsewhere? Is one talking about a fully managed web application that gets fault tolerance and availability right out of the box because the platform is handling all of that? Is one talking about failing over to a cloud data center that is essentially a private cloud hosted somewhere else? One has to be specific about what "cloud" means in any given scenario.

But let's take a little bit closer of a look. Could you port, say, the Sabre reservation system into AWS? No. Could you run flight plan filing software as a web app in Azure? Perhaps so, yes. Can you run an entire airline from Google Cloud? Undoubtedly not. But could critical pieces -- those "gating points" I talked about earlier in this piece -- find themselves in the cloud? I think so, yes.

There is no disputing the fact that airlines are very complex operations—you can make a credible argument that there is no operation that depends more on pieces and parts arriving just in time in just the right places, which is why even minor disruptions due to a few thunderstorms over a specific airport can have huge consequences for customers. And in this piece, I am not by any means attempting to say I know better than airline CTOs or that I know this airline or that one is or is not investing in cloud technologies. I have no special or insider knowledge of airline IT investments, nor are any airlines my clients.

What I am saying, however, is that the rest of us can learn from looking at the both the causes and effects of these outages. And it seems like in both of these cases, both airlines could have put planes in the air and gotten customers moving had they been able to fire up alternate systems -- which very well could have lived in the cloud -- in a timely manner. No, it would not have solved the underlying technical issue, and no, it would not have been a seamless, perfect solution. But neither was what actually happened -- thousands of stranded customers, vacations ruined and costs spiraling from paying customer damage claims.