10 items for your disaster recovery wish list

Version 1.0
April 16, 2008

By Rick Vanover

We all have to address disaster recovery (DR) at various levels, but we typically must apply the technology to fit rigid parameters—such as less cost or functionality—instead of being able to do it right. But what if you didn't have any limitations to hold you back? How would you create the perfect DR model? Here are some things (however unrealistic) that might go into building the perfect environment for meeting DR requirements.

1 The network is transparent

Providing transparent network connectivity is our number one challenge in making the ideal DR environment. If subnets for data center components were designed to be available across multiple locations without reliance on one piece in another data center, DR failover would be a breeze. Sure, a lot can be done to manage the use of a DR site through DNS and virtual switches—but if those could be avoided for a more natural configuration, the process could be made easier and work in a more transparent fashion.

2 The storage is transparent

Storage could arguably take the #1 spot on our list, since it's such a big pain in DR configurations. Technologies are available to handle storage replication and set up storage grids, but how many of us have the money to implement the functionality? The ideal DR storage system would also dispel any performance limitations when you're running the entire enterprise from the DR configuration. Limitations in performance may cause a selective DR, which makes for difficult decisions on what systems are truly required in the DR environment.

3 Everything starts with DR in mind

How many times have you come across a system that began as a pilot or simple test, was promoted to a live role, and is singular in nature and can't scale? These are DR plan inhibitors. If all systems are designed with the DR concepts in mind, all systems can comply with the same DR requirements and be an easy transition for administrators.

This extends to the peripheral components as well—storage, data recovery, networking, and access to the system should be created with DR in mind. But too many times, a system may have some but not all of the DR components in place. "Mostly compliant" with the DR model is still noncompliant.

4 All areas of IT meet the same requirements

Have you ever been irritated by partial compliance with an enterprise DR policy? An example would be when one application meets a different standard of DR—so maybe only a few clients can run the application in the DR configuration. Wouldn't it be great if the standing policy for the organization was to have full compatibility for the DR configuration? The ideal DR policy would provide funding and enforce the requirements for the DR configuration across all systems and groups within IT.

5 Disaster recovery is performed in a few steps

How we get to a solid and robust DR configuration will vary widely by size and scope, but the perfect conversion to the DR would a quick and contained process that is identified in a few steps per system, or a few steps for the entire environment. With the DR configuration so accessible, this would also be a good opportunity to enforce regular intervals where the DR configuration is used.

6 Documentation for failover to the DR site is clear and simple

An overly complex procedure to use a DR site can ruin the usability of the mechanism. The ideal DR environment has consistent and clear documentation that is practiced regularly so there's no guessing in switching to the DR model. In fact, regular use of the DR model can ensure that the remote DR site works as expected, keeps staff familiar with the procedure, and extends the life of primary systems by increasing idle time at the primary site.

7 All data recovery is native

The most challenging part of DR is the data recovery process. If a data recovery model is patched together using various scripts, watchdog programs, or other solutions that are not native to a product's feature set, the risk of data corruption and DR failure goes up. The ideal DR model would have solutions built into the product that consider all parts of a solution, as many products use more than just a database to provide the overall application.

8 Performance in the DR model isn't compromised

A comprehensive DR plan that meets all requirements from a design perspective yet can't handle the load is worthless. You don't want to have to decide which applications and systems are available at the DR site when you're in a DR situation. Limitations such as Internet connectivity, network bandwidth, shared storage throughput, backup mechanism availability, and storage capacity are all factors in gauging the overall performance for the DR site.

The perfect DR situation would be an exact inventory in the remote data center that models that of the primary data center. However, maintaining an equipment inventory in lockstep with another data center is nearly impossible. So the next-best solution would be a remote data center that meets or exceeds a performance benchmark set by the primary data center in all relevant categories.

9 The user experience in the change-over is nothing more than a reboot (if that)

Managing the transition to the remote data center is difficult enough on the data center. But the user side of the transition should be made as seamless as possible. Strong DR plans and mechanisms frequently base technology on DNS names (especially CNAME records) that can be easily switched to reflect a new authoritative source for the business service. This can include standby application servers and mirrored database servers, as well as migration to new versions with the simple DNS change.

Managing the refresh or the caching of the names can be a little tricky, but either having clients reboot or run the *ipconfig /flushdns* command on Windows clients can usually refresh any caching. The same goes for server systems that are affected by a DR transition; they may need to refresh their own DNS cache, so the same configuration steps may need to be followed on the server platform.

10 All things are possible for the small environment, too

The more robust DR configurations tend to present themselves naturally to the large enterprise. However, the small IT shops are at a resource disadvantage when it comes to architecting a comprehensive DR plan. The ideal DR model would be applicable to big and small environments, and all of the objectives could be reached with the small organization. Technologies such as virtualization have really been a boon for the small environment to achieve their DR objectives, and that frequently is the cost justifier for the initial investments in storage and management software.

Rick Vanover works for Safelite Auto Glass (Belron US) in Columbus, OH, providing central Windows-based server administration. Previous experiences included working for Dematic Corp (formerly Siemens L&A, Siemens Dematic, Rapistan) in various capacities deploying custom software solutions to the material handling industry using a mix of current hardware and software products. You can reach Rick at b4real@usa.net.

Additional resources

- TechRepublic's Downloads RSS Feed XML
- Sign up for the Downloads at TechRepublic newsletter
- Sign up for our IT Leadership Newsletter
- Check out all of TechRepublic's free newsletters
- 10 things to look for in an offsite backup provider
- 10 things you should cover in your business continuity plan
- Worst practices for disaster recovery

Version history

Version: 1.0
Published: April 16, 2008

Tell us what you think

TechRepublic downloads are designed to help you get your job done as painlessly and effectively as possible. Because we're continually looking for ways to improve the usefulness of these tools, we need your feedback. Please take a minute to drop us a line and tell us how well this download worked for you and offer your suggestions for improvement.

Thanks!

—The TechRepublic Content Team