JDEtips Journal January/February 2007 Volume VIII Issue 1

JDEtips On Disaster Recovery

Disaster Recovery Planning – A Case Study

By Gregg Larkin

Editor's Note: *Think "disaster" and you envision images of something horrific and destructive; Mother Nature (or even mankind) at their worst. But sometimes, it doesn't take an act of God to take down a system and stop business cold. Smaller leaks can sink the system ship just as quickly. And, while it may be cost prohibitive to prepare for a catastrophic, yet unlikely, event, it does make prudent sense to have a Disaster Recovery Plan(DRP) in place for more probable events—from hardware failures to power outages. If you haven't devised a plan yet (or you're not sure how fail safe yours is), Gregg Larkin shares a high-level view of an actual DRP design and the testing that goes with it.*

Disaster Recovery Planning – how often have we seen these three words together? We know the concept, but what does it take to build and support a strong JD Edwards® system to survive a disaster? In this month's issue we'll take a look at the terminology, architecture, planning, testing, and philosophy behind Acme Product's solid, disaster proof JD Edwards system. {Author's note – the company profiled in this article would like to remain anonymous, but the concepts and architecture discussed are real.}

JDETips – Readers, I would like to introduce you to Yosemite Sam, IT Director of Acme Products, maker of the world's finest rocket jet packs, steamrollers and novelty products. Mr. Sam, your company is in the business of mayhem and calamity; can you tell us how you safeguard your JDE system against mayhem and disaster?

Yosemite Sam – I'd be happy to. We take our mayhem seriously, and we guard against it just as seriously. Before we get into the architecture, let's discuss some terminology. There are a number of terms in and around disaster recovery planning and system reliability that need to be understood.

Fault tolerance – a fault tolerant computer system is one that continues to operate properly in the event of a failure of one or more of its components. Most servers have multiple, redundant power supplies, hard disks, and network cards.

High Availability – a high availability system is one that is designed to ensure a certain, absolute degree of operational continuity during a given measurement period. Availability is defined as the ability for the user community to access the system to submit new or modify existing work. Most IT departments include a schedule of planned downtime for system maintenance. A system that is high availability is not necessarily, and is rarely, a Continuous Availability system.

Continuous Availability – a continuously available system is one that avoids all single points of failure, is highly available, and allows online hardware, network, operating system, and application changes and modifications. Short of a total catastrophe, a continuously available system is always on-line. This type of system is extremely rare because it is extremely expensive to create and maintain.

The 99.999% Availability Fallacy

Beware the vendor, consultant or IT guy that promises 99.999% system availability. Almost all computer systems have planned or unplanned downtime. There are approximately 525,600 minutes in a year. Here are the stats for availability:

- 99.9% availability = 43.8 minutes/month or 8.76 hours/year of downtime
- 99.99% availability = 4.38 minutes/month or 52.6 minutes/year of downtime
- 99.999% availability = 0.44 minutes/month or 5.26 minutes/year of downtime

If you think can find a computer system that is available 99.999% of the time, I have a nice bridge in Brooklyn that I would like to sell you.

Downtime – there are two kinds of downtime: planned and unplanned. An IT department takes planned downtime to do upgrades, install patches, replace hardware, and other tasks. Unplanned downtime occurs when you have major and unexpected system failures. A highly available system is one where the unplanned downtime is very minimal. To achieve this, you need a combination of good planning, good hardware, systems to monitor and alert operators when computers fail, and most importantly, you need a well trained and responsive IT staff.

Service Level Agreement (SLA) – A service level agreement is that part of a service contract that ascertains a certain level of service. An "uptime agreement SLA" specifies normal business hours, response time for system outages during business hours, response time outside of business hours, and planned

JDEtips Journal January/February 2007 Volume VIII Issue 1

time for system updates. At Acme we have a two-hour response time SLA agreement. That means that an IT professional will respond to a Severity One outage (the system is down and multiple users are affected) within two hours. Usually SLAs do not specify the allowable down time of the system, they just specify that an IT professional will respond within a specific period of time. In practice, there is a great deal of pressure exerted to diagnose the problem and restore the system as fast as possible.

Disaster – In the IT world, there are different types of disaster that you need to plan for: internal issues, external attacks (a.k.a. "hackers" or viruses), hardware or software failure, natural forces, and "acts of God". Most companies, even large misfortune 500 companies like Acme, will make plans for everything short of an "act of God." Disasters of that magnitude are so prohibitively expensive to plan for that most companies will not make that kind of investment. Let's focus on designing a solution to get past a major loss of hardware due to a natural disaster like a fire or a prolonged power outage.

JDETips – Thank you for the explanation of the terminology, can you tell us a little bit about your architecture?

Yosemite Sam – With pleasure. We have two data centers, located two miles apart. That makes things close enough for the staff to manage both centers but reduces the odds of a large scale natural disaster taking out both centers. We spread our servers across both data centers, making sure the mission critical applications like JDE can survive an outage of one data center or the other. Our database/enterprise server is a Windows 2003 cluster, with one node in the primary data center, and the other node in the Disaster Recovery (DR) center. The servers are high end HP servers with a high degree of fault tolerant components. The C drives on the servers (which are a redundant pair), are local to the server. The rest of the data drives are located on the Storage Area Network (SAN).

We spread our servers across both data centers, making sure the mission critical applications can survive an outage of one data center or the other.

JDETips – Time out, what is a SAN?

Yosemite Sam – A SAN is a remote storage system consisting of a large grouping of disk drives and disk controllers that is detached from the servers. A SAN can host data from a large number of servers. Studies have shown that there is a large amount of unused disk space in a data center where all of the servers use locally attached storage. In a SAN, the storage is shared amongst a large number of servers.

Our SAN is replicated in real time to a secondary SAN in the DR data center. The goal of disaster recovery is to place copies of the data outside the radius of an anticipated threat. The two SANs communicate using a Fiber optic channel. Fiber optic channels can be as long as a company likes or can afford. In the US, our two data centers are two miles apart. In South America, our data centers are over 200 miles apart. Prior to the advent of Storage Area Networks, we were limited to at most a few meters of distance between the server and its storage. That is insufficient to protect against a disaster.

In addition to the real time data replication, we are also able to "snapshot" our data. That means that we are able to "take a picture" of the complete database at a given moment in time. This "snapshot" happens almost instantaneously and gives us a quick way to recover a database server in case of hardware failure. The "snapshots" are very disk space intensive, so only one or two "snapshots" are retained on the SAN.

We have close to thirty other JDE servers. We evaluated each server to decide how important that server is to our continued operation. If it is an important server or service, it has a counterpart in the DR data center. For example, in a disaster, we would not be building packages and deploying ESUs, so we do not have a second Deployment Server. We have two batch servers, one in each building. Fifteen of our terminal servers are in the main data center, five are in the DR center.

Protecting each data center are large UPSs that can run all of the servers in the data center for up to an hour on battery. In our primary data center, we have a large diesel generator that kicks on automatically if the power goes out for over a minute. For cost sake, we did not opt to install a generator for the DR site. If there is a prolonged power outage, we gracefully take down the servers in the DR room and work with