ArcBITS Newsletter

ArcSys Hot Tip

In the Company Builder screen (CB) there is filed called "Data purge time in years". You can choose a number between 2 and 10. Once set, a super user can type in the command EXPUNGE from the menu. This allows you to run the process in either a Trial or Final mode. Trial will show you how much data would be deleted. Final is the real thing. Once done, the data is gone. Forever. Zilch. Nada. A report will be produced at the end showing how many records and bytes were purged from each file. Consider running this process on a yearly basis. Use with caution.

"First the wind started to blow, and now this."

The Northern Utah Wasatch Front is accustomed to strong winds which usually come in April and May. But September proved to be out of the

Photo: Saige Miller

"norm" as a cold front moved from Canada down along the east side of the Rockies. The high pressure system sitting over Utah caused the ideal condition where steady hurricane forces caused extensive damage to property. Temperatures plummeted 50 degrees and 180,000 power customers lost electricity. This, of course, included ArcSys customers.

Planning for a potential disaster is just smart business. "If you fail to plan you plan to fail."

If you work from home, you are relying on its Internet connection as well as the connection at the server. Two different networks—two weak links in a chain. On the "home" end, do you know how to utilize the WIFI hot spot connection on your phone?

If a known storm is pending, consider printing out schedules for the next day or two. Maybe even add printing the last visit notes for patients.

Check the battery backup on your server.

Make certain that the nightly file save is running. Red Planet does lots of checking to make certain it is running as well as alerting you. But, funky things can occur and if you fail to read the warnings, your site is at risk for data loss.

One solution to the latter problem is to consider using a server service such as Fiber.net. ArcSys (as well as several of our clients) utilizes this business for its data processing needs. This hardened site has standby diesel generators and provides 99.99% uptime. When considering your next server upgrade, this may be the solution for your site, too.

MULTIVALUE
NoSQL for BUSINESS

Lessons Learned

There have been several lessons learned during the past 40 years while being involved with MegaWest Systems, Companion Technologies and ArcSys. In this final issue of publication of *ArcBits*, we'll take a trip down memory lane.

The first lesson: *If you sell it, you still own it.* The early days of insurance processing meant printing paper claims. These weren't just single sheets. Nooo. These were multi-part forms because a copy needed to be kept by the office—for filing. The "big" thing of the day was to have printers that could print at least 300 lines per minute (about 5 claims). We thought it would be swell to sell not only the printer but the paper claims, too. So we bought (I don't remember how many) boxes of claims. The biggest problem with printers is that they would jam. A printer had to have a heavy duty motor to pull and feed paper through. As it turned out, the claims that we had purchased had this thick glue which was used to bind the multi-part form. It made the forms so thick that it was hard for the printers we sold to pull the claims. Yikes. Which of the two do we refund to the client? Faulty claims or insufficient printer? We ended up getting out of the claims business, refunded the clients and learned our first lesson.

The second lesson: *Pursue multiple solutions simultaneously and never single thread.* The critical elements here are time and resources. Most people try to solve a problem with solution A. When that doesn't' work, the proceed to B, C and so forth. In the 80s and 90s we were selling a lot of minicomputers and when they crashed was rarely convenient. The usual solution was to contact hardware support who would diagnose the problem and repair on-site if they had the spare part. If not, an order would be placed for the part(s). In the mean time, the clock is ticking. No claims are being generated or appointments made. Now the spare part arrives, installed and it doesn't fix the problem. Back to square one. Yikes. The lesson learned—while waiting for spares—was to take the last backup tape and load on another system. Depending on the physical size and capacity this became quite a challenge in the early days. We did this recovery procedure 3 years ago by pulling the last backup file over the Internet to our server in order that one of our clients could run their payroll.

The third lesson: *When they client says they encountered a problem, believe them 100%.* The challenge here is that the problem can rarely be replicated. It is to easy to dismiss the client as being out in the night. You just have to dig deep and re-examine every program and block of code to see if a mysterious event could occur under the right circumstances. We continue to face this challenge today.

The fourth lesson: *Keep a log of everything.* The early days of MegaWest were always full of surprises. The client would call and describe a problem. You would research and fix things. A couple of months later another client would call with a similar problem. You would research and fix things—again. With a log of these events, you don't have to re-invent the solution. But, you must be diligent in keeping the log and describing things consistently. ArcSys has such a data base of 62,000 events covering 20 years. Further, you need to have a log of every event or transaction that is taking place and an easy way of seeing the log. This has become even more apparent with the use of services like Updox where the events are outside your realm.

The fifth lesson: *It takes 3 times the work to undo something.* A simple example is adding a charge to a client. Entering a payment requires the same amount of work but now it has to be applied to the charge and then determined if the amount remaining is in suspense or private. See? Lots of work. In the old days with green screen displays it was easy to print a character on the screen. If the user pushed the backspace key, then you had to move the cursor to the left, print a blank, move the cursor to the left so that it was positioned to receive a new character to be entered. See? Lots of work. If you think about your daily work, you know the hurdles you jump through when you mistakenly file something in the wrong folder.