



WORK FASTER — WITHOUT A HARDWARE UPGRADE

PURGE AND ARCHIVE WITH INTEGRITY

by David Shea, DCSoftware, Inc.

Yes... you CAN do it

Your machine is slowing down. There are two ways to speed up processing — get a bigger machine or get rid of some data. Getting a bigger machine costs real money. Getting rid of some data is complicated due to the integrated nature of the files. At first, adding disk space provides an acceptable performance improvement at a reasonable cost. Eventually, however, the rising slope of the “data volume versus dollars required to process it” curve backs you into a corner — more hardware or less data. Pick one. Eventually, most shops will want to purge.

Purge functionality included with PeopleSoft

Standard PeopleSoft provides some purge functionality. The programs tend to be fairly straightforward (i.e. simple criteria), limited in their analysis capability, and have a short “reach” (i.e. they tend not to get data from related files). For example, purges exist for accounts payable and accounts receivable. They focus on the individual document level. If the voucher is old enough, if it’s closed and if various balances are zero, it can be purged. But the purge ignores relationships with other documents (such as payments). The GL purge operates at a fiscal period level. Given a fiscal year and period, the standard PeopleSoft purge will summarize prior period GL detail and then purge the detail. But — it ignores the fact that some other modules (like AP) might still need that detail. These purges will reduce the size of your files, but at a potentially high cost — breaking data integrity.

Data Integrity — Definitions

Data integrity is a key consideration for any purge and archive project. But first, there are several definitions of data integrity to consider:

1. Textbook — headers and details. If you have an order detail, you should have an order header. Purging headers without purging details or vice-versa would be at the very least “poor form.”
2. Within functional areas — ex: vouchers and payments. Some documents in a functional area relate to others.

It would be unwise to purge vouchers from Accounts Payable without also purging the related payments. The reverse is also true — it would be unwise to purge a payment unless all of its related vouchers will also get purged.

3. Across functional areas — PeopleSoft is an integrated system. You cannot always draw solid circles around modules. Purchasing is expecting to find accounts payable data. Accounts payable is expecting to find GL data. In many cases, purging data from one module without consideration of the related modules will result in problems with various programs within the software.
4. Batch integrity — some functional areas within the software group data into batches. GL, AP and AR data are obvious examples. Purging some records in a batch but not the entire batch should be avoided.

Data integrity — textbook vs. PeopleSoft

The first definition of integrity — headers and details — is conceptually easy. We shouldn’t purge an order unless all the lines are ready to purge. Once the decision is made to purge an order, the related child files can be easily brought along.

The second and third definitions — relationships across document types or across functional areas — make purging an interesting challenge. We can look at an individual voucher in Accounts Payable and decide if it, by itself, is ready to purge. But to be safe what we really need to know is if the voucher and any related document are ready to purge. The killer is the “any related document” part of the equation. One voucher can be paid by more than one payment. Each payment can pay multiple other vouchers. Each of those other vouchers can be related to still more payments. And so it goes — for who knows how many iterations. We have now complicated the purge decision-making process considerably.

The fourth definition — batch integrity — is another tough one. The concept of batches is prevalent in GL, AP

and AR. It can be extrapolated to the Cardex file, since daily GL batches of inventory activity originate back in the Cardex. It's a stretch, but you could extrapolate from the Cardex back to the source documents — the work orders and purchase orders — and carry the batch concept even further.

Practical considerations for data integrity

Now, we have all these competing files and integrity objectives swimming around in our heads — orders, receipts, vouchers, payments, general ledger, cardex, batches, chickens, eggs. Where do we start? We can start “upstream,” with the source documents — the orders. We can start “downstream” with the general ledger. We might ponder starting mid-stream (cardex) but that's doubtful.

Given all the competing (and sometimes conflicting) factors to take into consideration, for sanity purposes we need to pick a starting point and a general approach to purging. We really need to draw some circles around the data.

Integrity — drawing the circles someplace

Purchase orders lead to receipts. Receipts lead to inventory and payables. Inventory leads to component issues to work orders, scrap and cycle count transactions. Payables lead to checks, and all of the above lead to general ledger activity. Purchase orders span fiscal periods. Purchase order and inventory data get assigned to GL batches by virtue of the calendar, getting mixed in with lots of other transaction activity for a given day — including that horrendous return-to-vendor transaction for a purchase order that was closed three years ago (we've all seen these, right?). Now, that three year old PO is related to current activity.

At one extreme, take the reality of the data, impose the file inter-relationships on it, then apply the batch constraints. If you pursue theoretical perfection, you'll spend years and a ton of money developing your purge approach and eventually find the single record that's eligible to purge. Have the record embroidered on a T-shirt and declare your project a success.

At the other extreme, put the blinders on and ignore data relationships and batches. Get rid of a ton of data and spend inordinate amounts of time dealing with integrity issues in your data.

The trick is to find the happy, practical medium.

Over the last three years of developing purge functionality for PeopleSoft clients, we (my implementation partners and I) have learned a great deal

about what makes practical sense — what data relationships are reasonable to consider and also in what sequence various functional area purges should be run. There isn't enough space to get into all the details here. This knowledge has been distilled into the ARCTOOLS/400 Module for PeopleSoft (see next page for more information). The goal is to purge and archive the data with integrity. So — the data is taken out of the production environment in a sensible manner, and it's subsequently added to the archive environment in a sensible manner. For example — the A/P purge gets not only F0411/13/14 data, but also the related F0911 data, and it maintains batch integrity. The importance of this integrity will become clear later.

The Ivory Tower vs. the Real World

Any purge and archive solution needs to be flexible and understandable. Solutions can be developed in the Ivory Tower and work just fine on Vanilla Data. The Ivory Tower with the Vanilla Data doesn't have the same issues and problems as the Real World. The Real World can have uninitialized fields, orphan records and badly converted legacy data, just to start the list. A purge process that is enclosed in a black box with no visibility into how it works and no method for tweaking the process inside the box is risky. It may work in some scenarios, but may have serious trouble in others. A purge process needs to be straightforward and understandable. It should be easy to tweak and easy to emulate for other functional areas or other software platforms.

Accessing the old data

The first question I get asked is, “How do I get rid of the old data?” The second question I get asked is, “How do I get it back?”

Keep in mind — we're not getting rid of the data. We are getting it out of the way. Also bear in mind that we're getting it out of the way with integrity. Data that was supposed to stay together was archived together. The data that is now in the archive should support many of the report and inquiry functions within the software.

There is a spectrum of functionality for accessing archived data. At the low end is the ability to pull up records using Query, SQL, MS Access or the like. At the high end is the ability to see the archived data along with the live data in all the production programs as if it were never purged. Both ends of the spectrum are possible, as is every point in between. Of course there is a cost associated with reaching the high end. Somewhere along the line is the point where 20 percent of the cost nets 80 percent of the functionality.

I have had customers simply move data off to tape. Others are satisfied with the “low end” of the spectrum

— either leaving the archive files on the AS/400 or moving them to the network and an MS Access database. Others have substantial data warehouses on other platforms. At least one customer has implemented the “high end” approach for at least some of their programs. The most “bang for the buck” (the 80/20 point) can be realized by building a PeopleSoft environment with your archived data.

Remember that the data was purged from production with integrity. It was also added to the archive with integrity. In many cases, the archive environment can be turned into a PeopleSoft inquiry only environment with relatively little effort. So, if a user wants to pull up a six year old voucher and doesn't find it in production, it must be in the archive environment.

Bringing data back vs. accessing the archive

Technologies exist for “hierarchical data storage.” New data is kept in live. After X months it's moved to an online archive. After Y months it's moved to near-line storage. After Z months it's sent off to tape and into the salt mine for posterity. Some of these technologies tout the ability to bring data back into the production files. While this may sound enticing, be very careful. Bear in mind that we just went through a tremendous effort to purge and archive data from production with integrity. That integrity spans files, functional areas and batches. In many cases, we summarize the archived data to keep production in balance. Do not start pulling data off of some near-line data storage device and plunking it back down into production. Data integrity for production will be compromised, and you may now be in a situation where the data is back in live and also in the archive. Since it is old data, chances are that the data will be archived again with the next round, resulting in two copies in the archive.

Archive the data and leave it there. If you need to get at the old data, access it where it sits — in the archive.

Summary

Data archiving sounds fairly straightforward when you first start thinking about it. As you proceed further down the road and recognize the realities of file relationships, data integrity and batches, you start realizing just how difficult a task purging really is. PeopleSoft has some purge functionality, but many customers have discovered that more is needed than what is provided in the standard software. Some customers have tried tackling this on their own, with varying levels of success or failure.

DCSoftware has spent years developing a purge and archive approach for PeopleSoft that utilizes ARCTOOLS/400 as the purge engine. The result is a product that provides the ability to purge and archive with integrity, without locking users out of the system, without adversely impacting other work on the system, and on a flexible time schedule.

Discover how to purge and archive from PeopleSoft with integrity. Visit DCSoftware at www.arctools.com, by email at info@arctools.com or call 1.877.ARCTOOLS.

About David Shea

David Shea has been working in the information industry for more than 20 years. He spent six years with one of the largest systems consulting firms in the world, working in defense and commercial manufacturing environments on mainframe and midrange platforms, and has more than 18 years experience in ERP environments. He was the IS Director for various companies in the medical products and biotech industries, and is the founder of DCSoftware and the chief architect of ARCTOOLS. He can be reached at 508.435.8243 or by email at dshea@arctools.com.

About DCSoftware

DCSoftware, Inc. was founded in 1996 to provide ways to improve performance in AS/400 environments. The ARCTOOLS/400™ product allows purge and archive functionality to be developed quickly and easily. The ReorgWizard™ product allows files to be reorganized without the extended file locks required for “standard” reorgs.

ARCTOOLS was created to fill a gap in many software application packages — to allow purge and archive functionality to be developed and maintained as easily as writing a Query, while still utilizing the safety and flexibility of an RPG program. ARCTOOLS allows “standard” purge functionality to be developed that can subsequently be easily tweaked for individual customer requirements. It runs without locking users out of the files, can be stopped and restarted as needed, and can be “throttled” to manage the performance impact to other work on the system. ARCTOOLS, the award-winning purge and archive software, is in use around the world in numerous software environments including PeopleSoft, BPCS, MACPAC, Infinium, JBA and custom software shops.

The ARCTOOLS/400 Module for PeopleSoft was developed in collaboration with PeopleSoft in Denver. The goal was to create a “standard” collection of purge and archive routines for various functional areas in the PeopleSoft software using ARCTOOLS as the engine. Each purge and archive routine is intended to be as easy to understand as possible, and to be easily modified to individual customer requirements. Documentation is extensive. DCSoftware has partnered with several implementation partners with offices in the United States, Canada, United Kingdom, France, Poland, Australia and the Far East. See www.arctools.com for more details.