Double Entry Stock Keeping

TECHNOLOGY · INDUSTRY · RETAIL · LIFESTYLE

FROM THE EDITOR

▶ After the doom and gloom that has afflicted the IT industry in the last three years, things seem to be looking up in 2004. Even though the spectre of a massively devalued dollar looms as the US treasury desperately seeks to export its gargantuan budget deficit pain elsewhere, we should see some real growth in our sector this year.

Things are certainly looking up for IBM which, as *iSeries NEWS UK* went to press, reported that fourth-quarter earnings had more than doubled and had surpassed Wall Street expectations, with revenues of $2.7bn. Of course, those earnings would not have been possible without the weak dollar, but as Big Blue is a crucial industry bell-wether, the news can only be welcomed. But why is the company so bashful when it comes to the iSeries when it releases these earning reports?

'Revenues also were higher for iSeries midrange servers,' said the official statement. So that's clear then. Although we report some more detailed figures on the iSeries in Newswatch this issue, Big Blue often shrouds iSeries statistics with mystery. However, although some of its figures (or the lack of them) can be misleading, my investigation this month into the platform's status in the important Asia-Pacific region points to a bright future in that part of the world.

Elsewhere in this issue, John Amans looks at how Microsoft's .NET approach differs from J2EE-based environments like WebSphere and John Dreyfus reports on an example of how e-business should be done. And after a long hiatus, during which time he has been contributing some great articles for our parent magazine *iSeries NEWS International Edition*, Paul Morris returns to our pages with another of his unique takes on business processes, with the intriguing concept of using the principles behind double-entry book-keeping to manage your inventory.

Like many of our contributors, Paul bases his articles on real-life business scenarios, and we are always on the lookout for similar real-life concepts from our readers. If you have a good idea for an iSeries-related article, do get in touch with me at seamus@iseriesnetwork.com

Seamus Quinn, **Editor**

DOUBLE-ENTRY

PAUL MORRIS EXPLAINS A MODEL APPROACH TO INVENTORY MANAGEMENT

'TRIGGER
PROGRAMS
PROVIDE A GOOD
METHOD OF
CONTROLLING
LINKED UPDATES
TO MULTIPLE FILES'

> In this article I am going to discuss how the principles of double-entry book-keeping can be applied to stock-keeping records and what advantages this may bring. I admit I am not an accountant; in fact, those accountants who have had the misfortune to work with me will agree that my grasp of accounting principles is basic – one phrase of mine used in discussions with them is: 'Just tell me what bucket to throw it in and is it plus or minus?'.

However, there are some principles that I like about accounting systems, the main one being that of double-entry book-keeping. What I wish to discuss here is the application of this principle to stock-keeping records within an inventory management system.

To start with, if you are a creating a simple journal within a typical general ledger (GL) system, for each transaction two records are written; the first taking value out of one

STOCK-KEEPING

account, the second putting it into the other account (I said my accounting is basic). The value of the journal does not matter; it can be a few pennies in a rounding account, or large sums in a sales account. What does matter is that two transactions get written that are complementary to each other. Another point to note is that even if money is coming in to or going out from the company, the two transactions still get recorded.

ONE FOR ALL

One stock control system that I experienced used 'single-entry book-keeping', in that only one movement record was written for the posting of a stock movement, the 'to' and 'from' sources being recorded on just the one record within the stock control system. Other systems used two transactions for internal movements, but only one at the point the stock left or came into the company. Just like

a GL journal, a stock movement can be for a stock value of just a few pennies or for a large sum of money, so I believe similar controls to what is provided in the GL should apply. Therefore, I would suggest that two movement records should be written for all the stock transaction, in much the same way as the general ledger.

Let us start by taking the example as shown in Figures 1a and 1b *(overleaf)*. Figure 1a is a simple inventory balance file and has a starting balance of 100 units for part P1 in warehouse W1. In step 1 we do an inventory transfer from warehouse W1 to warehouse W2 of 40 units. Figure 1b shows the transactions in a simple movements file (a date and time, amongst other details, would also be required) and Figure 1a shows the new balances. There is nothing of note in the balance file, but I would like to make the following points about the transactions

as shown in Figure 1b:
- the quantities balance each other, so the net effect of inventory is zero
- the transactions are tied to each other by a movement reference number
- the resulting balance of the inventory is reflected in the transaction. This makes for a very tight audit trail in that the resulting balance of the last entry for a part should reflect the current value in the inventory balance.

You can see it becomes easy to construct all manner of enquires and queries over these movement records – my rule of thumb is 'could an average user write a query over the file and get meaningful data?'.

If we now consider the movement of stock in and out of the company (such as purchases and sales), then the analogy with the GL breaks down at this point because accounts are separated into balance sheet and profit and loss, but there is no equivalent concept ▫

FIGURE 1A – INVENTORY BALANCES

WAREHOUSE	PART NUMBER	STOCK BALANCE	STEP
W1	P1	100	START
W2	P1	0	START
W1	P1	60	1
W2	P1	40	1
W1	P1	50	2
W2	P1	25	3

FIGURE 1B – INVENTORY TRANSACTIONS (MOVEMENTS)

WAREHOUSE	PART NUMBER	MOVEMENT QUANTITY	FROM/TO WAREHOUSE	TRANSACTION REFERENCE	RESULTING BALANCE	STEP
W1	P1	-40	W2	TR1	60	1
W2	P1	40	W1	TR1	40	1
W1	P1	-10		CO1	50	2
W2	P1	-15	**	CO2	25	3
**	P1	15	W2	CO2	0	3

in stock records. So let us now consider what the options there could be for the stock records.

The first option is that a single movement record is written. This is shown as a sale in step 2 with a sale of ten units from warehouse W1. This is acceptable, but does break the principle of being able to view transactions from either viewpoint.

Another option is to write the complementary record to represent the 'outside world'. This keeps to the concept of matching pairs of transactions (our double entry). This is shown in step 3 as a sale movement of 15 units from warehouse W2. Here a warehouse code of '**' is used to represent the outside world.

I do not like this second option for two reasons. One, it does not bring anything to the party – there is no real benefit, and, two, a simple query cannot be written that could answer a question similar to 'what have been the total movements in or out of the company this month?'. An artificial selection to exclude a specific warehouse code would be required, otherwise the sum of the movements would always be zero.

At this point I would go for the first option as there is no knowledge, nor interest, in the product once it leaves the company. Incidentally, I

have designed a successful inventory system (and I am aware of others) based on this concept, with one of the benefits being that the auditors liked it for the tracking it gave them (make friends with the auditors is another of my rules of thumb). I must admit, though, that I have always felt a little uneasy about the one-sided nature of these transactions.

But is there a third option if we can devise something equivalent to the GL? Would there be any benefit if we could? Let us now explore this concept.

If we look at the stock values in the GL, we see the monetary value of it is held in one or more balance sheet accounts. As we receive stock, the value in these accounts goes up. As we sell stock, the value of these accounts falls. The balancing entry of the double entry is equivalent to profit and loss accounts. Can we translate this to an inventory equivalent? Look at Figures 2a and 2b – I will process the same transactions as before.

In Figure 2a we can consider the combination of warehouse and part number as being equivalent to a GL account and that all our stock records are 'balance sheet'. So our starting 'value' at the beginning of the month is 100 in the W1 P1 'account'. We transfer this to the W2 P1 'account' as shown on step 1, then

there is no affect in the total value of the 'balance sheet' stock holding and the picture is not much different from before — although I am now showing month-to-date (MTD) figures, much as you would see in the GL. If your stockrooms are in different balance sheet accounts you could see a similar transaction reflected in the GL.

For step 2 let us sell ten of the product P1 from warehouse W1 on a customer order. On the inventory balances file we now have a P&L 'account' for customer sales. Figure 2b shows that we get the two movement records taking the quantity out of the stock 'balance sheet' and into the sales 'P&L'. Figure 2a shows that the quantity has been removed from the stock 'balance sheet' and is now in the stock 'P&L'.

But what does all this effort give us?

- our query user can still write a simple query, especially if there is a logical file selecting only the balance sheet accounts (instead of omitting a list of one or more artificial warehouses)
- we now get the kind of view of our stock that accounts get of the GL without having to select and sum transactions. Questions like 'What sales for product P1 has there been this month?' can instantly be answered without running queries or hav-

WAREHOUSE OR P&L TYPE	PART NUMBER	TYPE	STOCK BALANCE	MTD	STEP
W1	P1	B/S	100		START
W2	P1	B/S	0		START
W1	P1	B/S	60	-40	1
W2	P1	B/S	40	40	1
W1	P1	B/S	50	-50	2
SALE	P1	P&L	10	10	2

FIGURE 2B – INVENTORY TRANSACTIONS (MOVEMENTS) – DOUBLE ENTRY

WAREHOUSE OR P&L TYPE	PART NUMBER	MOVEMENT QUANTITY	FROM/TO WAREHOUSE	TRANSACTION REFERENCE	RESULTING BALANCE	STEP
W1	P1	-40	W2	TR1	60	1
W2	P1	40	W1	TR1	40	1
W1	P1	-10	SALE	CO1	50	2
SALE	P1	10	W1	CO1	10	2

ing specialised enquiries

- there is scope for providing the equivalent of a 'trial balance'. Had I included the take-on transactions in these examples, then the sum of the P&L and balance sheet accounts would be zero – a good audit check
- we can build on to this concept the same sort of procedures for 'month end' clearing the MTD quantities, rolling up the YTD, etc
- we can tie the stock records closely to the GL to provide a tight audit trail.

The design of these stock 'accounts' is a bit like designing a chart of accounts for the GL. The GL can have many accounts trapping a low level of detail or few accounts holding a smaller breakdown of information. In the same way, we can provide for purchases and write-offs, or possibly sales by warehouse, even sales by customer should we desire it. The users would have to decide on what level of granularity is required. One advantage is that this gives a straight view of the transactions, clear of any journal adjustments (in other words you get to see the raw data before the accountants have messed around/applied journals to it).

To finish this discussion, I would just like to mention a few thoughts on the physical implementation, irrespective of the option chosen.

As I implied in a previous article on accounts receivable transactions (see iSeries NEWS UK, June 2003), trigger programs provide a good method of controlling linked updates to multiple files. A trigger program would be attached to the movement record that would update the corresponding balance record to ensure the figures remain consistent.

As the two transactions are matching pairs, it would be nice to be able to write one and have a trigger program automatically write the other. But this would call the trigger program again, which starts getting a bit messy. Programming code is beyond the scope of this article, but an ILE procedure could be written (preferably placed in a service program) so that by calling the procedure once, it would cause the two records to be written. I must admit that I do like these modern features of OS/400, as they allow a lot of business logic to be written once and either be called implicitly (like a trigger program) or explicitly (like a procedure).

Is all this worth it? I cannot answer that question for your application or organisation, but I believe it is a concept worth considering. The idea does provide another view of the stock data, before accounting adjustments are made. ❶

PAUL MORRIS IS THE DIRECTOR OF WELWYN SUPPORT SERVICES LTD, A SMALL SOFTWARE HOUSE OFFERING SYSTEMS AND PROGRAMMING SUPPORT TO THE iSERIES COMMUNITY IN THE UK.
www.wssltd.demon.co.uk
paul@wssltd.demon.co.uk