

SENIOR CONSULTANT

The Voice of the Investment Management Consultant

Not A Crisis but a Crossroad Part II: Defining the Challenge in Processing Separate Accounts

Bevin Crodian

In [Part I of our series](#), we addressed the question of whether or not the separate account industry was facing a crisis in its operations. Clearly, the industry is not in crisis, but equally as clear is the observation that its growth is hampered by operational inefficiencies. If there is a crisis here, it is more in the lack of commonly accepted terms and definitions for discussing the process of managing separate accounts. This is especially true when considering operations in the middle and back office. It should be obvious that if we can't engage in a coherent dialogue about these issues without pausing every few moments to define our terms, we have a problem. In response to this, we defined three functional criteria and give an example of each:

1. **Complexity.** Complexity is represented by the ability to account for more sophisticated securities like derivatives, ordinaries and asset backs.
2. **Customization.** Customization is shown by the ability to block transactions on specific securities and to manage portfolios for tax efficiency.
3. **Scale.** Scale is shown by the ability to process many thousands of transactions quickly and accurately.

The priority of these criteria will allow us to describe operations in the middle and back office. For what is commonly known as the "institutional" application serving the tax-exempt client, the functional priority for operations is: (1) complexity, (2) customization and (3) scale. For the "private wealth" application serving the high net worth client, it is: (1) customization, (2) complexity and (3) scale. And, for the "wrap" application serving the mass affluent client, it is: (1) scale, (2) customization and (3) complexity.

We also asserted in Part I that nearly every channel or client segment will require, in varying degrees, a

common set of middle and back office operations. At a high level of description, the operations for the middle office are pre- and post-trade compliance, blocking of trades, trade execution and trade allocation. For the back office, the levels of operation are: account opening and termination, portfolio accounting, corporate actions, reconciliation, performance measurement, billing, reporting and data warehousing. These operations often require the use of different and sometimes unrelated technologies. Ideally, these technologies

need to be integrated into one unified platform that will support process efficiency across all distribution channels and client segments. Whether or not what follows represents an overly optimistic wish list will be determined as we proceed; however, our discussion will focus on what is available in today's immediate technological environment – which is defined here as technology capable of a successful installation in the year 2004. Also, for the sake of simplicity, we will ignore, for the time being, the question of outsourcing. Here is what the platform could look like.

The Platform

If the platform is to be capable of adequately supporting different clients and distribution

channels, then it has to be able to meet, both simultaneously and equally, all the criteria of complexity, customization and scale. As is often the case, we can better understand a problem by analyzing it from different perspectives. Let's try these: design, foundation, software application, and integration.

Design

To start, we should acknowledge that very few persons managing in the middle and back office have

**IF THE PLATFORM IS TO
BE CAPABLE OF
ADEQUATELY
SUPPORTING
DIFFERENT CLIENTS
AND DISTRIBUTION
CHANNELS, THEN IT
HAS TO BE ABLE TO
MEET, BOTH
SIMULTANEOUSLY AND
EQUALLY, ALL THE
CRITERIA OF
COMPLEXITY,
CUSTOMIZATION AND
SCALE**





been given the opportunity to build their ideal platform ... dream maybe, but not build. Consequently, design usually occurs when a COO, who is laboring under the constraints of a legacy system, is given a specific mandate of adding some new functionality, like billing, for example. In this case, design work is the creation of a limited functional specification that is (with a hope) developed adequately and then (with a prayer) integrated into the existing system. In the general course of business, these are necessary endeavors, of course, but they can lead to a truncated, if not despairing, view of middle and back office operations.

To build the ideal platform, you must first consider the basic technological foundation needed to support the requirements of complexity, customization and scale. However, the identification of the appropriate technological environment and its components is necessary but not sufficient for building a platform. As anyone who has had to assemble a doll house on Christmas morning can attest, having a list of the necessary components can still place you a long way from a successful implementation. You also require some working knowledge, or vision, of what the final product is supposed to accomplish. Clearly, the design of our operational platform is an iterative process that will require constant testing and modification, but it also requires some initial and broad understanding of the intricacies of the investment process – or workflow. Focusing on the requirements of complexity, customization and scale can keep the designer on track.

Foundation

The foundation for a successful operating platform is a four-legged stool supported by its hardware, operating system, database system and data transmission. The hardware must be sized to handle the platform. Obviously, some knowledge of the potential scale and complexity of future transactions is necessary to make this decision. Also, a feasible plan for expanding computing capacity must accompany the original hardware decisions. While computing capacity is determined by the laws of physics and economics, the decisions about the operating system and database are driven more by

convention. An operating system is a program that manages the basic hardware and software resources of the computer, like the processor, memory, disk space, etc. Regardless of the inroads being made by Linux, Windows, Unix or MVS (IBM mainframe), operating systems are still the most commonly used to process back office functions. The database is the system that allows you to store and retrieve data. Relational databases, as opposed to the older hierarchical versions, are now the preferred structure. Currently, Oracle is dominant in the database business, but other names are DB2, Sybase and Informix. The need for easy modification of software applications requires that the foundation of the platform operate on

**TO BUILD THE IDEAL
PLATFORM, YOU MUST FIRST
CONSIDER THE BASIC
TECHNOLOGICAL
FOUNDATION NEEDED TO
SUPPORT THE REQUIREMENTS
OF COMPLEXITY,
CUSTOMIZATION AND SCALE**

systems in common use. Ease of modification is the key driver of the potential customization of your operations as well as the future technological extension of your platform. Finally, a platform that supports the middle and back office must transmit a lot of data to different participants in the investment process: trading desks, custodians, consultants and ultimately, clients. Clearly, the preferred method of data transmission, today, is to use the web wherever and whenever possible.

**Software Applications:
Build vs. Buy**

Let's forget for a moment the question of what software applications are necessary or preferred for the platform. Let's assume that there is some core functionality like trade and order management that everyone will agree is a necessary component of the platform.

Assuming that this decision has been made, then you are faced with the "build vs. buy" dilemma. But first, let's consider some criteria that may guide us in making our decision. They are cost, time and flexibility. Regarding cost, some range for the required application should have been included in your original design specification. Therefore, the cost problem, although not necessarily easy to solve, should be fairly straightforward. Simply balance the cost of licensing and installing the application vs. building it, assuming, of course, that you possess the resources to make this calculation. Admittedly, this is like saying that to build a pyramid, you need to move a lot of stone, but at least the challenge is clear. When purchasing software, the issue of cost centers less on the initial licensing fee, which is definite, and more on the installation costs, which can be indefinite. Consequently, foremost in your considerations should be the old truism that "time is money." Whether you are building the application yourself or someone else is installing a licensed product for you, development/installation time will greatly impact your total cost.

But let's assume for the sake of our example that you ultimately decide to buy (no pyramids for you). The next criterion to consider is the flexibility of the application. Remember, most applications are built for a specific range of tasks that may be similar, but not identical, to the requirements of your specific market or distribution channel. Consequently, regardless of the software application you choose, it will often require some, possibly significant, modifications before it can support your business. The extent of this modification and your ability to achieve it with your existing IT staff or consulting relationships is clearly a concern when deciding what to buy.

Integration

Earlier, we discussed the foundation for the platform and the challenges of picking a particular software application. But what are these applications? At a gross level of description, we can identify the need for software applications to support the following tasks:

- Tax-lot accounting
- Trade and order management
- Performance measurement



- Reconciliation (electronic and/or manual, depending on the capabilities of the custodian)
- Straight-through processing of new accounts
- Data conversion (i.e., the process of moving existing business from your prior system onto your “new” platform)
- Interfaces to trading desks and custodians
- Data warehousing and reporting

We also indicated earlier that identifying the components required for the platform was a difficult and necessary step, but not sufficient in itself. Here is why. All the components have to be able to work together efficiently. Having a highly functional tax-lot accounting system is fine, but if it is operating on a database that is incompatible with your trade and order management system, it will be an awkward fit. We said earlier that the design process was iterative, and this is just another example. The software applications have to first support their operational tasks, but it may be of even greater importance that they allow integration with other components of the platform. For our purposes, integration is defined as the ease in which different software applications can communicate and function together. This is usually accomplished through a messaging interface and a shared database environment (i.e., you really do NOT want to maintain two copies of all of your data – holdings, tax lots, portfolios, prices, security characteristics, etc.). Although not a small task, the creation of an efficient messaging interface and integrated database environment will allow faster processing and more efficient workflows. Here again, your initial design is paramount. Messaging between applications with incompatible databases can be very difficult. Although integration might be seen as the last step from a development sense, its feasibility is determined by the correct design decision – which is one of the first steps in building the platform. Now let’s examine just two examples of how a robust suite of tightly integrated software applications running on a common foundation can provide an efficient platform.

Performance Measurement System. In order to correctly measure performance, the system must be able to easily retrieve information from the tax-lot accounting engine.

Furthermore, both these systems must possess the ability to process transactions on more complicated securities like ordinaries and asset-backs. This is necessary to accommodate the complexity of the institutional business. However, if the performance measurement system is not closely integrated to the tax-lot accounting system, the performance measurement calculation will be slow at best and dysfunctional at worst. This is not necessarily an issue for the institutional business, but in the wrap space, the platform has to be able to handle thousands and thousands of transactions quickly and accurately. One approach that can increase speed is the application of more computing power, but this also increases the cost of the platform. Consequently, close integration

NOW, ASSET MANAGERS CAN STOP DREAMING ABOUT A PLATFORM THAT PROVIDES GREATER FUNCTIONALITY, FLEXIBILITY AND COST SAVINGS. IT IS WITHIN THE FOOTPRINT OF TODAY’S TECHNOLOGY.

between the performance measurement system and the tax-lot accounting system allows the platform to scale the transaction in a cost-efficient manner.

Trade Management System. In a highly functional platform, the trade and order management system must be tightly integrated with the tax-lot accounting system. Moreover, it must also be in synchronization with the new accounts and custodian reconciliation systems. Consider a tax-sensitive client who also wants to apply a socially conscious screen to their portfolio. Here are some of the account specific information that the trade management system will have to access: (1) record of buys and sells; (2) tax calculation method – LIFO, FIFO, HIFO, etc.; (3) date of transaction; and (4) restricted securities. To also avoid errors, your portfolio managers must be working on real-time information while the screen is being applied. For example, they need to be aware of

any intra-day reconciliation adjustments that are being made to the client’s account by the operations staff. In this case, a common database coupled with a well-conceived messaging interface is required to allow the trade management system to communicate easily with the new accounts, reconciliation and tax lot systems. This ease of communication will allow portfolio managers to implement an error-free customization for a socially conscious, tax-sensitive client in their high net worth business, and finally they will be able to perform similar tasks at scale for their wrap business.

Conclusion

Initially, we set out to describe the requirements for a platform for the middle and back office in the separate account industry. It had to provide process efficiency while meeting the functional criteria of complexity, customization and scale. These functions are required to simultaneously support all separate account products: institutional, private wealth management and wrap. We also stated that this should be accomplished with current technology. Appropriately, there has been no mention in our description of any process or software application that is not currently available in some way. Consequently, it is fair to assert that using today’s technology, it is feasible to create a tightly integrated suite of flexible software applications that will provide process efficiency in the middle and back office. Such a configuration provides a core that can greatly accelerate product innovation and cost savings in the separate account industry. Through an open architecture built on a fully integrated relational database, functional upgrades can be included, both quickly and cheaply. More importantly, functional enhancements can be implemented without modifying the entire pre-existing system. Now, asset managers can stop dreaming about a platform that provides greater functionality, flexibility and cost savings. It is within the footprint of today’s technology. ■

About the Author

Bevin Crodian is CEO of Market Street Advisors.