SunGard Trading and Risk Systems 33 St. Mary Axe, London, EC3A 8AA • Tel: 020 7337 6000 • Fax: 020 7337 6010 • Email: marketing@risk.sungard.com

The Data Challenge of Basel II

David M. Rowe, PhD, *Group EVP for Risk Management* and Dean Jovic, PhD, *Managing Director, Basel II*

The biggest obstacle to complying with Basel II is data fragmentation. SunGard's David Rowe and Dean Jovic discuss the problem and propose a practical strategy for meeting the challenge.

The Burden of History

Data fragmentation is an uncomfortable fact of life in every organization. This reflects 25 years of decentralized decision-making regarding information systems. Before the personal computer, data processing was a heavily centralized corporate function. It was characterized by "the guys with the white coats in the glass air-conditioned room with the raised floors." Business users were at the mercy of an elite core of knowledgeable technologists. The result was a high degree of consistency in technology at the enterprise level but great difficulty in keeping up with the needs of individual business units.

This situation began to change with the introduction of the PDP-10 mini-computer by Digital Equipment Corp. in the mid-1970s. For the first time, acquisition of serious data processing equipment did not have to be a multi-million dollar proposition. Mini-computers brought the price of significant computing power within the department managers' budgets, many of who were only too ready to escape the clutches of the "high priesthood" at the corporate data centre.

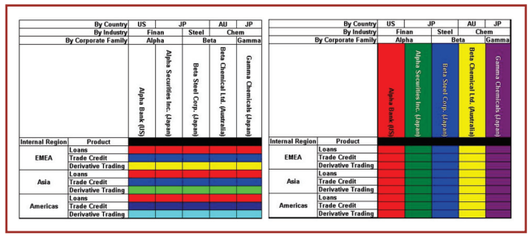
The options for localized computing expanded further with the advent of the personal computer. The trend accelerated when IBM entered the field, giving PCs mainstream acceptability. It is no exaggeration to say that these developments were the beginning of a revolution.[1] The ability to implement quickly new analytical capabilities meant better local decision-making and was the basis for whole new products and markets such as derivatives.

While localized computing brought greater nimbleness, it also had its dark side. Quality control suffered as applications were deployed without the strict quality assurance of centralized computing. Backup and archiving of data were either inconsistent or non-existent. Perhaps most serious, however, even fully accurate and reliable data became scattered across multiple platforms. These data were on different machines (sometimes with different operating systems) in inconsistent formats, with limited documentation and usually with no means of external access.

A firm's technology often reflects the organization's priorities. Throughout much of the past two decades, decentralization and local empowerment have been popular management trends. The needs of individual business units to meet market changes were considered to be of primary importance. Enterprise-wide information was mainly limited to financial reporting and long-term planning. These trends and the availability of increasingly powerful localized technology accelerated the fragmentation of information processing and data storage.

The Rise of Enterprise-wide Risk Management

Risk management has recently moved from a localized to an enterprise-wide function. This reflects the broadening realization that risk is inherently a portfolio concept. To measure risk accurately at the enterprise level requires analysis of not only individual local risks but also how they interact. This does not require simultaneous access to every scrap of data throughout the firm. It does, however, require the ability to analyze a minimum core of information on a unified basis. In addition, for credit risk, the problem is even more fundamental. Even localized analysis for individual obligors often requires consolidating data across regions, products and organizational divisions. The following graphic illustrates the point.



As shown on the left, data consistency tends to run parallel to organizational segments and these generally reflect regional and product categories. Such data include credit exposures to all customers dealing with that product and region. To be sure, data fragmentation is not exhaustive. In the above illustration there is one integrated global system for loans and a common trade credit system for two of the three regions. Nevertheless, it is a rare organization today that can claim full data integration on a single platform for all products across all regions.

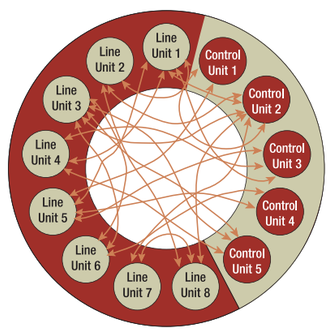
On the other hand, to analyze the credit risk for a single obligor requires a "single customer view." As shown on the right, this demands consolidation of data that cut directly across all the individual systems that are, at least partially, fragmented by region and product. Thus data consolidation is necessary just to achieve an integrated customer view before even considering the analysis of portfolio dynamics in estimating enterprise-wide risk.

Data Integration Strategies

This leads to the question of the best strategy to achieve the required degree of data integration both to achieve best practice risk management and to meet the requirements of Basel II.

Default Solution

The first approach could be called the default solution. It generally occurs in the absence of a corporate strategy for dealing with this issue. It comprises many point-to-point data feeds developed on a bespoke basis as the need arose. The following graphic illustrates this.



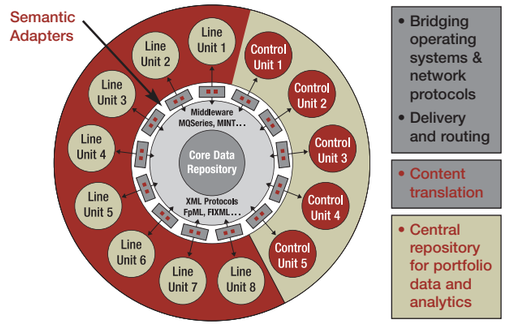
Such point-to-point data links are often the fastest and easiest way to solve an immediate problem. The downside is that there is no consistency in the format and the logic to create recurring data files is usually unique to each instance. In the end, maintaining such data feeds becomes a major hidden (or sometimes not so hidden) cost of keeping the whole process running. As new features are added to existing products, new products are introduced and new business units are added via merger and acquisition, the structure of these feeds needs to be revised accordingly.

It has been estimated that just maintaining such links consumes well over half the data processing maintenance budget of many large organizations. Also, this is inherently a batch update strategy. Having initially solved the data transfer problem via point-to-point file transmissions, it is very hard to move from a batch orientation to a real-time event-driven architecture.

An "Ideal" Strategy

An ideal strategy centers on self-describing messages built on the eXtensible Mark-up Language (XML). Properly speaking, XML is not a mark-up language at all, but rather a meta-language, i.e. a syntax within which a true mark-up language can be developed. Such mark-up languages include the Financial products Mark-up Language (FpML) and the Financial Information eXchange Mark-up Language (FIXML). These define specific semantic content for describing products, transactions and events for a specific business domain.

This approach is illustrated in the following graphic.



The disorganized snarl of point-to-point connections is replaced with a corporate information backbone. There are three broad functions that need to be implemented for this approach to be successful. The first is bridging disparate operating systems and network protocols as well as ensuring that information being transmitted is properly routed and received by the appropriate recipients. This is the task of standard middleware products such as MQSeries from IBM and SunGard's MINT Knowledge.

The second, and more troublesome, requirement is content translation. First, this demands a well-defined mark-up language such as FpML or FIXML. Second, it demands a series of adapters to translate content between the individual local systems and the standard XML-based mark-up language.

Finally, effective portfolio modeling requires that core data be consolidated in a central repository for ease of analysis and assured archiving in a consistent format.

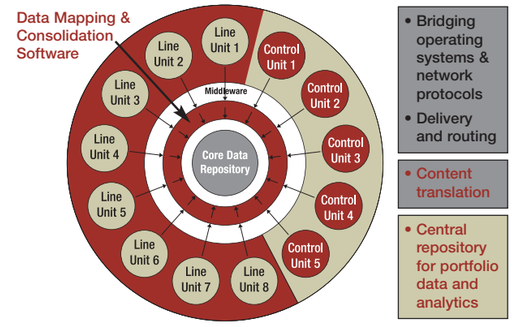
The advantage of this approach is that it is extensible. New features can be added to the mark-up language without disturbing existing messages that do not use these new features. Also, once the adapters are written, a single piece of information can be transmitted to multiple destinations by placing it on the ring in standard form. Furthermore, this approach is inherently modular and lends itself naturally to an event driven-environment.

But there are some serious disadvantages to this approach. First, the industry standard mark-up languages have been understandably slow to develop. This has left many institutions reluctant to forge ahead on their own since they know they will have to make major revisions when an industry standard is established. Second, the process of developing the adapters is a significant investment and these would have to be modified if the structure of the core mark-up language were revised. Furthermore, the payoff from these investments tends to be broadly distributed across the organization rather than accruing mainly to the business units which develop them. A less serious drawback is that transmissions based on this approach are much more "verbose" than with traditional approaches such as fixed format files. The global overhang of unused communications capacity, however, is likely to minimize this problem.

The drawbacks noted here have resulted in only a slow acceptance and implementation of this ideal approach. While we suspect that this will be the standard approach to this problem in ten to fifteen years, it is unlikely to represent a viable alternative in time for Basel II compliance.

A Practical Alternative

But there is a practical middle ground between bespoke point-to-point file transfers and an ideal self-describing messaging environment. This involves inserting a layer of content translation software between the local systems and the central data repository. The following graphic illustrates this approach.



This software layer plays a dual role. First, it provides a visual data-mapping environment that is usable by a business analyst who does not have to be a programmer. It allows the user to define the appropriate correspondence between fields in the remote database (or flat file output from the remote system) and the central data repository. Having defined these correspondences, the software creates a standard translation file to preserve this correspondence.

The second role of the software is to perform periodic transformations and transfers of actual data from the local systems to the central database. For this task it uses the meta data in the translation file created and maintained by the business analyst.

This is an attractive middle ground between the other two approaches: Contrasted to bespoke point-to-point links, it introduces much greater discipline and consistency in the process of creating correspondences between data in the remote systems and their counterparts in the central database. Moreover, changes in the local data formats can be spotted relatively easily and the resulting problems corrected in a timely manner. This ability can be strengthened by defining "sanity checks" on the values of inputs as part of the metadata in the correspondence table. This often allows issues to be trapped in the translation process even if the local change has not been communicated to those in charge of the central data consolidation.

The biggest advantage of translation software relative to immediate adoption of the ideal approach is that it does not require a comprehensive markup language. In addition, the translation tables can be created easier and faster than the semantic adapters. Finally, this strategy can support an event-driven messaging approach to data transfer. This is done by allowing the translation software to read and write information in self-describing messages, typically formulated in XML-based markup languages. Hence it can exist comfortably with, indeed can support, a gradual evolution toward the ideal event-driven approach.

Obviously, an approach to data consolidation based on translation software does not create virtual many-to-many interoperability, which is the ultimate goal of the ideal system. But it does greatly streamline the process of creating and maintaining a central repository of the data needed to perform meaningful enterprise-wide risk analysis. As such, it should be seriously considered as an option for any organization struggling with the data consolidation and analysis requirements of Basel II.

Reference

1. Reportedly a saying among corporate technology staff in the 1980s was, "The users are revolting, in both senses of the word."