Beyond relational databases

Data and analytical fragmentation have been risk management obstacles for decades, but efforts to overcome the problem have been rooted within the paradigm of relational databases. David Rowe argues moving beyond this framework holds the key to success

The past 30 years have seen two major shifts in the prevailing technology paradigm. In the late 1980s and early 1990s, client-server applications became the accepted state-of-the-art. These combined the benefits of shared central resources – typical of the older mainframe world – with the local computing resources of increasingly powerful desktop machines. Typical applications involved a 'fat client' with a server database. Most of the analytical heavy lifting took place on the client, while the server provided a shared database utility. The major drawback of this approach for enterprise applications was the need to undertake co-ordinated rollout of any updates to the fat application on multiple, often far-flung, client machines.

The rise of the internet in the late 1990s ushered in the 'thin' client paradigm we know today. Client machines did not require specialised applications. All that was needed was a generic application capable of parsing and interpreting self-describing files on the server, displaying their content appropriately and providing facilities for users to respond as desired to centrally generated messages and queries – what we all know today as a browser.

These two approaches share one common feature, however – namely, a broad reliance on relational databases for many types of structured data storage and retrieval. Within the framework of their table definitions, these databases enabled highly complex queries to access, filter, sort, group and otherwise manipulate data in a very flexible fashion. In this sense, relational databases and structured query language represented a huge advance. They were the culmination of years of trial-and-error development of database management systems.

Relational databases continue to be a powerful and effective tool in an environment where the structure of an underlying dataset is stable. Where they fall short is in dealing with a dynamic environment where the underlying data structure itself is unstable and subject to frequent revisions. In the context of trading activities, market data conventions tend to be comparatively stable. It is primarily trade terms and conditions that evolve in response to competitive forces. Accommodating such changes requires modifications to table structures that can have knock-on effects for existing applications that process pre-existing trades. For this reason, new trade types are often not immediately and smoothly accommodated in existing trading and risk systems. This creates side pockets of special processing that often fail to be included in broader enterprise risk systems.

David Rowe is senior strategist for risk and regulation at Misys in London.
Email: david.rowe@misys.com

"One part of the path forward involves abandoning the relational database paradigm for electronic storage of trade terms and conditions"

One part of the path forward involves abandoning the relational database paradigm for electronic storage of trade terms and conditions. The alternative is to represent trades via self-describing documents consisting of key-value pairs. This still requires a semantic structure in the spirit of FpML and similar XML-type representations. Using such a document store for recording and accessing trades introduces radical modularity into this dimension of the process. It is also a means of accommodating inconsistent relational database structures in multiple front- and middle-office trading systems. In effect, each trading system can continue to function on its own, providing it generates self-describing documents for each of the transactions it has booked. In this way, each trading system is responsible for maintaining its own trade document store, but the union of all these trade documents provides a modular representation of all trades that can be analysed at the enterprise level.

This type of system naturally lends itself to modular and massively parallelisable valuation routines that can be distributed to multiple nodes. There would have to be a central registry of documents similar to domain name servers for the internet. All this registry need contain, however, is a document ID and location. A separate metadata index records key/value pairs (or, in some cases, ranges of numerical values linked to certain keys such as 'TradeValue') and the document IDs in which they appear. This allows efficient identification of relevant documents, serving a function similar to, but on a much smaller scale than, Google's massive web index.

Of course, web search need only be suggestive, not definitive. An enterprise-wide risk management system would have to be much more complete and precise in its indexing. The index also has to evolve as new trade types and additional trade characteristics are developed. The point, however, is that none of this added indexing needs to affect pre-existing documents or create secondary impacts on existing analytic routines.

In effect, rather than trying to reform their universally fragmented data and analytical environments, financial institutions should look beyond the relational database mentality and embrace emerging technologies for search and massive parallelisation that lie behind our everyday experience with the internet. **R**

Risk 0214 Rowe.indd 50 29/01/2014 14:24