Using a data warehouse to solve risk, performance, reporting and compliance-related issues

Tom Stock

Received (in revised form): 7th January, 2011

GoldenSource Corporation, 22 Cortlandt Street, New York, NY 10007, USA;
Tel: +1 212 798 7100; E-mail: tstock@thegoldensource.com

Tom Stock *is SVP of Product Management for GoldenSource Incorporated which provides enterprise data management and data warehouse software to the financial services industry. Tom has developed enterprise systems for American Express and Bank of New York as well as GoldenSource and has over 20 years' experience in the software industry. Tom is a veteran manager at GoldenSource and holds a BA in finance and an MBA in accounting from the University of Wisconsin Madison.*

ABSTRACT

This paper discusses how a data warehouse fits into an overall enterprise data management (EDM) strategy. It discusses three common business problems in the financial services industry and how a data warehouse can be used to solve these typical industry issues. Real life case studies are used to illustrate data warehouse usage in solving performance and attribution, risk and customer reporting related business problems.

Keywords: data warehousing, enterprise data management (EDM), performance data capture and reporting, risk data warehousing and reporting, asset management, data warehouse

INTRODUCTION

There is no doubt that in today's world the need to fully manage your data environment is no longer a 'nice to have' but a necessity. Whether your company is a buy side, sell side or custody and security services organisation the overall benefits of implementing an enterprise data management (EDM) strategy are numerous and include:

- Improved customer service through a holistic view of your customer relationships.
- Greater customer satisfaction due to improved data access and reporting.
- Enhanced reliability and audit ability of risk, regulatory and compliance data.
- Lower costs through less fails and exceptions due to better data quality.

From the boardroom to the back office, EDM is now an accepted concept but is a concept that is not always well understood. This paper will discuss how an overall EDM strategy, including a data warehouse, can be used to solve real-life business issues. It will first describe how a data warehouse fits into an overall data management framework. It will then look at strategies for a successful data warehouse implementation. It will close by describing these strategies in action through a series of case studies where a data warehouse was used to solve business problems in real-world scenarios.

Journal of Securities Operations & Custody
Vol. 3 No. 4, pp. 305–315

1753–1802

THE DATA WAREHOUSE AND EDM

Master versus operational data

When we use the term 'EDM', enterprise data simply means all of the data a financial institution uses or generates as part of its daily activities. This data includes:

- Security master data such as security descriptive and indicative data, issuer, ratings and classification data, as well as information relating to corporate actions.
- Market data such as prices, interest rates and swap rates, as well as derived data such as zero curves and volatility surfaces.
- Counterparty data such as institutional hierarchies, ratings and classifications.
- Customer data such as name and address information, credit hierarchies, standing settlement instructions, as well as the relationships between customers and accounts.
- Transaction and position data including trades, cash receipts and disbursements, portfolio holdings and receivable and payable balances.

In examining the above data types it becomes clear there are really two distinct classes of data in this list, master data and operational data. Master data is data that can be brought into the organisation from outside sources such as data vendors or can be sourced inside the organisation from a single application for redistribution to the rest of the organisation. An example of outside sourced master data is security reference data coming from a financial data vendor like Bloomberg or Reuters. An example of internally sourced master data is a discount curve from a risk system that is distributed to the rest of the organisation as the gold copy.

The main characteristic of data in the operational data class is that it is generated by the operational systems of the organisation. Although the original source may be outside the organisation the one indisputable source of valid data of that type is an operational system within the organisation. An example is a securities transaction. Although the source of that data may be a trade completed outside the organisation the one true source of that data is the accounting system that processed that trade.

In summary, the main difference between master data and operational data is that the authoritative source for master data can be external data stored in a master data environment but the original source of operational data can never be the internal data repository; it must be an operational system. Referring back to the above list of data types, the first four could be considered master data, as the primary source for this data for use inside the organisation could be the master data repository. However, transaction and positional data is considered to be operational data as the original source of that data would be an operational system within the organisation.

Operational systems versus data warehouse

The reason for the above distinction is that there is a lot of confusion surrounding data warehouses versus security masters, versus customer masters, etc. The bottom line is that the systems that manage master data are different and distinct from data warehouses. Master data systems are meant to be the source of reference data for the entire organisation but are not the data warehouse for the organisation.

A data warehouse is a repository (collection of resources that can be accessed to retrieve information) of an organisation's electronically stored data, designed to facilitate reporting and analysis.[1]

According to Inmom a data warehouse is:

- subject-oriented — data in the data warehouse is organised so that all the data elements relating to the same object are linked together;
- time-variant — changes to the data in the data warehouse are tracked and recorded so that reports can be produced showing changes over time;
- non-volatile — data in the data warehouse is never over-written or deleted. Once committed, the data is static, read-only, and retained for future reporting;
- integrated — the data warehouse contains data from most or all of an organisation's operational systems and is made consistent.

This means that in a warehouse scenario, data is taken directly from an operational system, standardised and/or normalised to fit in with the rest of the organisation's data and then loaded into a persistent data store. Since in a master data application the data is potentially changing and being enriched after it has been loaded, they are in essence operational systems and therefore cannot also be a true data warehouse.

An integrated EDM strategy

Given the distinct nature of master data applications versus the data warehouse how do they work together? The answer is that without a clear strategy as to how to manage master data, implementing a data warehouse becomes much more difficult. If an organisation does not have consistent master data across all its applications it is attempting to consolidate, it will end up in a perpetual error correction cycle. During the warehouse load it will discover master data problems. It will go and fix these problems in the source systems. As the source systems are not connected for master data, new inconsistencies will be discovered in the warehouse load causing it to once again correct the data in the operational systems. Having an EDM strategy for master data will eliminate this vicious cycle and greatly improve the success of a warehouse project.

Implementing a successful data warehouse strategy

Getting started

Whether an organisation decides to buy or build a warehouse, any successful data warehouse implementation strategy starts out with knowing the output. Whether it is a feed to a risk system, customer reporting, or a source for performance and attribution input data, knowing what types of data and the frequency of use is the first step in the analysis phase. Of course, this does not mean that it should only map the information it needs for a specific output. Rather, once it has determined the data types it needs for its output it should map as much of that data type as it can in the time frames it has. Using this approach will help focus on the data types needed to solve a particular business problem and provide added value to future implementations. Once the output is known the organisation can then proceed to confirm the data structures and determine the operational sources of that data.

Now the difficult work can begin. Unless an organisation has a current warehouse it is replacing or has fairly new operational systems with good documentation, by far the largest effort on any warehouse project is analysing all the fields in the operational systems, determining their business meaning, and deciding how to normalise these fields to the warehouse. In the author's experience this task can easily take 30 per cent to 50 per cent of a project.

Phasing your implementation

Since the full implementation of a company-wide data warehouse can be a daunt-

ing task, most successful projects are broken up into phases that tackle a specific business problem. A successful phasing strategy is one that provides clear business value in a relatively short period of time, usually three to six months. The most successful strategies involve taking a subset of data and following the entire processing cycle from extraction to final output. The most common approaches that have proven successful are taking a limited set of data from multiple systems end-to-end or a more robust set of data from a limited number of systems end-to-end.

Under the first approach one defines a limited subset of data that can be extracted from multiple systems. For example, a large asset manager may ultimately be interested in a full warehouse for position, transaction, lot, and performance data. However, for phase one, they may decide that the first business problem they wish to solve is reporting assets under management across the firm. In order to implement this very targeted solution they would need to integrate the positional data from all their accounting systems into the warehouse and create a report to correctly calculate the firm-wide assets under management. Of course in order to support this requirement they would need some instrument information and portfolio (account) data as they would need to understand the firm's relationship to the portfolio (administration versus investment management). They would also need this reference information to avoid double counting of investments in funds that they also manage.

Under the approach of taking a robust set of data from a limited number of systems there are several ways to structure the phasing. For a large global firm they may want to roll out the implementation by region. For example, an institutional asset manager with operations in North America, EMEA and Asia Pacific may elect to tackle customer reporting for North American clients as the first business problem. To achieve this objective, they would need to integrate their North American investment accounting systems fully for position, transaction, and cost lot data first and then create their customer reports. Once this first implementation is complete they can then roll it out to the other regions.

However, if a firm manages multiple asset classes they may want to roll out their warehouse based on asset class. Consider an insurance company that manages real estate, commercial mortgages, syndicated loans, over-the-counter (OTC) derivatives, equities and fixed income securities. They may choose structured products underlying exposure reporting as the first business problem to address. In this case they would need to extract significant information from the derivatives management system, not just around the holdings of these derivatives but also around the structure of the instruments themselves. For example, a collateralised debt obligation (CDO) has multiple tranches backed by pools of debt that may be asset backed securities, which themselves can be backed by individual mortgages, credit card receivables, etc. Once all this information is extracted and consolidated in the warehouse then finally the underlying exposure reports can be written.

Unfortunately, the real word is not usually as simplistic as the above scenarios. Most large asset managers are located in multiple regions, invest in multiple asset types and service multiple lines of business. But the key point is still the same; pick a small initial project with true business value and your overall warehouse project will be off to a good start.

In the author's experience what does not generally work well is the '*Field of Dreams*' approach. Under this scenario a lot of initial effort is put into establishing a

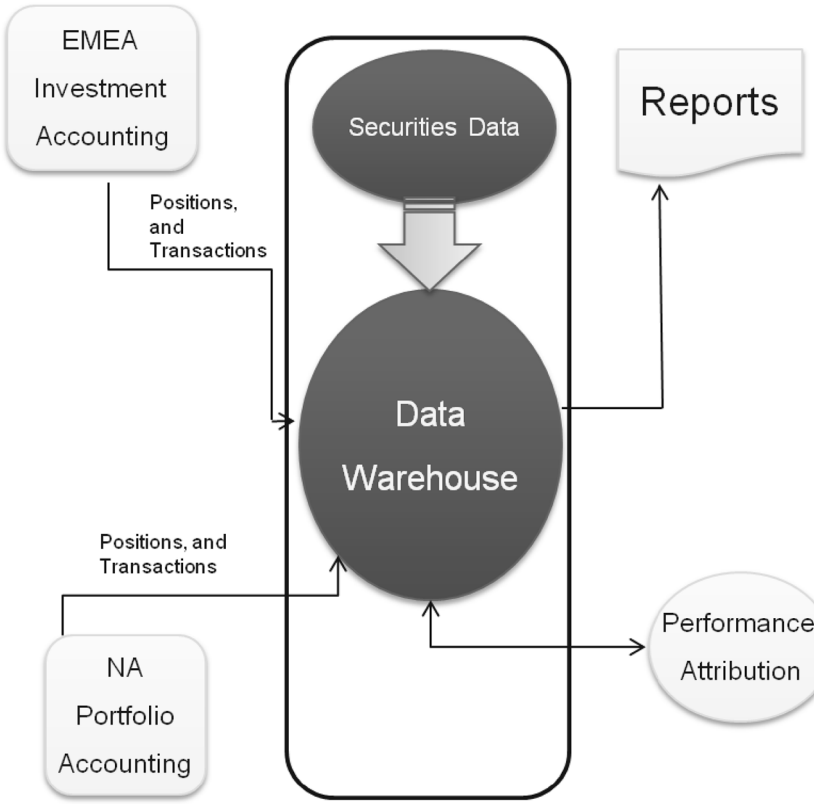

Figure 1 Data warehousing for performance and attribution data provisioning and reporting

robust complete data warehouse, however with no target business problem as a driver. The idea being: '*If we build it they will come*.' Typically this is an IT-driven approach with no identifiable business sponsor who can prove business value from the project. Therefore, funding is usually diverted to more ROI-driven projects before a complete solution can be achieved.

CASE STUDIES

Now that some of the terminology used in data management has been defined and strategies for successfully implementing a data warehouse have been discussed, the remainder of this paper will present several case studies of real business problems. The author will describe how the firms approached these business problems using an overall EDM strategy. Each case study will present the business problem, the EDM solution for that problem, the method of implementation the firm chose, and any challenges that arose during the implementation cycle and how they were overcome.

Consolidating data for performance and attribution

Business problem

A large Institutional asset manager had operations in several parts of the world including Europe and North America. Their primary issue was customer dissatisfaction related to performance reporting especially for those clients who had accounts in multiple regions. They wanted to centralise Performance and Attribution globally to ensure quality and consistency.

The challenge was that the firm had multiple investment accounting and performance and attribution systems with large differences in data quality and timeliness.

A second issue they wanted to address was the inflexibility of their current customer reporting system when it came to integrating performance data. To solve this problem they wanted to reuse the data warehouse they constructed for performance for customer reporting purposes as well.

Solution

The overall solution to this problem was to install a data warehouse and implement a new performance and attribution system. The operational position and transaction data needed for the performance calculations was to be fed from their multiple in-house accounting systems. For master-data management, the firm already had an in-house security master system up and running so the security master consolidation work had already been accomplished. Account master data was taken directly from the individual accounting systems as the total number of accounts was relatively small and there was relatively little overlap of account information on the different systems. For reporting they implemented a standard business intelligence tool and built custom reports. For the first two phases of the project, reporting was performed directly off the performance system. Phase three of the project involved loading performance and attribution data back into the warehouse from their newly implemented system and changing all reporting to be driven off the core data warehouse.

Implementation

The overall implementation approach chosen was to deploy the full warehouse extract, load and publish from the more robust EMEA investment account system first and then tackle the more limited North American investment account system. Thus, the firm followed the geographic phasing approach outlined above. This approach allowed the firm to get something up and running relatively quickly, which had real business value. It also solidified business sponsorship for the more difficult second phase of the project.

One major challenge for this implementation was that they needed to both compensate in the warehouse for missing data as well as enrich the positions provided by both accounting systems for performance-related values. Needing to generate missing data can be a fairly frequent occurrence when consolidating different systems with different capabilities. In this instance one of the systems adjusted the holdings for corporate actions but did not provide a transaction that represented that change. Another issue they had was that one system provided settlement date positions only so the trade date positions in the warehouse needed to be generated from the settlement date positions and the pending transactions. Enriching data is also a very frequent occurrence in a data consolidation project. In these circumstances for derivative transaction for both accounting systems, the team wanted to calculate the effective exposure before it was passed on to performance.

In this case both the creation of the missing transactions as well as the calculation of the trade date positions were accommodated by implementing a separate process that ran post daily load, but before the data was published to the performance and attribution system. This approach allowed for independent loading of data types to optimise performance, while at the same time provided timely data generation where necessary. The calculation of the effective exposure, however, was done as part of the load process itself as there was no dependency on other

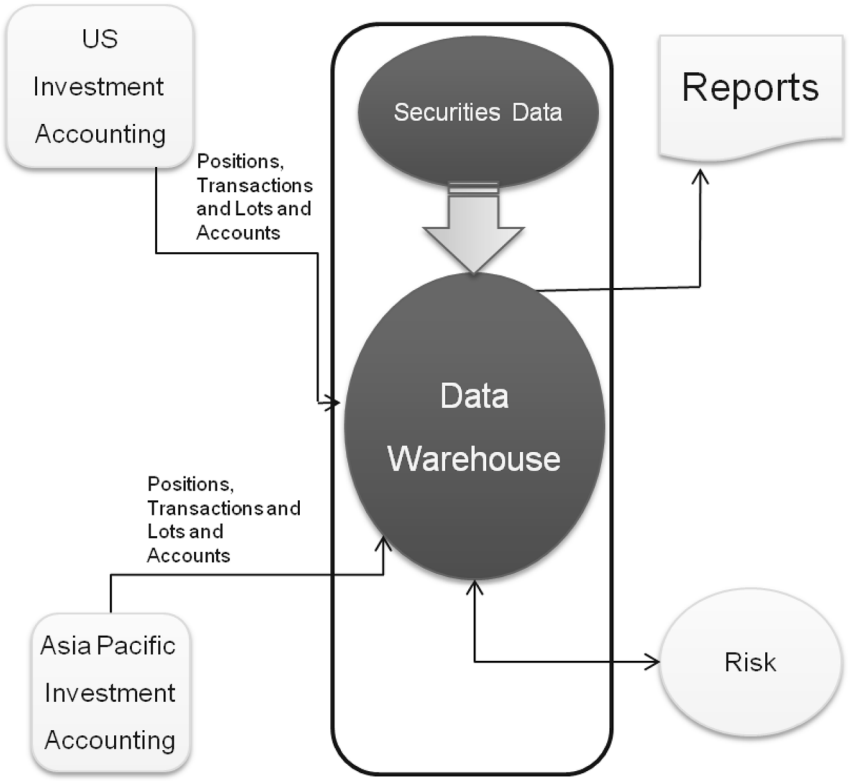

Figure 2 Data warehousing for risk data provisioning and reporting

non reference data. The point here is that the warehouse platform one implements not only needs to be able to provide for simple translation and validation but also needs to be flexible enough to accommodate sophisticated calculation routines as part of the load process as well as support separate data generation and aggregation that can occur after the data has been loaded.

Coping with a changing regulatory environment

Business problem

The firm undertaking this project was the asset management arm of a large insurance company. This company has locations in both North America and Asia Pacific, which use different accounting systems for their very complex derivatives deals. Their current process for calculating both credit and market risk had been a manually intense process, prone to error and did not provide a good audit trail. Due to increased regulatory oversight following the recent financial crisis, and in order to comply with the imminent requirements under Solvency II, a new approach to risk management was required. As stated by the Senior Supervisors Group that comprises senior financial supervisors from seven countries (USA, Canada, France, Germany, Japan, Switzerland, and UK:

> 'The implementation of highly developed risk data infrastructures requires more automation and fewer manual workarounds — two important conditions that can improve the accuracy and timeliness of risk data aggregation.'[2]

As in the previous case study, a secondary requirement was to leverage the work

that was done for feeding their risk system to provide a flexible platform for *ad hoc* and management reporting as well.

Solution

The firm decided to implement a data management layer including a data warehouse and a new risk management system. From the very beginning, this firm determined that in order to ensure the accuracy of the calculated risk metrics and to comply with the new regulations such as Solvency II, which mandated clean auditable input data as the basis for risk and reporting, they needed to install a data management layer as an integral part of the new risk system implementation. The operational data for positions, transactions and lots is being fed directly into the data warehouse from the two regional accounting systems. For master data administration this firm decided to implement a security master for reference data scrubbing. Although the source of the derivates terms and conditions was the accounting systems themselves, a security master was implemented as the source of data on the underlying securities of the derivatives, as well as direct holding of regular publically traded instruments. Direct vendor feeds for curve and surface-related information were also brought into the security master and scrubbed before they were loaded directly into the data warehouse for reporting purposes. In this instance, account data was not a big issue as all holdings were simply related to those areas that managed those asset types.

Implementation

In this instance, the approach that was undertaken was to integrate data from both accounting systems as part of the first phase of the project, but only data related to market risk. Credit risk would be accommodated in the second phase of the project. Overall reporting for portfolio management would be phase three. Thus, in this case the phasing strategy that was selected was to take a slice of data from all systems in order to solve one business problem, and then expand the data extracts in subsequent phases.

One of the main challenges of this implementation was the integration of positional data from multiple accounting systems. As the assets that were being consolidated were owned by the firm itself, the accounting for the assets followed the accounting principles and currencies of both the local and base entities. Thus, IFRS and US GAAP balances in both US$ and JP¥ had to be separately tracked and stored. Care had to be taken that the same cost or market values calculated on different accounting basses or in different currencies were not aggregated.

A second issue faced as part of this implementation is an issue faced in almost every data warehouse implementation. That is, the issue of ensuring data consistency and reconciliation. As the data is coming from multiple sources how do you ensure all of the data from all of the sources is loaded and is correct? In this instance, due to the criticality of the data being loaded, two separate methods of ensuring consistency were utilised.

The first method involved reconciling the business relationships between the operational data stored in the warehouse. This reconciliation involved ensuring the sum of the positions (for the respective accounting methods and base currencies) equalled the summary records loaded, ensuring that yesterday's position plus today's transactions equalled today's position and finally ensuring the sum of the individual cost lots equalled the position. This type of reconciliation ensures that all of the data actually loaded is internally consistent but not necessarily that all of the data was loaded.

In order to prove all of the data was

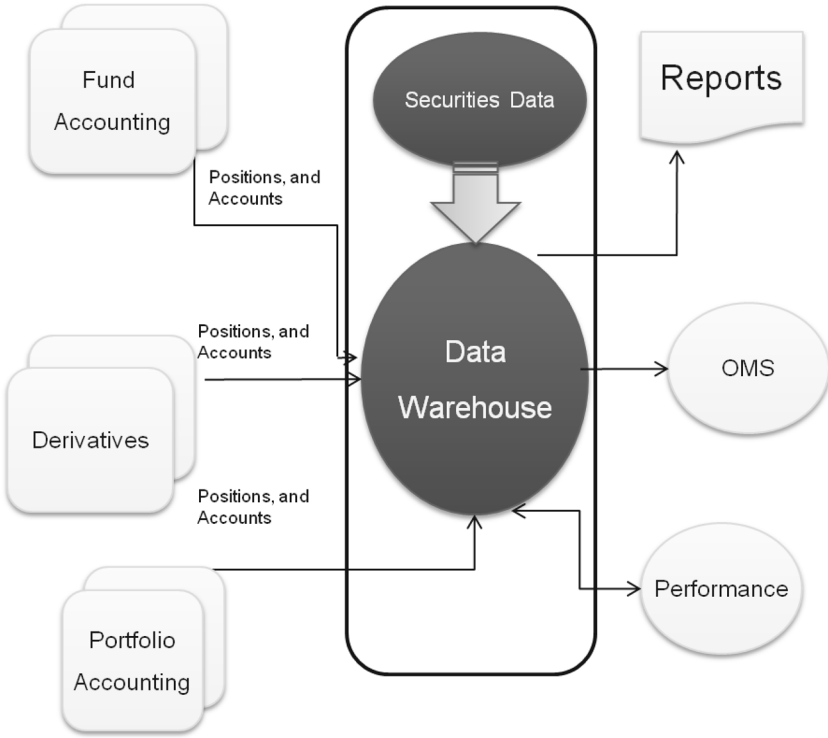

Figure 3 Data warehousing for customer reporting data provisioning and reporting

actually loaded, a second reconciliation of the individual positions to the source operational systems was also implemented. In order to implement this second reconciliation, a separate file of positional data was loaded into the warehouse from each accounting system. This data was then compared to the data independently loaded into the warehouse as part of the core processing to ensure they matched.

The final issue that needed to be addressed as part of the implementation was around data aggregation. In order to correctly assess and report risk, exposure and limits, individual position data needs to be aggregated along a number of different dimensions. These dimensions include issuer, credit rating, and country of risk, to name just a few. The decision that needs to be made is: should this aggregated data be persisted in the warehouse or can it just be calculated when needed? In order to properly answer this question, one needs to understand the volume of data being dealt with, the number and types of aggregations required and the response times needed for accessing or reporting on this data. Once all of this information is in hand, an educated architectural decision can be made. In this instance it was decided that the volume, latency and consideration did not justify persisting the data, so it was decided to do the aggregation as part of the reports.

Integrated customer reporting

Business problem

The last company case study examines a large asset manager that has an institutional business as well as a funds business. The company was undergoing a merger with

another asset manager and they needed to consolidate all their management reporting and end-of-day/beginning-of-day feeds to their risk and trading systems. Longer term they also wanted to consolidate redundant systems.

Solution

This company had all of the pieces of an EDM strategy in place before the merger. For reference data they already had a security master and account master in place. For operational position, transaction and lot data, they had a data warehouse that was used for reporting purposes. Thus, their overall solution to the issues presented by the merger was to use this opportunity to refresh their data warehouse technology and implement a new repository while at the same time ensuring their current security master could support all of the instruments from both the merging entities.

Implementation

The overall implementation approach this company chose is a data isolation strategy. Under this approach, a consolidated data warehouse is created as the first step of the integration. All reporting and analytics feeds are then driven off this new warehouse. Once this has been accomplished, it effectively isolates the reporting and analytics applications from the operational systems of the firm. This isolation allows the firm to combine and decommission operational systems without major changes to downstream systems.

The end state scope of this implementation was to store all standard data types from all operational systems to feed all reporting, trading and analytics systems. However, in order to break this up into more manageable pieces, this company utilised the approach of taking a limited subset of data from multiple systems. For phase one, positional data from all operational systems was loaded into the warehouse and a new customer reporting framework was utilised to produce customer reports. In subsequent phases, additional data types were added such as transactions, lots, benchmarks and performance data. Additional destinations were also added in these later phases, including performance, risk and trading systems.

For the security master data this company already had a security master in-house. They were currently undergoing a process to make sure that all data for both entities was accommodated. Thus, for the warehouse the only task that needed to be undertaken was to ensure that the data published to the warehouse had all the necessary data elements needed for reporting.

For account data there was already one account master repository in use at one of the merged companies; however, for the other company there was no single repository of account data. Thus, an account cross-reference needed to be built in the data warehouse so that the same account residing on multiple systems could be consolidated in the warehouse. For example, a single client could have traditional assets in an account that was processed on the standard investment management system while the same account could have derivative assets that were processed on the derivatives accounting system. The account numbers could be different on these two systems but with a proper cross-reference they could be combined together at the point the warehouse positional data was loaded.

One major issue encountered during this implementation was that the company discovered that they did not have all of the data in the operational systems or their current account master repository to fully complete the cross-reference. In order to solve this problem, it was decided to allow

limited update access to the warehouse so that the users could complete this data. Although this practice is against the general principle that a data warehouse should not be updated directly, this is an example where sometimes it is necessary to manually create reference data if it does not currently exist in the operational environment.

A final major challenge for this implementation was that the firm wanted to have the warehouse up and running in just three to six months. With six major operational systems to analyse and map as well as integration with their security master to complete, this was a tall order. Two approaches were considered to solve this problem. The first approach was to leverage the mapping work that had already been done for the existing warehouse to create mappings into the new data warehouse. The second approach was to build a mapping off the existing data warehouse into the new one. The final decision was to leverage the mapping already done to create new mappings as it greatly reduced the analysis time required while at the same time minimising the costs associated with running multiple data warehouses at the same time.

CONCLUSION

Embarking on an overall EDM or data warehouse programme can be a daunting task. However, if one fully understands the EDM landscape and chooses a winning implementation strategy the chances of success can be greatly improved. The following are some key points to remember when beginning a project:

- Understand the total data picture and make sure the system bought or built is suited to the data management problem being targeted.
- Begin the analysis on the output side of the picture for the data types needed but map as many of the attributes for those needed data types as possible in that phase. This process will target the work effort while still providing flexibility as data needs evolve.
- Do not underestimate the time and effort it takes to analyse operational systems. Cutting corners in this part of the project will come back to haunt you as data inconsistencies appear in the end results.
- Implement a system that has enough flexibility to handle complex calculations and data manipulations.
- Be cognisant of the best practices rules for data management but do not be afraid to tailor these to your specific circumstances.
- Always phase the overall programme and choose a first project that is short in duration (three to six months) and provides tangible business benefits to the business sponsor.

Keeping these key points in mind will greatly increase the chances of completing a project on time and within budget as well as ensuring a robust data warehouse is created that will soon become the single source of truth for the entire organisation.

REFERENCES

(1) Inmon, W. H. (1995) 'What is a Data Warehouse?' *Prism*, Vol. 1, No. 1.
(2) Senior Supervisors Group Observations on Developments in Risk Appetite Frameworks and IT Infrastructure 23rd December, 2010.