Breaking the Data Integration Bottleneck:

Accelerating Data Warehousing, Business Intelligence and Performance Management Applications

White Paper

by Richard Sherman
Athena IT Solutions

CoSORT
THE OPEN SYSTEMS STANDARD

Table of Contents

Overview

Businesses in all industries and sizes need to gather data from many sources to operate, improve performance, and better interact with customers, suppliers and partners. Although it's relatively easy to enable real-time data access, business people typically cannot use this "raw" data. They need it transformed to create consistent, comprehensive and current business information they can use for decision-making. For years, IT departments have been building systems to gather and transform data from disparate sources into business information. Data integration is the cornerstone of these efforts.

The data integration efforts enabling these systems are data warehousing (DW), Master Data Management (MDM) and Customer Data Integration (CDI). Business intelligence (BI) and corporate performance management (CPM) systems leverage this data.

In this white paper, we will discuss data integration approaches, identify performance bottlenecks, and recommend approaches to break those bottlenecks.

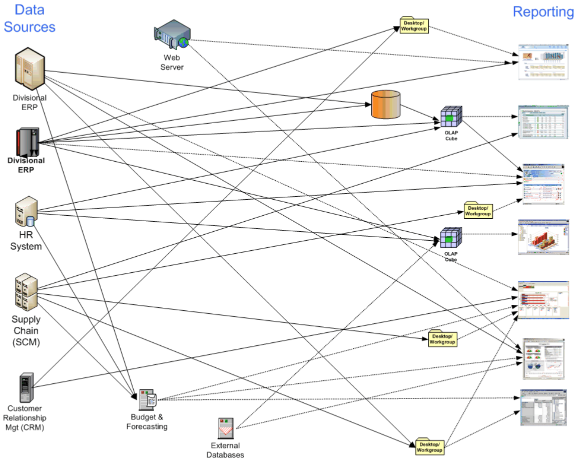
Approaches to Data Integration

Data and Process

Data is created both inside and outside an enterprise in diverse systems. It's also created by people such as consumers, partners, suppliers and employees using Microsoft Office and other applications. These diverse systems store data on many facets of the business, such as:

- Product, customer, supplier and account identifiers.
- Patient, physician, procedure and diagnostic identifiers from different insurers or healthcare providers.
- Customer addresses and contact information
- Product, marketing, sales and other organizational hierarchies

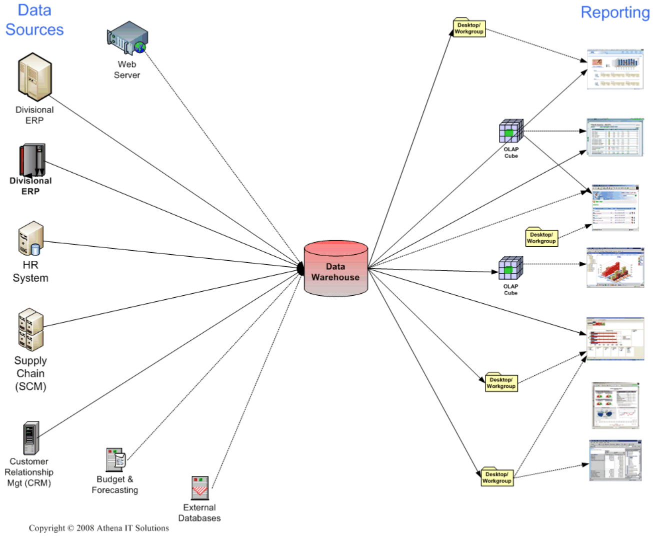
Because the data is stored on a variety of systems, however, problems can arise. When people access data directly from these source systems they will obtain different numbers based on which systems they select, how they process the data, and how it is aggregated. Business people then waste time disputing the resulting differences and trying to reconcile the numbers.



Copyright © 2008 Athena IT Solutions

Figure 1: Data Sources to Reporting Spider Web

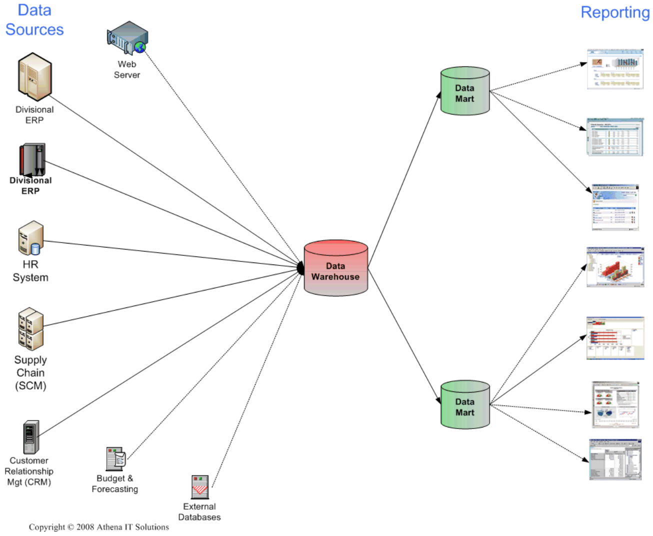
For consistency, it is a best practice to build a data warehouse (DW), enabling an enterprise to gather, cleanse, consolidate and transform the data once, and then share that consistent data with everyone who needs it. This is the most efficient and effective approach to sourcing data from diverse systems. It makes a lot more sense than performing these processes repetitively every time someone needs a report. With the huge volume of data in use today, it takes too long for a business person to wait while the data for their reports is gathered, cleansed, conformed and transformed in real-time.



Copyright © 2008 Athena IT Solutions

Figure 2: DW Replaces Spider Web

The DW was a great initial step that has evolved into "hub and spoke" architecture. This approach uses a DW (the hub) to anchor the data integration process and creates data marts, cubes. or flat-files (the spokes) from the DW to support reporting and analysis. The spokes enable business functions (e.g., finance, sales, marketing and HR), business processes (e.g., supply chain or customer services), or divisions and product groups to select, transform and aggregate data specific to what they do.



Copyright © 2008 Athena IT Solutions

Figure 3: DW Hub & Spoke Replace Spider Web

Technologies

Data Warehouse ETL (Extract, Transform and Load) tools have been around for over a decade. They have evolved from simple code generators to much more sophisticated data integration suites. Today, the data integration market is split into three approaches.

The first two approaches focus on ETL solutions from the best-of-breed market leaders and the rest of the market. The consensus, validated by numerous industry analysts, is that Informatica and IBM (Ascential) are the ETL market leaders. Having evolved into data integration suites, these comprise the first approach. The second approach represents lower cost and often adjunct products from the rest of the ETL pack, with strong offerings from SAP (via Business Objects), Oracle, Microsoft and SAS.

The third approach is based on a hand-coded, “build vs. buy” approach to ETL. Hand-coding is often selected because of the high initial investment (software costs, supporting infrastructure and resource skills) associated with packaged ETL. IT may also determine that hand-coded scripts are a better fit for their ETL tasks.

> *We are merely describing the data integration landscape rather than making value judgements on which tools are the top products, or if hand-coding is a better approach. We will leave those debates for another day.*

Data Integration Bottlenecks

Most enterprises, whether they are using ETL tools or developing complex hand-coded scripts, encounter data integration bottlenecks. Even if they are not experiencing these problems today, they are likely to face them in the near future as information demands continue to increase. Bottlenecks can cause major problems. For example, the value within the time constraints of an update cycle, or if data marts/cubes cannot be refreshed quickly enough for a business unit to react to rapid changes in the market.

The common reaction is to throw infrastructure (memory, disk space, additional servers, etc.) at the problem, or attempt to better tune the data integration code. These remedies are like the triage given at accident scenes. They stop the bleeding, but more medical attention is needed to save the patient. Likewise, investing in infrastructure or tuning may improve performance in the short-term, but the bottlenecks will most likely quickly reappear. You need to make a more fundamental change if you truly wish to break out of this bottleneck trap.

Data and Process Bottlenecks

There are two phases of enterprise data integration: data preparation and data franchising.

Data Preparation

The lion's share of a data integration effort is spent in data preparation ("Figure 4: Data Preparation Processes") processes between source systems and the DW. This includes:

- **Gathering data -** Enterprises of all sizes need to integrate many internal disparate source systems, plus those of outside suppliers, partners, etc.
- **Reformatting data** - Conversion and reformatting of the source system data into the DW format while conforming to current business and technical definitions.
- **Consolidating and validating data -** Multiple data sources consolidated to match formats and provide a single, consistent definition. The data is validated by querying dimensions or reference files to determine if it matches specific business rules.
- **Transforming data** - Business transformations, i.e. business rules and performance metrics to formulate the data into the context for business analytics.
- **Data cleansing** - Data quality checking includes reformatting, consolidation, validation and transformation, and involves multiple passes through the data. This typically either significantly slows ETL processing or is too complex to implement.
- **Storing data -** Data stored in DW, enabling further processing such as data franchising.

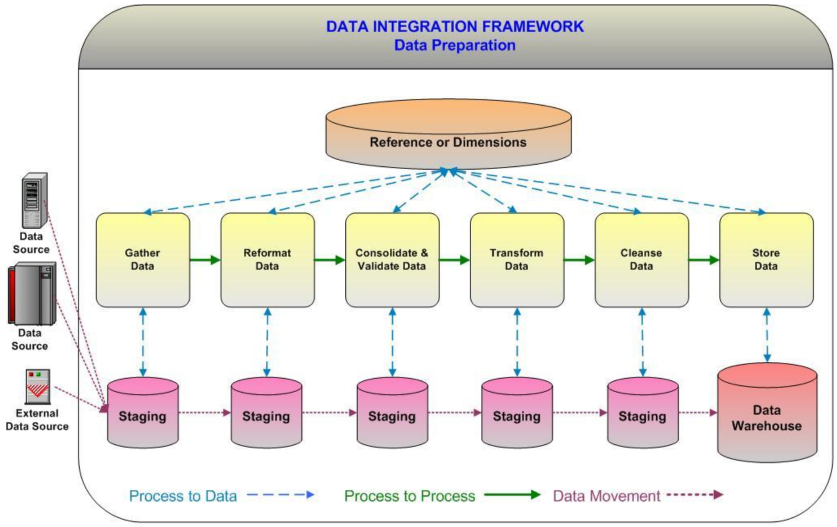
It is necessary to gather data from disparate source systems with minimal impact on the source systems and often within a specific time window. A common myth is that all you have to do is point an ETL tool at the source systems and pump data into the DW. If only that was true! Although ETL tools often access data directly from source systems, it is common for flat-file extracts to be used for both performance and political reasons.

Source systems are often mission-critical, transactional applications that run fully loaded during the day, with minimal nightly time windows for extractions. This means that all the data of the day has to be quickly pulled out in a short period of time. Since the source systems were built for update speed, and not for high-volume extracts, a common approach is to extract the data into flat files as quickly as possible. This limits the amount of time extraction processes are running on the source systems, but also offloads more data transformations to subsequent processing.

Corporate policies and politics are also factors. Application staff responsible for source systems do not want anything slowing performance in any way. So, the quicker data is pulled out of the source system, with the least amount of processing, the better.

Companies do not just extract flat-file extracts from internal sources systems. Many are also exchanging data with external partners, suppliers and information services, such as credit scoring agencies. For example, in the healthcare industry, payers or insurers exchange claims and payments with healthcare providers via flat files. Flat files are the most commonly exchanged data types in external transfers of data between companies.

Although using flat files may seem somewhat antiquated to some, the reality is that even when source system data can be pulled directly, it is still often more efficient to use flat files. With all these flat files being used, there is a significant opportunity to improve data integration performance by pre-processing or staging data using flat-file utilities.



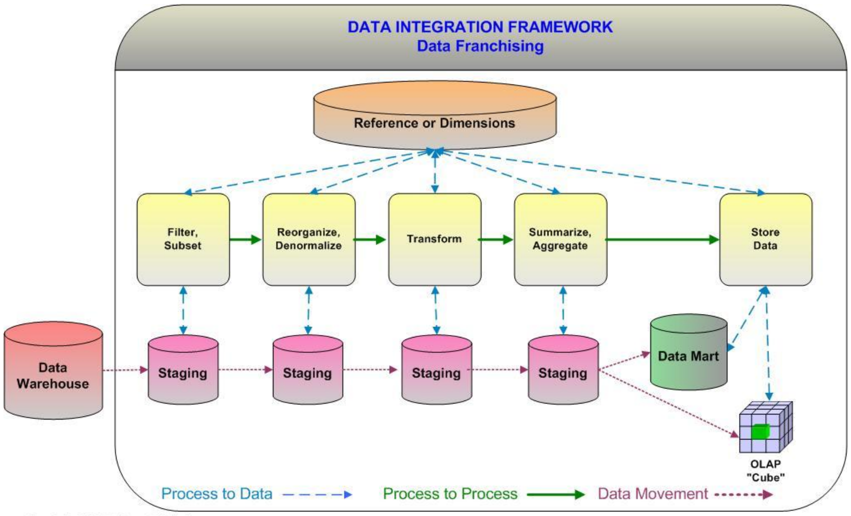
Copyright © 2007 Athena IT Solutions

Figure 4: Data Preparation Processes

Data Franchising

The purpose of **data franchising** ("Figure 5: Data Franchising Processing") is to "package" data so it is easier for business people to understand and use. This improves the usability and performance of BI tools and Microsoft Excel. Data may be franchised from a DW into data marts, cubes or data shadow systems. This involves:

- **Filtering or creating subset data**. Generally the DW has far too much data for business people to handle. Most people have responsibilities for specific product lines, geographies or business groups, causing them to filter or subset the data every time they perform an analysis. The most efficient way to handle this is to filter the data once for everyone who needs to examine it and load it in a data mart or cube.
- **Reorganizing or denormalizing the data** – Once filtered, data needs to be reformulated to match the requirements of the BI tool or spreadsheet being used for reporting. DWs are organized into hundreds or thousands of tables while the typical BI or reporting tools works best with flattened structures such as data marts or cubes.
- **Transforming data -** Business transformations, i.e. business rules and performance metrics, need to be processed to formulate data into the context for business analytics.
- **Aggregating or summarizing data** – Business people generally start their analyses using aggregated or summarized data to examine overall trends, and then drill into details to examine exceptions or results outliers. Many tools need to have the data pre-aggregated or summarized to meet the performance demands of business people.
- **Storing data** – Data is stored in the data mart, cubes or data shadow system for reporting and analysis by the business person.



Copyright © 2007 Athena IT Solutions

Figure 5: Data Franchising Processing

Data Shadow Systems or Spreadmarts

Although BI software has garnered major publicity for reporting and analytics for business users, the reality is that the most pervasive BI tool is the spreadsheet, Microsoft Excel. Business units gather data and create their own reports using spreadsheets because of the low cost, or simply because they cannot wait for IT-generated reports.

Data shadow systems ("Figure 6: Data Shadow Systems") typically start off with business "power" users gathering data for a report. These then evolve gradually into an application supporting more complex reporting. Data shadow systems are not planned, but evolve over time. The power user generally uses Microsoft Access and Excel because they are familiar with these readily available tools. The data shadow system is in reality a data franchising operation that is built by the power user rather than the IT department.

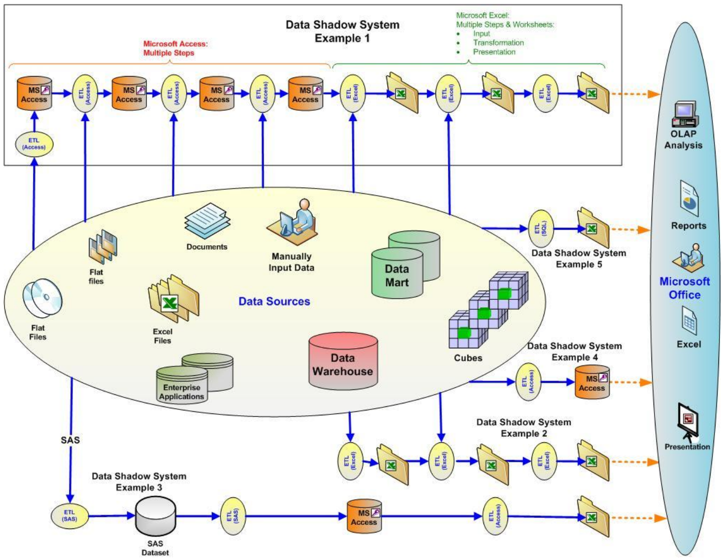


Figure 6: Data Shadow Systems

Technology and Tool Constraints

Enterprise Data Integration and ETL Tools

Regardless of what ETL tool is used, data preparation and franchising processes must still be performed. Often people lose sight of the details and just assume their ETL package can perform all these processes flawlessly. There are several common misconceptions encountered when using ETL tools that result in increased costs (development and maintenance), delayed projects, and performance traps:

- The "behind the curtain" myth
- The Open Database Connectivity (ODBC) & Java Database Connectivity (JDBC) trap
- A database loading dirty secret - bulk load/extract
- The failure to plan data staging

The first misconception is the "behind the curtain" myth. There is a common belief that ETL tools are like "magic boxes." That is, data goes in and information comes out, easily and efficiently. These expectations rise with the price of the ETL tool. People expect that these tools will always produce the most effective approach. Fantastic throughput numbers are published, however, people fail to notice the results "may vary based on your environment." Great performance is a result of setting up architecture and processes to leverage pragmatic approaches rather than just plugging in an ETL tool.

The second misconception is that you can implement convenience and portability in connectivity without a loss in performance and functionality. When ODBC came on the scene, there was a debate about its performance and functionality versus native database drivers. Still, OBDC was adopted for convenience and portability. Although ODBC and JDBC are great options for connectivity, many database operations either cannot be done, become too complex, or lag significantly in performance using these approaches.

One of the hidden "tricks" used by those who have been very successful in data integration and data migration projects is to leverage bulk loaders and extractors. The quickest way to get data in and out of a database is to shut off database operations such as referential integrity, logging, indexing and locking while using a bulk loader/extractor. Many ETL tools can use utilities behind the scenes that pre-process data prior to loading, further reduce "expensive" database operations and improve performance. Turning these database features off may seem counter-intuitive, but that is precisely how to increase the throughput of data integration processes, especially when coupled with pre-processing techniques.

Finally, a significant short-sightedness in many data integration efforts is failing to plan for and architect data staging. Data preparation and data franchising processes generally utilize data staging to accomplish their tasks. These staging areas may be in memory, in temporary tables or even in files outside a relational database. Too often, these data staging areas are thrown in the data integration work flow on a stopgap basis and are not leveraged. Data staging areas offer a significant opportunity to improve throughput and performance. For example, if you use bulk loaders and extractors to create flat files in data staging areas, you will further boost performance, and get access to data for other uses off-line (e.g. hoc transformation, replication, conversion, or reporting).

Hand-coded applications

Much of the data integration performed in an enterprise is accomplished by hand-coded applications. The approach taken is dependent on who codes the application: IT or business groups.

If IT creates the ETL processes, they will create SQL scripts that often become lengthy and complex. They will rely heavily on the database to process data and continually expand or patch SQL code as demand for data increases

If business groups develop the ETL processes, they will embed them in their data shadow systems. Typically, the business uses Microsoft Excel, Microsoft Access (shifting to Microsoft SQL Server 2005), and sometimes tools such as SAS. Since coding is not on most business people's resumes, these systems are a hodgepodge of processes cobbled together that are often cumbersome and require much manual intervention to run.

Regardless who creates the hand-coded applications; a major shortcoming is that each individual developer tends to rely heavily on tools with which they are already familiar. Consequently, hand-coded applications have a tendency to be slow, and to get slower as they are 'enhanced.' Still, hand-coding presents a perfect opportunity to leverage some best practices discussed above: bulk loaders/extractors, data staging and pre-processing to improve performance, quality and reliability.

Recommendations

Data and Process

Business Intelligence (BI), Performance Management (PM) and Master Data Management (MDM) applications leverage the data integration processes implemented through Data Warehousing (DW) initiatives.

It is always in fashion to describe a list of IT best practices, and there is also a tendency to emphasize the use of the 'latest and greatest' emerging technologies. However, this does not always reflect what works best for a given application. The data integration sessions most attended at conferences discusses 'tips & tricks' or pragmatic practices. The people in the trenches are always looking to improve data integration performance and throughput as their BI, PM, MDM and DW demands expand.

There are two fundamental tips to implementing the best performing data integration processes that will scale as data demands expand:

The first tip is to process data in sets rather than the traditional row-by-row. The processes vary between data preparation and franchising, but the basic unit of work still remains data sets rather than the sequential row-by-row method implemented in most ETL processes. This sequential approach constrains the update cycle, often bogging down processing, and almost always limiting the ability of IT to proactively manage data quality. Unfortunately, the row-by-row approach is how most ETL jobs are designed.

The second tip is that relational databases and SQL are terrific for many things, but certainly not for everything. Critical bottlenecks are often directly related to the overuse of relational technology as a "one size fits almost all" tool. Just as a spreadsheet should not be used in lieu of a database, a database should not be used in lieu of tools that are better suited to high-volume processing. When designing and implementing data integration applications, determine what you need to do, how you are going to do it, and then select the tool that best matches the what and how.

CoSort Features and Benefits

Accelerating Data Integration

Enterprises need to gather data from many sources, conform and cleanse the data, and then store it in a DW. These are only the first steps in data's journey to MDM, BI and PM applications. After the DW, it has to be filtered, transformed, aggregated and then stored yet again (in data marts, cubes or files) for reporting and analysis. These processes typically require large sets of data. Data integration experts continue to leverage a key approach in their toolkit: using flat-file utilities to handle large sets of data and then loading the results of those operations into a relational database for DW and BI.

The IRI / CoSort utilities that accelerate these processes are:

- *FACT for Oracle*: high-speed bulk Oracle extracts to flat files
- *CoSort*: fast integration, transformation, conversion, security and reporting for large flat files
- *RowGen*: referentially correct, bulk test data generation for DW volume and performance testing

Speeding DW Unloads and Loads

Reducing database load/unload processing requires that data be extracted to, transformed, and then loaded from, flat files. These in-and-out processes are very fast database tasks that support large sets of data. In addition, these bulk operations scale very efficiently so enterprises do not encounter bottlenecks as their data needs inevitably expand. *CoSort* utilities support parallel processing, greatly increasing throughput and reducing expensive and time consuming read/write operations. For example, pre-sorting flat files can save database resources and dramatically improve the performance of bulk load utilities.

FACT for Oracle rapidly unloads large Oracle tables to a wide variety of flat files, using all supported combinations of native SQL SELECT features. The data is then available for DW pre-load sorting, transforms, change and summary reporting, or replication. Data can either be written to files, or piped directly into *CoSort* (for transformation, migration, reporting, etc.) or other products and applications. *FACT for Oracle* also creates the metadata for simultaneous transformations via CoSort and loading via SQL*Loader.

CoSort does the heavy lifting, sorting and transforming of data into pre-processed datasets optimized for loading and querying within the target DW, data mart, or cube. *CoSort* offers a wide spectrum of data operations via flat-file processing, enabling enterprises to manipulate and process large data sets throughout their data preparation and franchising processes.

Specific CoSort data integration functions include:

- Select and filter
- Sort and merge
- Join and lookup
- Cross-calcs and PCREs
- Aggregations
- Field remapping
- Field masking and encryption
- Data and file type conversion
- Sequence
- Complex (user) transforms

Change Data Capture

Change Data Capture (CDC) is a common technique to improve performance when gathering data. The key principle is to only process data that has been added, changed or deleted since the last time the data was gathered. This is conceptually straightforward and definitely improves operations IF you can do it.

CDC processing requires comparison of large sets of data to determine what has changed and what has not. Unfortunately, many enterprises find that CDC is difficult to accomplish with their existing relational databases or ETL tools. *CoSort* delivers efficient CDC by enabling large sets of data to be compared and tagged for inclusion in the CDC data set. This offloads the CDC bottleneck from the typical enterprise toolkit and onto a file-base tool designed for integrated, high-volume file compare and reporting jobs.

Data Cleansing

Data cleansing is another area where data needs to be processed in sets, and in-line with other functions. Too often this area is neglected in BI and DW processing because of the bottlenecks encountered. *CoSort* performs multiple scrubbing processes and compares data sets in a single pass, enabling a very effective data cleansing approach. When specialized cleansing logic is required, *CoSort* users can link to a custom library routine that transforms the data at the field level in the same pass as its other operations.

Scalable Data Preparation for BI, PM, MDM, etc.

Finally, data franchising processes convert the data locked in data warehouses into information usable by BI, PM and MDM applications. These processes are often hand-coded because of their deceptive simplicity, but many enterprises run into performance bottlenecks during this processing.

Data franchising involves filtering, transforming and aggregating data to create business information. It is easy to hand-code these processes if the amount of data is small, not updated frequently, and does not require many transformations. However, enterprises quickly encounter limits in the volume, frequency and extent of the processing they can perform through hand-coded applications or even ETL tools.

If you instead off load the data from relational databases and choose to operate on the data in large sets, *CoSort* will perform these franchsing operations very efficiently and quickly. *CoSort* can also produce segemented hand-off files in CSV and XML format (for BI tool imports), along with detail and summary reports for billing applications, web posting, and similar-use targets. *CoSort*'s combination of scalable performance and flexible output formatting allows enterprises to: 1) handle their inevitable increases in data volumes; 2) remove bulk transformation overhead from the advanced-BI-tool layer; and, 3) more immediately derive business value from their transactional data.

Integration with Enterprise Data Integration and ETL Tools

Data integration processes occur whether you are hand-coding applications or using ETL tools. A terrific endorsement of *CoSort* utilities and flat files in data integration is that many ETL tools can leverage these utilities via plug-ins, API connections or utility calls. The vendor roster includes the market-leading data integration suites from Informatica and IBM, along with related offerings from SAS, ETI, DataStreams et al. This presents an opportunity for enterprises that have made investments in these ETL tools to match the best technologies for the job, i.e. file utilities for pre-processing large data sets and then handing off the results to their selected ETL tool for operations on the subsets.

The mixing of ETL tools with *CoSort* enables enterprises to manage and develop their data integration processes from one platform, i.e. their ETL tool, while taking advantage of *CoSort*'s performance, scalability and ability to change, secure, and re-represent data.

Metadata Management

When developing software it is a best practice to use source code management to manage and control the process. Similarly, data integration is best managed when developers use metadata management in their development processes. Metadata is the "source code" of data; it defines what data is, how it is transformed, and where it is used.

CoSort offers indigenous metadata management activities that can be used in the development and maintenance of data integration processes. These include:

- Data and job lineage through explicit, easy-to-use data definition files
- Version control through job scripts and an .xml audit trail
- Data and application separation via metadata centralization and re-usability
- Compatibility with the metadata used in other tools (like FACT and RowGen)
- Conversion of existing data definitions in COBOL copybooks, CSV and ELF web log file headers and SQL*Loader control files

The *CoSort* data definition file (.ddf) format is also supported by the Meta Integration Model Bridge (MIMB) application from Meta Integration Technology, Inc. (MITI). MIMB allows ETL, BI, ERP and other software tool users with existing flat-file definitions to automatically convert (and thus leverage) their metadata into *CoSort* .ddf format. This facilitates the use of *CoSort* with many other tools because the conversion saves time over manual metadata re-definitions, and eliminates potential errors

Privacy and Security

Accompanying pervasive data accessibility is a great responsibility, and one that is often regulated to ensure data security and privacy. Well publicized breaches have occurred because IT did not adequately secure data being transferred or stored. *CoSort* has built-in, role-based access control (RBAC) functions to secure data in files down to the field level, if necessary. Confidential data at rest can be masked, de-identified, filtered, and/or encrypted while also being transformed and processed in large volumes. These capabilities help address GLBA, HIPAA, PCI, EU Data Protection Directive, SOX, FERC/NERC, and other privacy regulations. As this is another topic, there is another *CoSort* white paper and web site solution area dedicated to it.

Conclusions

The fundamental processes in data integration continue to present challenges to enterprises. This is not because the fundamentals have changed, but rather that expectations for data volumes to support reporting and analysis continue to increase. Although many look for the next emerging technology to solve data integration bottlenecks, it is clear to those that have been successful in handling their enterprise's data volume needs that techniques and tools already exist, are affordable, and can be quickly implemented.

The best approach to successful, large-scale data integration is to leverage file utilities for those processes that require processing of large, structured data sets. This approach may not get the attention that emerging technologies may, but look under the covers of many successful data integration applications and you will find that this approach is part of the savvy data integration practitioner's 'tips & tricks' arsenal.

About the Author

RICHARD P. SHERMAN is the founder of Athena IT Solutions, a Boston-area firm offering data warehousing and business intelligence consulting and training. Sherman has over 20 years of experience in data warehousing and business intelligence systems, having worked on more than 50 implementations as a director/practice leader at PricewaterhouseCoopers and while managing his own firm. He is an expert instructor and speaker at industry conferences and seminars and teaches at Northeastern University's Graduate School of Engineering. He has a monthly column "The Data Integration Advisor" in DM Review, writes articles for searchDatamanagement.com and has been quoted in financial and industry publications. Sherman writes for two blogs: The Data Doghouse and Informatica's Enterprise Data Management blog. He can be reached at rsherman@athena-solutions.com.

About IRI, The CoSort Company

Founded in 1978, IRI is headquartered in Melbourne, Florida, and represented in over 30 international locations. IRI's suite of data manipulation products are based on best-of-breed, parallel processing technologies and proprietary algorithms that deliver optimal performance for bulk data integration, migration, protection and reporting. Visit www.iri.com, or call 1-800-333-SORT for information.

Product information links:

- CoSort 9
- FACT for Oracle
- RowGen (test data)