Information integration solutions
February 2006

WebSphere® software

A roadmap to enterprise data integration.

Colin White
BI Research

Contents

Data integration in the enterprise

Companies are generating an ever increasing amount of data, and studies show that uncontrolled data growth and data integration problems are slowing down the deployment of new applications. In a recent study by The Data Warehousing Institute (TDWI),[1] for example, 69 percent of the organizations surveyed said that data integration issues are a barrier to implementing new applications. The three main data integration concerns of survey respondents were data quality and security, lack of a business case and inadequate funding, and a poor data integration infrastructure.

To help solve data integration problems companies are increasing their funding of data integration projects.

In the Worldwide Data Integration Spending 2004-2008 Forecast report, IDC estimates data integration spending worldwide will increase from $9.3 billion in 2003 to $13.6 billion in 2008. For data integration to be successful, however, this funding must be used to solve data quality problems and to build an enterprise-wide data integration infrastructure.

The objective of this report is to take a detailed look at how organizations can plan and build an enterprise infrastructure for supporting data integration applications. It will review current data integration techniques and technologies, and offer suggestions as to which of these could be used for any data integration application. It will also demonstrate how IBM's data integration solution can be used to support an enterprise-wide data integration environment.

Characteristics of data integration

Data integration involves a framework (see Figure 1) of applications, tools, techniques, technologies and management services for providing a unified and consistent view of enterprise business data to business processes and business users.

- **Applications** *are custom-built or vendor-developed solutions that utilize one or more data integration tools.*

- **Tools** *are off-the-shelf commercial products that support one or more data integration technologies. These tools are used to design and build data integration applications.*

- **Technologies** *implement one or more data integration techniques.*

- **Techniques** *are technology-independent approaches for doing data integration.*

- **Management services** *support the management of data quality, metadata, and data integration system operations.*

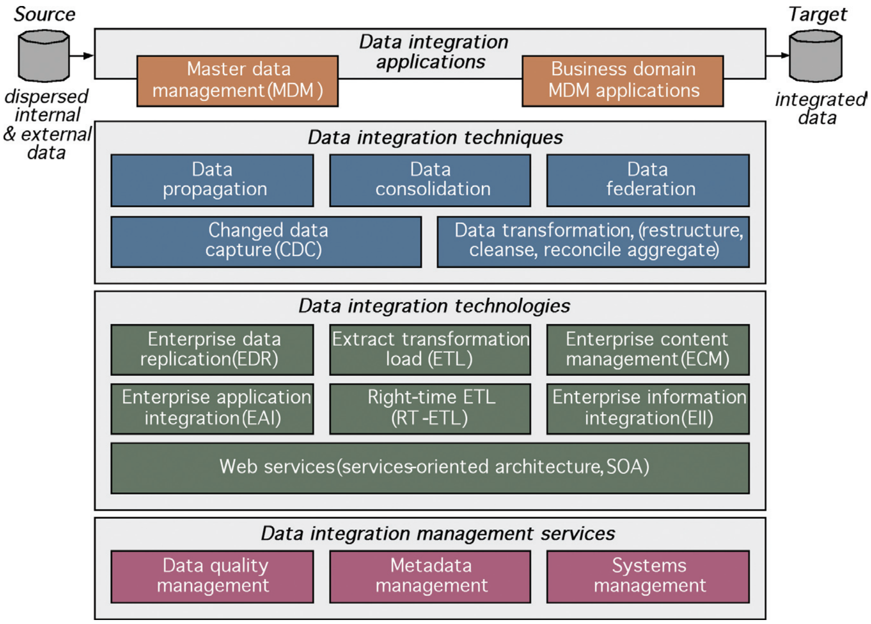
Source
dispersed internal & external data
Data integration applications
Master data management (MDM)
Business domain MDM applications
Target
integrated data
Data integration techniques
Data propagation
Data consolidation
Data federation
Changed data capture (CDC)
Data transformation, (restructure, cleanse, reconcile aggregate)
Data integration technologies
Enterprise data replication (EDR)
Extract transformation load (ETL)
Enterprise content management (ECM)
Enterprise application integration (EAI)
Right-time ETL (RT-ETL)
Enterprise information integration (EII)
Web services (services-oriented architecture, SOA)
Data integration management services
Data quality management
Metadata management
Systems management

Figure 1. Components of a data integration infrastructure.

In this report, we first review the techniques, technologies and management services used in data integration projects, and then look at how data integration applications and tools are designed, built and deployed using these capabilities.

Data integration techniques

The three main techniques used for integrating data are data consolidation, data federation, and data propagation (see Figure 2). These three techniques may in turn use changed data capture and data transformation techniques during data integration processing.

Data consolidation

Data Consolidation captures data from multiple source systems and integrates it into a single persistent data store. This data store could be, for example, a data warehouse that is used for business intelligence application reporting and analysis, or a content repository containing unstructured information such as documents, images, and web pages.

With data consolidation, there is usually a delay, or *latency*, between the time updates occur in source systems and the time those updates appear in the target store. Depending on business needs, this latency may be a few seconds, several hours, or many days. The term *near-real-time* is often used to describe target data that has a low latency of a few minutes, or maybe a few hours. Data with zero latency is known as *real-time* data, but this is difficult to achieve using data consolidation.

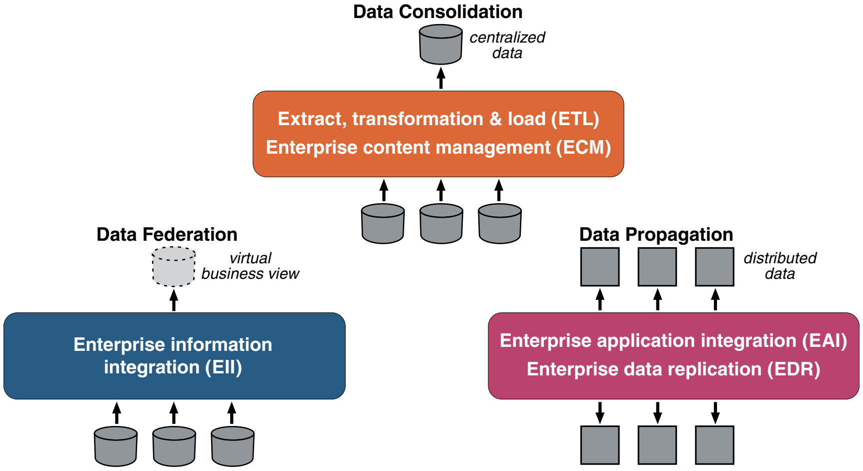
Data Consolidation
centralized data
Extract, transformation & load (ETL)
Enterprise content management (ECM)
Data Federation
virtual business view
Enterprise information integration (EII)
Data Propagation
distributed data
Enterprise application integration (EAI)
Enterprise data replication (EDR)

Figure 2. Data integration techniques: consolidation, federation and propagation .

Target data stores that contain high-latency data (more than one day, for example) are built using *batch* data integration applications that *pull* data from the source systems at scheduled intervals. Low-latency target data stores, on the other hand, are updated by *online* data integration applications that continuously capture and *push* data changes to the target store from source systems. This push approach requires the data consolidation application to identify the data changes to be captured for data consolidation. Some form of changed data capture (CDC) technique is usually used to do this.

Pull and push consolidation modes can be used together – an on-line push application can, for example, accumulate data changes in a staging area, which is queried at scheduled intervals by a batch pull application. It is important to realize that push mode is *event*-driven and pull mode is *on-demand* driven.

The advantage of data consolidation is that it allows large volumes of data to be transformed (restructured, reconciled, cleansed and/or aggregated) as it flows from source systems to the target data store. The disadvantages are the computing resources required to support the data consolidation process and the amount of disk space required to support the target data store.

Data consolidation is the main approach used by data warehousing applications to build and maintain an operational data store or an enterprise data warehouse. Data consolidation can also be used to build a dependent data mart, but in this case the consolidation process uses a single data source (i.e., an enterprise data warehouse). In a data warehousing environment, ETL (extract, transform, and load) technology is one of the more common technologies used to support data consolidation. Another data consolidation technology is ECM (enterprise content management). Most ECM solutions focus on consolidating and managing unstructured data such as documents, reports, and Web pages.

Data federation

Data Federation provides a single *virtual* view of one or more source data files. When a business application issues a query against this virtual view, a data federation engine retrieves data from the appropriate source data stores, integrates it to match the virtual view and query definition, and sends the results to the requesting business application. By definition, data federation always *pulls* data from source systems on an on-demand basis. Any required data transformation is done as the data is retrieved from the source data files. Enterprise information integration (EII) is an example of a technology that supports a federated approach to data integration.

One of the key elements of a federated system is the metadata used by the data federation engine to access the source data. In some cases, this metadata may consist solely of a virtual view definition that is mapped to the source files. In more advanced solutions, the metadata may also contain detailed information about the amount of data that exists in the source systems and what access paths can be used to access it. This more extensive information can help the federated solution optimize access to the source systems.

The main advantages of a federated approach are that it provides access to current data and removes the need to consolidate source data into another data store. Data federation, however, is not well suited for retrieving and reconciling large amounts of data or for applications where there are significant data quality problems in the source data. Another consideration is the potential performance impact and overhead of accessing multiple data sources at run time.

Data federation may be used when the cost of data consolidation outweighs the business benefits it provides. Operational query and reporting is an example where this may be the case. Data federation can be of benefit when data security policies and license restrictions prevent source data being copied. Syndicated data usually falls into this latter category. It can also be used for as a short-term data integration solution following a company merger or acquisition.

The source data investigation and profiling required for data federation is similar to that needed with data consolidation. Organizations should therefore use data integration products that support both data consolidation and federation, or at least products that can share the metadata used for consolidation and federation.

Data propagation

Data Propagation applications copy data from one location to another. These applications usually operate online and *push* data to the target location; i.e., they are event-driven. Updates to a source system may be propagated asynchronously or synchronously to the target system. Synchronous propagation requires that updates to both source and target systems occur in the same physical transaction. Regardless of the type of synchronization used, propagation guarantees the delivery of the data to the target. This guarantee is a key distinguishing feature of data propagation. Most synchronous data propagation technologies support a two-way exchange of data between a data source and a data target. Enterprise application integration (EAI) and enterprise data replication (EDR) are examples of technologies that support data propagation.

The big advantage of data propagation is that it can be used for the real-time or near-real-time movement of data. Other benefits include guaranteed data delivery and two-way data propagation. The availability of many of these facilities will vary by product. Data propagation can also be used for workload balancing, backup and recovery, and disaster recovery.

Data propagation implementations vary considerably in both performance and data restructuring and cleansing capabilities. Some enterprise data replication products can support high volume data movement and restructuring, whereas EAI products are often limited in their bulk data movement and data restructuring capabilities. Part of the reason for these differences is that enterprise data replication has a data-centric architecture, whereas EAI is message- or transaction-centric.

A hybrid approach

The techniques used by data integration applications will depend on both business and technology requirements. It is quite common for a data integration application to use a *hybrid* approach that involves several data integration techniques. A good example here is a customer master data management (CDM) application where the objective is to provide a harmonized view of customer information.

A simple approach to CDM is to build a consolidated customer data store that contains customer data captured from source systems. The latency of the information in the consolidated store will depend on whether data is consolidated online or in batch, and how often updates are applied to the store.

Another approach to CDM is data federation where virtual business views of the customer data in source systems are defined. These views are used by business applications to access current customer information in the source systems. The federated approach may also employ a metadata reference file to connect related customer information based on a common key.

A hybrid data consolidation and data federation approach may also be appropriate. Common customer data (name, address, etc.) could be consolidated in a single store, but customer data that is unique to a specific source application (customer orders, for example) could be federated. This hybrid approach can be extended further using data propagation. If a customer updates his or her name and address during a Web store transaction, this change could be sent to the consolidated data store and then propagated to other source systems such as a retail store customer database.

Data integration technologies

A wide range of technologies is available for implementing the data integration techniques outlined above. This section reviews three of the main ones: extract, transform, and load (ETL); enterprise information integration (EII); and enterprise application integration (EAI). It also briefly reviews enterprise data replication (EDR) and enterprise content management (ECM).

Extract, transform, and load

As the name implies, ETL technology extracts data from source systems, transforms it to satisfy business requirements, and loads the results into a target destination. Sources and targets are usually databases and files, but they can also be other types of data stores such as a message queue.

Data can be extracted in schedule-driven pull mode or event-driven push mode. Both modes can take advantage of changed data capture. Pull mode operation supports data consolidation and is typically done in batch. Push mode operation is done online by propagating data changes to the target data store.

Data transformation may involve data record restructuring and reconciliation, data content cleansing and/or data content aggregation. Data loading may cause a complete refresh of a target data store or may be done by updating the target destination. Interfaces used here include *de facto* standards like ODBC, JBDC, JMS, for example, or native database and application interfaces.

Early ETL solutions involved running batch jobs at scheduled intervals to capture data from flat files and relational databases and consolidate it into a data warehouse database managed by a relational DBMS. Over recent years, commercial ETL vendors have made a wide range of improvements and extensions to their products. Examples here include:

- *Additional sources–legacy data, application packages, XML files, Web logs, EAI sources, Web services, unstructured data*
- *Additional targets–EAI targets, Web services*
- *Improved data transformation–user defined exits, data profiling and data quality management, support for standard programming languages, DBMS engine exploitation, Web services*
- *Better administration–job scheduling and tracking, metadata management, error recovery*
- *Better performance–parallel processing, load balancing, caching, support for native DBMS application and data load interfaces*
- *Improved usability–better visual development interfaces*
- *Enhanced security–support for external security packages and extranets*

These enhancements extend the use of ETL products beyond consolidating data for data warehousing to include a wide range of other enterprise data integration projects.

Enterprise information integration
EII provides a virtual business view of dispersed data. This view can be used for demand-driven query access to operational business transaction data, a data warehouse, and/or unstructured information. EII supports a data federation approach to data integration.

The objective of EII is to enable applications to see dispersed data as though it resided in a single database. EII shields applications from the complexities of retrieving data from multiple locations, where the data may differ in semantics and formats, and may employ different data interfaces.

In its basic form, EII access to dispersed data involves breaking down a query issued against a virtual view into subcomponents, and sending each subcomponent for processing to the location where the required data resides. The EII product then combines the retrieved data and sends the final result to the application that issued the query. More advanced EII solutions contain sophisticated performance facilities that tune this process for optimal performance.

EII products have evolved from two different technology backgrounds – relational DBMS and XML. The trend of the industry, however, is toward products supporting both SQL (ODBC and JDBC) and XML (XQuery and XPath) data interfaces. Almost all EII products are based on Java.

Products vary considerably in their features. Query optimization and performance are key areas where products differ. EII products that originate from a DBMS background often provide better performance because they take advantage of the research done in developing distributed database management systems (DDBMS).

Most EII products provide *read-only* access to heterogeneous data. Some products provide limited update capabilities, however. Another important performance option is the ability of the EII product to cache results and allow administrators to define rules that determine when the data in the cache is valid or needs to be refreshed.

Distinguishing features to look out for when evaluating EII products include the data sources and targets supported (including Web services and unstructured data), transformation capabilities, metadata management, source data update capabilities, authentication and security options, performance, and caching.

EII versus ETL
It is important to emphasize that EII data federation cannot replace the traditional ETL data consolidation approach used for data warehousing. A fully federated data warehouse is not recommended because of performance and data consistency issues. EII should be used instead to extend and enhance a data warehousing environment to address specific business needs.

EII is a powerful technology for solving certain types of data access problems, but it is essential to understand the trade-off of using federated data. One issue is that federated queries may need access to an operational business transaction system. Complex EII query processing against such a system can affect the performance of the operational applications running on that system. An EII approach can reduce this impact by sending less complex and more specific queries to the operational system.

Another potential problem with EII is how to transform data coming from multiple source systems. This is a similar problem that must be addressed when designing the ETL processes for building a data warehouse. The same detailed profiling and analysis of the data sources and their relationships to the targets is required. Sometimes, it will become clear that a data relationship is too complex, or the source data quality too poor, to allow federated access. EII does not in any way reduce the need for detailed modeling and analysis. It may in fact require more rigor in the design process, because of the real-time nature of data transformation in an EII environment.

Both ETL and EII have a role to play in data warehousing and data integration, and organizations will need to implement both of these technologies. Rather than buying two separate products for ETL and EII, companies should look for vendors that support both technologies in a single integrated product set with shared metadata.

ETL vendors are beginning to offer an EII capability, which may be provided by the ETL product itself, or by using the services of a third-party product. Some ETL products use EII services behind the scenes to access heterogeneous data.

There are circumstances when the EII component within the offering must be deployed by itself on a separate system. An enterprise portal or dashboard application that employs EII to access a variety of data stores is an example of such a situation. In this case, the deployment of a complete data integration product set on the portal platform is not required and may be cost prohibitive.

Enterprise application integration

EAI integrates application systems by allowing them to communicate and exchange business transactions, messages, and data with each other using standard interfaces. It enables applications to access data transparently without knowing its location or format. EAI is usually employed for real-time operational business transaction processing. It supports a data propagation approach to data integration.

The direction of the EAI industry is toward the use of an enterprise service bus (ESB) that supports the interconnection of legacy and packaged applications, and also Web services that form part of a service oriented architecture (SOA).

From a data integration perspective EAI can be used to transport data between applications and to route real-time event data to other data integration applications such as an ETL process. Access to application sources and targets is done via Web services, Microsoft .NET interfaces, Java-related capabilities such as JMS, legacy application interfaces and adapters, etc.

EAI is designed to propagate small amounts of data from one application to another. This propagation can be synchronous or asynchronous, but is nearly always done within the scope of a single business transaction. In the case of asynchronous propagation, the business transaction may be broken down into multiple physical transactions. An example would a travel request that is broken down in separate but coordinated airline, hotel, and car reservations.

Data transformation and metadata capabilities in an EAI system are focused toward simple transaction and message structures, and they cannot usually support the complex data structures handled by ETL products. In this regard, EAI does not compete with ETL.

EAI versus ETL

Although some vendors would have you believe otherwise, EAI and ETL are not competing technologies. There are many situations where they can be used in conjunction with each other – EAI can act as an input source for ETL, and ETL can act as service to EAI.

One of the main objectives of EAI is to provide transparent access to the wide range of applications that exist in an organization. An EAI-to-ETL interface could therefore be used to give an ETL product access to this application data. This interconnection could be built using a Web service or a message queue. Such an interface eliminates the need for ETL vendors to develop point-to-point adapters for these application data sources. Also, given that EAI is focused on real-time processing, the EAI-to-ETL interface can also act as a real-time event source for ETL applications that require low-latency data. The interface can also be used as a data target by an ETL application.

Although several ETL and EAI vendors have announced marketing and technology relationships, the interfaces they provide are often still in their infancy. Potential users need to evaluate carefully the functionality and performance of these interfaces. It is expected, however, that the quality of these interfaces will steadily improve. At present, instead of using a dynamic EAI-to-ETL interface, many organizations are using EAI products to create data files, which are then input to ETL applications.

In the reverse direction, EAI applications can use ETL as a service. Several ETL vendors already allow developers to define ETL tasks as Web services. These ETL Web services can be invoked by EAI applications. This not only adds additional transformation power to the EAI environment, but also supports code and metadata reuse.

Enterprise data replication

Several other data integration technologies are worth mentioning. Data replication, for example, supports both the data propagation and changed data capture approaches to data integration.

Although EDR is not as visible as ETL, EII or EAI, it is nevertheless used extensively in data integration projects. One of the reasons for this lack of visibility is that EDR often is packaged into other solutions. All the major relational DBMS vendors, for example, provide data replication capabilities. Also, companies offering CDC solutions often employ data replication. EDR is used not only for data integration, but also for backup and recovery, and data mirroring and workload balancing scenarios.

EDR tools vary in their capabilities. Replication tools often employ database triggers and/or recovery logs to capture source data changes and propagate them to one or more remote databases. Using recovery logs has less impact on source applications. In most cases propagation occurs asynchronously from the source transactions that produce the updates. Some EDR products, however, support two-way data synchronous propagation between multiple databases. Several also allow data to be transformed as it flows between databases.

One of the more significant differences between EDR and EAI is that data replication is designed for the transfer of data between databases, whereas EAI is designed for the movement of messages and transactions between applications. EDR typically involves considerably more data than EAI.

Integrating unstructured data

Most of the data integration technologies discussed so far focus on structured data. This is changing, however. Several EII vendors now provide federated access to unstructured data sources, particularly text-based documents. ETL vendors are also working on the processing of unstructured data.

Applications that employ ETL and EII to process unstructured data often need to integrate or relate the results to structured information. An example would be a marketing application that retrieves product sales analytics and related product information about advertising and market surveys.

Another technology that handles the integration of unstructured data is Enterprise Content Management (ECM), which is focused on the consolidation of documents, Web information, and rich media. ECM products concentrate on the sharing and management of large quantities of unstructured data for a wide user population. These products add a content management layer on top of a shared data store. This layer provides metadata management, versioning, templates, and workflow.

An ECM content store can act as a data source for an EII or ETL application. The key here is not simply to provide access to unstructured data, but also to access the metadata that describes the structure, contents, and business meaning of that data. This is analogous to the issues associated with accessing and integrating packaged application data where the metadata is again important to understanding the business meaning of the data. In both cases, it is important to evaluate not only what data and application sources are supported, but also the level of integration with the source data and metadata.

Data integration applications

A data integration strategy and infrastructure must take into account the application, business process, and user interaction integration strategies of the organization. One industry direction here is to build an integrated business environment around a service oriented architecture (SOA). In an SOA environment business process, application, and data activities and operations are broken down into individual services that can interact with each other. Often an SOA is implemented using Web services because this technology is generally vendor platform independent and easier to implement than earlier SOA approaches.

Master data management and customer data integration

MDM does the job of providing and maintaining a consistent view of an organization's reference data, which may be scattered across a range of application systems. The type of data involved in this process varies by industry and organization, but examples include customers, parts, employees and finances. Most MDM applications at present concentrate on handling customer data because this aids the sales and marketing process, and can thus help improve revenues. New buzzwords here include customer data integration (CDI), customer identity management (CIM), and customer master data management (CDM). Personally, I prefer CDM.

MDM and CDM are often discussed as technologies, but in reality they are business applications. The objective of both MDM and CDM is to provide a consistent view of dispersed data. This view is created using underlying data integration techniques and technologies, and may be used by business transaction applications and/or analytic applications. The actual techniques and technologies used will depend on application requirements, such as data latency and the need to update or just read the integrated data. What MDM and CDM add to data integration are the business semantics about the reference data as it relates to the business domain and industry involved. The value of the MDM or CDM solution therefore arises not only from the technology platform provided, but also from the power of the business semantic layer. MDM and CDM data stores can act as data sources for data warehousing applications.

Defining the business meaning of data in MDM applications is complex and requires a thorough understanding of how the data is used throughout the organization.

Developing a data integration strategy

The lack of an enterprise-wide approach to data integration is becoming a major inhibitor to new application development in many organizations. To solve this problem, organizations should have a long-term objective to create a flexible enterprise data integration architecture that provides the techniques, technologies, and tools to support new data integration projects. The architecture should evolve over time as new application requirements are uncovered, and as new data integration technologies and products are introduced. This architecture is especially important for organizations that have a complex heterogeneous data environment involving large volumes of data.

There is more to creating a data integration strategy, however, than just building an enterprise data integration architecture. There is also the need to share skills across data integration projects and to capture best business practices.

As business transaction, business intelligence, and business collaboration processing become more intertwined, there will be the need solve political problems and also possibly to reorganize the IT organization to bring together the various factions involved not only in data integration, but also other enterprise integration disciplines.

Many companies are developing their enterprise data integration strategies using the services of a *data integration competency center*. The center's objective is not only to design and support an enterprise-wide integration architecture and to provide a shared pool of data integration skills and resources, but also to bring together all of the organization's business integration disciplines into a single group.

The IBM data integration solution

In this section of the paper we examine IBM's key data integration products. The objective is not to provide an in-depth product guide, but instead to give an overview of the main features of each product, and to review how each of them supports the data integration techniques and technologies discussed above.

IBM's data integration solutions consist broadly of three related product platforms:

- **DB2® Content Manager** *for the management and consolidation of unstructured and semi-structured data such as digital media, Web content, and corporate and workgroup documents.*
- **WebSphere® Information Integrator (WebSphere II)** *for the federation, propagation and searching of structured, semi-structured, and unstructured data.*
- **WebSphere Data Integration Suite** *for data quality improvement, and the consolidation and propagation of structured and semi-structured data.*

Technique	Product	Technology	Type of Data
Federation	WebSphere Information Integrator	EII	Unstructured Semi-structured Structured
Consolidation	DB2 Content Manager	ECM	Unstructured Semi-structured
Consolidation	WebSphere Data Integration Suite	ETL ETL	Semi-structured Structured
Propagation	WebSphere Information Integrator	EDR	Structured
Propagation	WebSphere Data Integration Suite	EAI	Semi-structured
Search	WebSphere Information Integrator	(Crawlers)	Unstructured Semi-structured Structured
Changed Data Capture	WebSphere Information Integrator	(DBMS log DBMS trigger)	Structured

Figure 3. IBM product support for data integration techniques and technologies.

Figure 3 summarizes the technologies supported by each product platform and the type of data handled by each approach. Overviews of the features provided with each product family are presented below. It is important to note that for brevity and ease of reading not all the data sources supported by the various products are listed in the overviews. IBM should be contacted for a full and up to date list of product features.

IBM DB2 Content Manager

IBM's enterprise content management (ECM) solution for consolidating unstructured and semi-structured information is provided by the DB2 Content Manager. This product set provides digital media management, Web content management, document management, and records management. A detailed discussion of this solution is beyond the scope of this document. Please visit the IBM Web site (www.software.ibm.com/data) for more details.

IBM WebSphere Information Integrator

WebSphere Information Integrator (WebSphere II) is often viewed in the industry as a federated data server, but in reality it offers not only data federation, but also data propagation, and an enterprise search capability. We'll first look at its data federation capabilities, and then examine its other data integration features.

Data federation

The WebSphere II data federation capability allows applications to access and integrate diverse structured, semi-structured, and unstructured data as though it were a single resource, regardless of where the information resides. This federation capability is supplied with the following product editions.

WebSphere II Standard Edition enables federated SQL query access to structured and semi-structured data. Data stores supported include relational systems such as IBM DB2 and Informix®, Microsoft SQL Server, Oracle, Sybase and Teradata, and semi-structured data like Microsoft Excel files, Web services, WebSphere MQ messages, and XML documents. Any system that provides ODBC or OLE DB interfaces can also be accessed. Relational database, WebSphere MQ and Web services data providers can also be updated. All other data providers are read-only. A supplied development kit allows custom code to be developed for supporting additional data stores. Key data federation features provided with the product include cost-based query optimization and integrated caching. The Advanced Edition of WebSphere II offers the same capabilities as the Standard Edition plus an unrestricted license for IBM's DB2 UDB relational DBMS.

WebSphere II Content Edition allows applications to access multiple content repositories and workflow systems through a single bi-directional interface. The repositories may contain documents, images, audio, video, and other unstructured information. The WebSphere II Content Edition enables these disparate unstructured content sources to look and act as one system. Tasks supported include check in, check out, view and modify content and metadata, workflow handling, information mining, etc. Vendor content stores supported include IBM (Content Manager, Lotus Notes,® Lotus Domino®), EMC (Documentum), Filenet, Open Text, Interwoven, Stellent, Hummingbird, and Microsoft (Index Server). A supplied toolkit lets organizations develop, configure, and deploy content connectors to additional commercial and proprietary repositories. Sample connectors are provided in the kit for accessing Google and file systems.

WebSphere Classic Federation Edition for z/OS® allows Windows and UNIX applications to use SQL statements to access to mainframe databases and files. These JDBC or ODBC SQL statements are dynamically translated by the product into native read and write API calls. Data stores that can be accessed include IBM VSAM and IMS®, CA-IDMS, CA-Datacom, Software AG Adabas, and DB2 UDB for z/OS. The product is driven by user-defined metadata that maps physical databases and files to virtual relational tables.

Federated database design

One important product that is related to the WebSphere II data federation capabilities is the **IBM Rational® Data Architect**. This product provides an integrated tool set for data modeling and data integration design. It combines traditional data modeling capabilities with metadata discovery, mapping, and analysis. The Rational Data Architect is appropriate for traditional data modeling tasks as well as simplifying the design and development of federated databases. It allows designers to define and maintain enterprise views of logical data models, discover relationships among existing databases, and create a target federated schema.

Data propagation

The data propagation capabilities of WebSphere II are supplied through the WebSphere II Replication Edition and the WebSphere II Event Publisher Edition. Both editions are included with the WebSphere II Standard Edition.

WebSphere II Replication Edition propagates and synchronizes information across multi-platform and multi-vendor environments. It provides two different options for replicating data from and to relational databases.

- **SQL replication** *where committed source changes are staged in relational tables before being replicated to target systems. Source and target database systems supported are DB2, Informix®, Microsoft SQL Server, Oracle and Sybase. Teradata is also supported as a target.*
- **Queue replication** *where committed source changes are written to messages that are transported through IBM WebSphere MQ message queues to target systems. Queue replication is a new, high-speed technology for moving transactions between DB2 database systems, and from DB2 to targets such as Oracle, Microsoft SQL Server, Informix and Sybase. Replication to third-party relational systems is done using the federated server capabilities of WebSphere Information Integrator.*

With SQL replication, source data changes are captured using either a log-based or database trigger mechanism and inserted into a relational staging table. An *apply* process asynchronously reads the changes from the staging table and handles the updates to the target systems. Target systems are usually read-only databases, such as a data warehouse. Data movement can be continuous, event-driven, or automated on a specific schedule, or at designed intervals. SQL expressions and stored procedures can be invoked to do data transformation during the replication process.

Queue replication augments, but does not replace SQL replication. It is well suited to on-demand applications where the lag time between a source data change occurring and the target being updated has to be minimized. Unlike SQL replication, it also provides bi-directional replication.

With queue replication, the *capture* program runs on the source system, reading DB2 recovery logs for changed source data, and writing it to WebSphere MQ queues. The *apply* engine determines transaction dependencies and replays transactions on the target system with the objective of maximizing parallelism and minimizing latency. Stored procedures can be used to transform the replicated data before it is applied to the target system.

Data propagation is also provided by the **WebSphere II Event Publisher Edition**, which captures database changes as they occur on the DB2 UDB recovery log, formats them into XML messages, and publishes them to WebSphere MQ for use by other applications. Any application or service that integrates with WebSphere MQ, or supports Java Message Service (JMS), can asynchronously receive the data changes as they occur. This facility can be used to provide information to information brokers and Web applications, or to trigger actions and processes that are based on updates, inserts, or deletions to source data.

The **WebSphere II Classic Event Publisher** extends the capturing of database changes to include CA-IDMS, IBM CICS® VSAM and IMS data sources.

Enterprise search

Enterprise search is a new feature of WebSphere II that is provided by the **WebSphere II OmniFind™ Edition**. This search capability is used to locate enterprise information stored in file systems, content archives, databases, collaboration systems, and applications. It performs content crawling, parsing and tokenizing, categorization, annotation, indexing, and searching. In addition to supporting the WebSphere II Content Edition and the DB2 Content Manager, OmniFind enables access to a variety of other content sources, including Web sources, news groups, Microsoft Exchange public folders, and relational database products such as DB2, Informix, and Oracle. The product also provides a search application that plugs into and works with the Google Desktop Search for Enterprise interface.

The OmniFind search capability can integrated into the IBM WebSphere Portal using WebSphere II OmniFind for WebSphere Portal, which allows organizations to leverage existing portal taxonomies for content navigation and categorization.

In addition to enterprise search, the WebSphere II OmniFind Edition also offers a text analysis facility, which can be used to extract concepts, facts, and relationships from text files. Third-party and external applications can access the text analysis engine through the IBM Unstructured Information Management Architecture (UIMA) interface.

What is UIMA?

UIMA is a software framework that supports the creation, discovery, composition, and deployment of a broad range of text analysis capabilities, and the ability to connect them to information services such as search engines and databases. The UIMA framework provides a run-time environment that enables text analytics components from multiple vendors to work together. IBM is proposing to give UIMA to the open source community.

Text analytics is used to analyze documents, comment and note fields, problem reports, e-mail, Web sites and other text-based information sources. The extracted information may be used, for example, to enhance the quality of search results, or to add text analytics to traditional business intelligence and data warehousing applications.

IBM WebSphere Data Integration Suite

The WebSphere Data Integration Suite supplies a data integration platform for consolidating structured and semi-structured data, and managing data quality. This suite is the result of IBM's acquisition of Ascential Software. In 2006, IBM is introducing significant architectural and functional changes, and creating a base for integrating the suite with IBM's other information integration products.

Two important new pieces of infrastructure are in the 2006 release, code-named *Hawk*. The first is a new metadata capability for coordinating all of the data used by the products in the data integration suite. The second is a foundation for a new, simplified user interface that will ultimately be leveraged across the entire product portfolio to make the software easier to use.

The objective of the improved architecture is to deliver a data integration platform that provides a shared metadata repository, metadata services with bi-directional metadata interchange, J2EE-based platform services, and a common parallel run-time engine. This architecture is shown in Figure 4.

The products that make up the WebSphere Data Integration Suite are reviewed below.

WebSphere DataStage® offers a scalable data integration engine that collects, transforms, and consolidates large volumes of source data. The product handles a wide range of source and target systems, including most database products, flat files, packaged applications such as PeopleSoft, SAP and Siebel, XML data, and Web services. Data collected by WebSphere DataStage may be received on a periodic or scheduled basis, or may arrive in near-real-time. The near-real-time collection facility captures messages from a Java Messaging Service (JMS) and WebSphere MQ message queues.

The **WebSphere DataStage SOA Edition** provides a service oriented architecture (SOA) for publishing WebSphere DataStage integration logic as shared services. This allows companies to develop libraries of data integration services that can be listed in a shared directory, and reused from project to project. These services can be called from any process or application using standards like Web services, JMS, or Enterprise Java Beans.

WebSphere DataStage MVS™ Edition provides native data integration capabilities for mainframe data. It supports the consolidation of legacy data with other enterprise data. It generates COBOL applications and custom JCL scripts for processing flat and VSAM files, and DB2, IMS and Teradata databases.

WebSphere DataStage for z/OS supports Unix System Services (USS) on IBM z/OS servers. This product is similar to the DataStage MVS Edition in that the design tools are used to generate DataStage jobs. Once generated, the DataStage jobs are moved to the USS environment for execution.

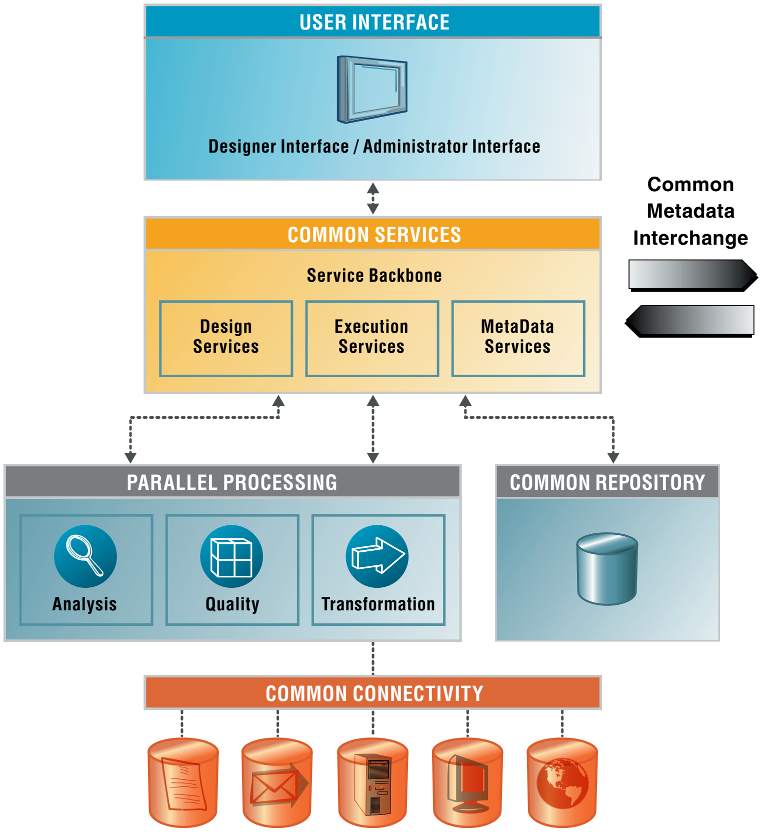

Figure 4. WebSphere Data Integration Suite architecture.

WebSphere Metadata Services is a component of WebSphere Data Integration Suite that includes an enterprise metadata directory, and supports the bi-directional exchange of metadata between leading data modeling, data quality, ETL, data profiling, and business intelligence tools. Its Web-based interface gives IT professionals and business users a reporting and search capability for accessing the metadata stored and managed in the repository. The interface gives users a graphical view of repository metadata and includes both data lineage reporting and impact analysis for data changes.

WebSphere Business Glossary is a new Web-based tool for business users to create and manage a business taxonomy or vocabulary. The tool can be used for documenting and collaborating about the meaning, dependencies, usage, quality and owners of business data.

WebSphere Information Analyzer is a new data content profiling, quality monitoring and auditing tool. It is used to validate and analyze source data values and column/table relationships. It facilitates source-to-target field mappings, does target database definition generation, and allows analysts do detailed exploration of exception data. It provides the ability for analysts to add business names, descriptions and other attributes to tables and columns. It also includes an integrated data quality methodology and customizable data quality dashboards.

WebSphere QualityStage™ is a data quality improvement tool. It provides analysts with a point-and-click interface for defining automated data quality validation and matching tasks. These tasks can employ pre-built objects and tables that can be customized for data quality operations. It includes capabilities to deal with phone numbers, email addresses, birth dates, events, and other comment and descriptive fields. It also produces metrics reports for supporting quality assurance programs. WebSphere QualityStage processing tasks can be integrated into real-time processes using either the SOA Editions of the WebSphere DataStage product set or standalone C or Java applications.

Master Data Management

As discussed earlier in this paper, Master Data Management (MDM) consists of applications that integrate and manage enterprise-wide master reference data for business entities such as customers, products, employees, finance, etc. Vendor solutions in this area currently cover a broad range of capabilities, but the industry direction is to build a complete MDM environment for managing an organization’s reference data for both operational and analytical processing purposes. IBM MDM strategy and development plans are consistent with this industry direction.

The data integration products reviewed in this paper form the backbone of a *master data integration* architecture. MDM applications can be built on top of this architecture.

The **WebSphere Product Center**, for example, uses the IBM data integration product set to supply companies with a repository for managing and linking information about products, locations, trading partners, organizations, and terms of trade. It also enables the propagation and synchronizing of this information across existing enterprise systems and external trading partners. IBM intends to offer similar capabilities for other MDM business areas.

To help build out its MDM solutions, IBM recently introduced WebSphere Customer Center, based on the acquisition of DWL; a leading provider of customer data integration middleware to companies in the banking, insurance, retail and telecommunications industries. WebSphere Customer Center aggregates multiple sources of data to provide a single integrated view of prospects and customers. The product delivers a single, real-time view of customer information and provides a set of business services to maintain and propagate this information to source systems. WebSphere Customer Center contains a J2EE service oriented hub architecture that comes with some 300 pre-built Java business services for handling customer data integration.

Connecting the pieces together

There are various ways of combining and using these product sets to support a complete data integration environment. Outlined below are some examples.

- *Unstructured and semi-structured information managed by the DB2 Content Manager can be indexed and searched using WebSphere II OmniFind.*
- *WebSphere Information Integrator can provide a federated view of DB2 Content Manager information. This federated view may be used by the WebSphere Data Integration Suite to access unstructured and semi-structured information.*
- *Structured and semi-structured data managed by multiple data managers in an organization can be presented in federated views to the WebSphere Data Integration Suite using WebSphere Information Integrator.*
- *WebSphere Information Integrator can be used to capture and propagate data changes to the WebSphere Data Integration Suite via WebSphere MQ.*
- *The data transform library of the WebSphere Data Integration Suite can be called from WebSphere Information Integrator.*
- *Web services support allows many of the capabilities provided by the IBM products discussed in this paper to participate in an organization's services-oriented architecture (SOA).*

Metadata considerations

The issue with complex data environments is not only the integration of data, but also the management and integration of metadata. Despite many efforts to do so, no single organization or vendor has ever completely solved this problem. Today, most approaches support metadata integration by replicating data between systems.

The Hawk release of the WebSphere Data Integration Suite provides a single repository for managing Suite metadata. A metadata interchange mechanism allows repository metadata to be exchanged with other products and applications. This mechanism will be used initially by IBM to share metadata with WebSphere Information Integrator and the Rational Data Architect. IBM's direction here is to provide a single metadata management environment for its information integration product platforms.

Conclusion

The direction of many organizations is to develop and deploy an enterprise-wide architecture for supporting data integration projects. This architecture is shown in Figure 1 at the beginning of this paper. When combined, the IBM DB2 Content Manager, WebSphere Information Integrator, and WebSphere Data Integration Suite product platforms support all of the major components of this architecture, which will enable IBM to strengthen its position as one of the leading information integration software suppliers to enterprises.

About BI Research

BI Research is a research and consulting company whose goal is to help companies understand and exploit new developments in business intelligence and business integration. When combined, business intelligence and business integration enable an organization to become a smart business.

BI Research
Post Office Box 398
Ashland, OR 97520
Telephone: (541)-552-9126
Internet URL: www.bi-research.com
E-mail: info@bi-research.com

IBM Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
02-06
All Rights Reserved

[1] Colin White. "Data Integration: Using ETL, EAI, and EII Tools to Create an Integrated Enterprise." TDWI Research Report, November 2005.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates. Offerings are subject to change, extension or withdrawal without notice.

All statements regarding IBM future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only.

The IBM home page on the Internet can be found at **ibm.com**