



Product White Paper

Overview

Many organisations recognise a disparity between the amounts of data they store or have available to them, and the extent to which they can use this information to gain Competitive Advantage. Metabula bridges this gap with intelligent software that releases the inherent value of structured data, overcoming the fact that different data sources look, talk and act in different ways.

Large amounts of structured data are already held within desktop, legacy and corporate databases, as well as applications such as ERP, CRM, etc. The next phase of development of the Internet will create an explosion in the volumes and diversity of structured information available. This is already happening, fuelled by structured XML and on-line Web Services (allowing programmatic access to resources across the Internet using the 'SOAP' protocol). Availability of data is not the issue. Having more and more data is not necessarily a useful thing, and delivering basic access is not sufficient. The winners will be the organisations who can connect people, applications and devices to manageable amounts of high quality, relevant information in a timely manner and meaningful context.

Already there are challenges in managing the proliferation of databases and other data sources internally within an organisation – research has reported that 30% of all IT expenditure worldwide is spent on integrating different systems and that for every \$1 spent on market- or partner-facing e-business initiatives, between \$5 and \$50 is spent to integrate this internally [Gartner Group, 2000]. The Internet has provided additional opportunities, but has not solved existing problems. Organisations have many systems fulfilling dedicated functions, but cannot call upon consistent information from across the enterprise. Further, in the Digital Economy the seamless integration of information also needs to extend across partners, suppliers and customers.

Metabula DataBridge is a non-invasive technology that fits within this scenario to solve a common problem where:

- You have data in multiple sources
- You want to use their information in a cohesive way
- You need to query and update the data
- You do not want move the data
- Creating a new system is not appropriate

DataBridge is a rapidly deployed technology, making it an attractive solution to common, database related problems within an organisation as well as a component in new application or Internet initiatives.

What is Structured Data?

If we use an analogy of a box of chocolates, then our structure (or 'meta-data') is the layout of the box and the definition of which chocolates go where - the chocolates that sit in this structure represent the data. This is a simplistic but effective analogy. The benefit of storing our data / chocolates within a specific structure is that it allows us to see what we have - in the case of the chocolate box, it means that we don't have to try each chocolate to find the one we want. But there are differences in the content and layout amongst different manufacturers and brands, and in the data world this is where DataBridge comes in.

What does DataBridge do?

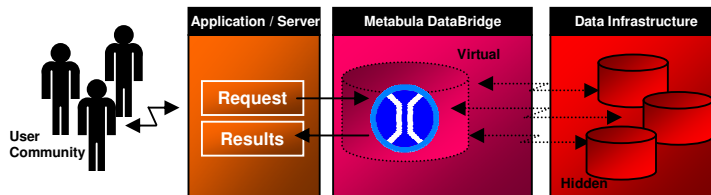
DataBridge is an intelligent data infra-structure technology. It provides a cost effective solution to data access and integration across multiple data sources.

The purpose of DataBridge is to provide transparent access to the combined information and knowledge represented across disparate systems. It hides the internal configuration details and access methods of each source, and accommodates changes dynamically over time.

DataBridge does not move or replicate data amongst sources, and the only data it stores is *meta-data*. Meta-data is 'data about data' – it provides the structure used to store and organise data.

DataBridge allows you to focus on *using* the information you need in a meaningful form, not where and how it is stored and structured. DataBridge interacts with the content of different data sources through a common structure ('logical data model') that provides consistent meaning, or *semantics*, to the information. You define the data model to suit your needs (or download an XML schema from a trading partner).

DataBridge separates the *use* of information from the *storage* of data. From a technical perspective, it allows you to de-couple 'business logic' from 'data services'. We will see later how the logical data model supports this, and provides insulation and portability across 'physical' sources that may be changed or replaced.



DataBridge may be used programmatically as a component in the architecture of new e-business initiatives, or web applications, or can be used to extend the scope of existing applications across different back end data sources. DataBridge plays a common role in all deployment scenarios - it delivers information in a consistent and usable form to applications and users that that simply want to make use of information, irrespective of where data resides, or how it is stored.

How can I visualise DataBridge

Think of DataBridge as a 'virtual database'. Picture it as an intelligent engine that uses knowledge of how and where data is stored ('meta-data') to dynamically integrate data that is physically stored in several applications. DataBridge hides the complexities of what is stored where, and how it is stored. It allows you to work with the information on your own terms, as if it were a single source.

The DataBridge Philosophy

The core competence of DataBridge is to provide information to users and applications. Its value is in connecting users and applications to complex data in disparate sources. It does this simply, with flexibility, higher performance, greater scalability, and more efficiently, than 'spaghetti' hard-coding. Applications, web servers or custom code can delegate to DataBridge the complexities of managing connections to data sources, interrogating schema, reconciling different structures, formatting queries in native SQL dialects, resolving inconsistencies, and iterating and integrating results sets.

As well as incorporating technology to overcome the technical hurdles of accessing data in different applications on different platforms in a heterogeneous environment, DataBridge seamlessly integrates data that lacks common semantic structure into a consistent, meaningful context.

The DataBridge philosophy is to:

- Provide data integration, non-invasively, across multiple sources
- Leave data where it is, avoiding unnecessary exchange and replication
- Provide transparent access with common semantics - 'logical data model' - across changing sources
- Optimise performance by distributing query processing
- Eliminate hard-coding through a meta-data driven implementation
- Leverage the standards of the next phase of the Internet (XML, SQL, OLE DB / ADO, SOAP, UML)

DataBridge builds upon key principles developed in the field of knowledge representation – developed within Cambridge University, international standardisation bodies and commercially within Metabula - to provide information in a consistent context. This overcomes a common problem with databases – there are many different ways of representing the same information; often there are no right or wrong solutions, just differences.

DataBridge is positioned to take advantage of the revolution in broadband data access, wireless applications, WAP, XML, web services, etc. that will allow it to play a pivotal role within the next generation of loosely coupled application architectures, connecting users to integrated service and data providers.

Examples

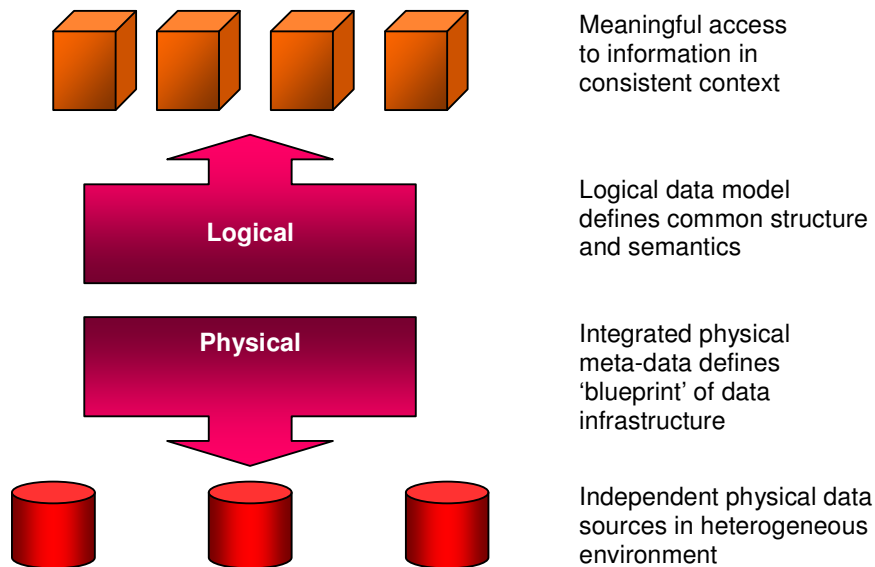
Let's look at a few example scenarios:

- **Implementation of new web application**
DataBridge can reduce the time and cost of implementation and de-skill the process. Implementing a logical data model between the application database and web server / business logic components also offers a lower cost of ownership through portability and the ability to integrate changes to data content dynamically. It allows the scope of data to be extended, as additional sources become available, or to respond to market demand for a 'richer' experience. Within a B2B context, DataBridge's XML toolkit provides powerful data exchange functionality and integration with Microsoft BizTalk.
- **Publishing of data**
DataBridge is non-invasive, allowing data in existing applications to be integrated and brought on-line in a usable format without impacting their operation or availability. 'Live' up to date data can be published, without batch exchange or replication, and you can provide integrated data from more than one source. Again, DataBridge's XML interface provides complimentary support for data exchange with partners.
- **Extending application scope**
There are hundreds of desktop tools that can connect to a database using a Microsoft OLE DB interface. DataBridge can extend the scope of these tools by allowing them to work seamlessly with information from more than one source. Solutions developed using these packages via DataBridge's logical model are portable across different back end data sources.
- **Custom Code / Application Development**
If you are writing your own code, then DataBridge allows you to simplify complex interactions with data sources. This area is the preserve of skilled software engineers and by using DataBridge you can make more effective use of this scarce resource elsewhere. DataBridge aims to eliminate and outperform custom coding, and provide a flexible solution with lower costs of implementation and ownership. Using DataBridge means that you can access a wide variety of sources without installing specific middleware for each one; it manages the connections to different data sources; it hides internal data structures; it handles variations in native SQL 'flavours'; and it distributes query processing to make the most effective use of available resources and minimise network traffic.
- **Prototyping and Evaluation**
DataBridge is shipped with tools to evaluate data quality (DataInspector); it supports UML; and integrates with design tools such as Rational Rose. As it can be deployed rapidly and non-invasively it provides a powerful tool to support prototyping and the definition of data models, a data architecture (i.e. what to move, what interfaces to build, what to leave where it is, etc.), and an implementation strategy.

How Does DataBridge Work

DataBridge uses a meta-data driven approach that hooks the structure and layout of data in existing systems into a common frame of reference – in DataBridge terminology, this is known as the ‘logical data model’. DataBridge can manage a catalogue of logical models, supporting different perspectives.

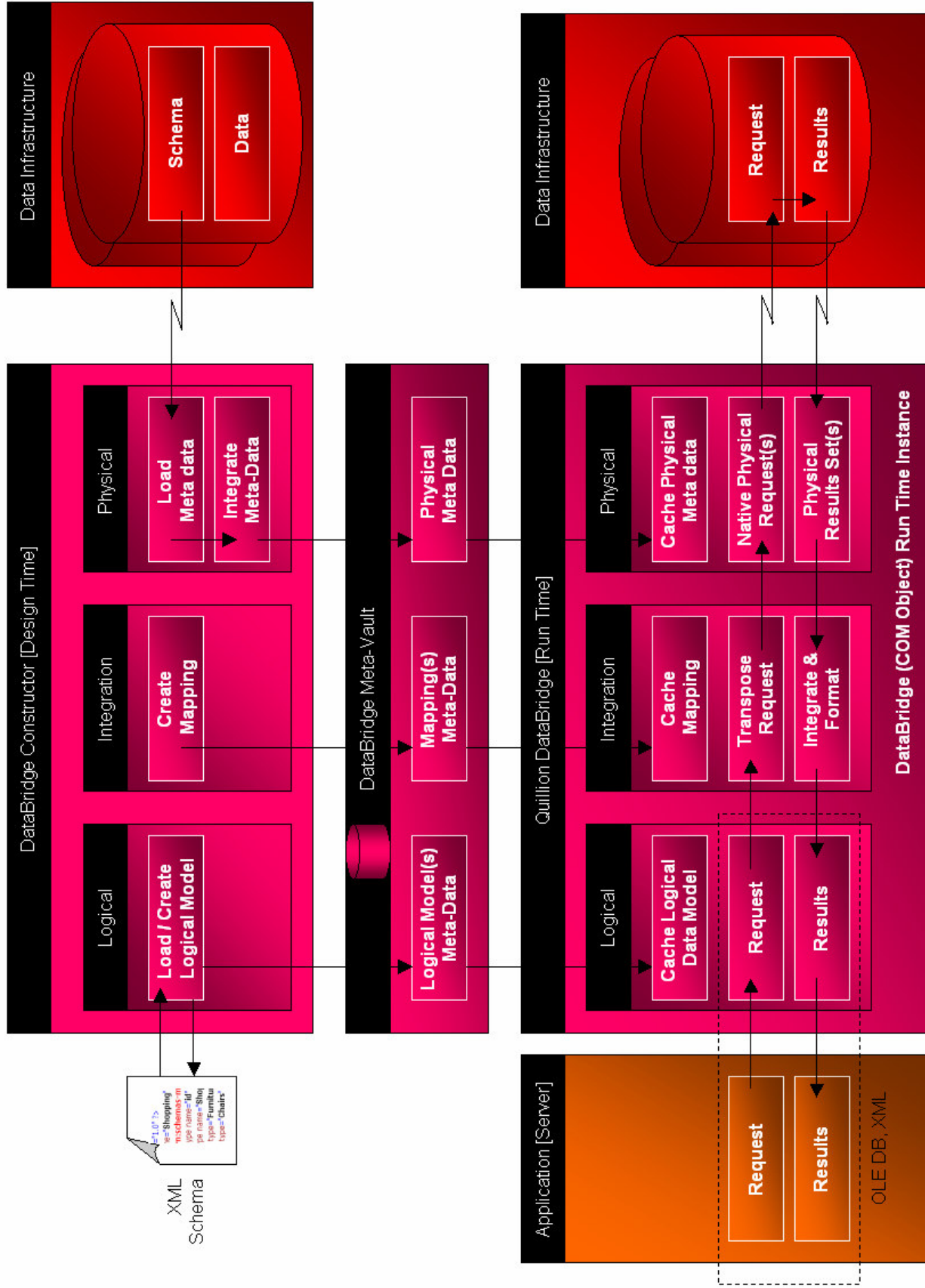
DataBridge’s logical data model approach provides transparent data access, regardless of the location, structure or format of the physical source. Front end consumers of the data talk to the logical data model, and are insulated from the back end data sources. This means that investment in front end user interface or business logic implementation is protected from changes in the back end data storage. In this way, DataBridge delivers flexible, future proof solutions and also provides portability across different sources.



The logical data model(s) in DataBridge are defined in data, not hard-coded, so they can be extended and changed at any point – this opens the door to innovative data driven solutions to provide dynamic data content for web sites or applications.

DataBridge Architecture

DataBridge is shown in schematic form on the following page, and then explained through a brief walkthrough.



DataBridge Walkthrough

DataBridge can be described by splitting its operation into design-time and run-time aspects.

- **Design-Time**

BridgeConstructor is a graphical (GUI) tool that is used to define the layout of data in each physical source, the details of the logical data model(s) and the relationships between the logical and physical. This information is stored as meta-data in the DataBridge Meta-Vault. This meta-data is used to control the operation of the DataBridge at run-time. Since all the controlling information is held as data, this allows DataBridge to absorb changes dynamically from any of the physical sources, the logical data models, or mappings.

Physical schema are loaded automatically – once in the meta-vault, associations can be defined between any tables and views, irrespective of where they reside. The logical data model can be created using BridgeConstructor, imported as XML, or built bottom-up, on-the-fly by dragging fields from the physical schema. Mapping information is created manually through a drag and drop process.

- **Run-Time**

When it is launched, the DataBridge Engine caches the information from the meta-vault. When DataBridge is called, it uses the meta-data to tell it where and how to retrieve (or update) what data, from what sources, and in what order. Each source passes back its own native results that are formatted and integrated by the DataBridge Engine into a consistent structure [context] defined by the logical data model. To whoever or whatever made the initial call, DataBridge behaves like a ‘virtual’ database - you make a request and DataBridge gives you what you asked for, hiding the internal details of the underlying sources.

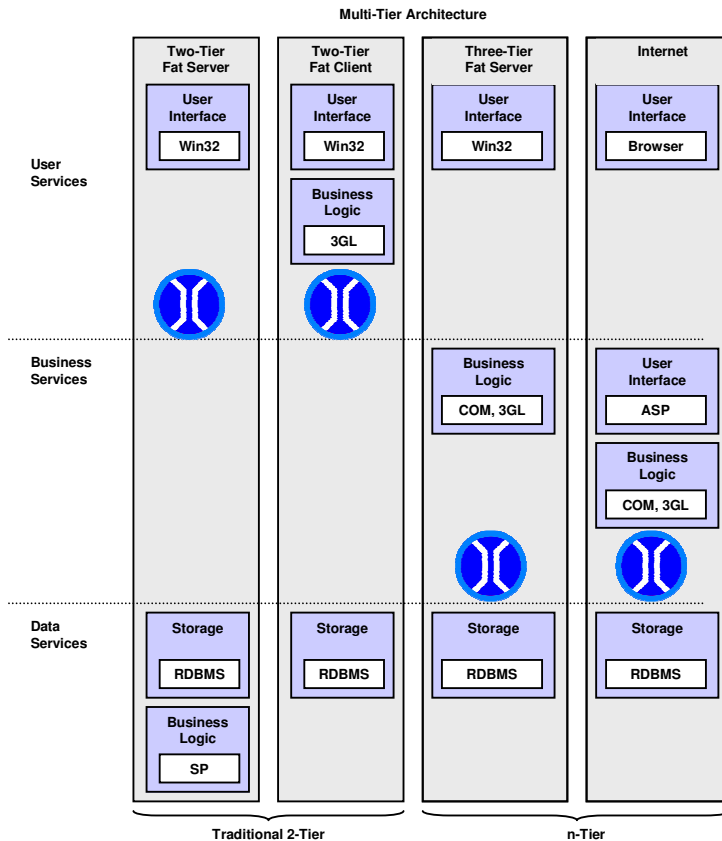
To complete the picture in terms of DataBridge’s modules:

- BridgeAdmin provides a set of administration functions, including user account management.
- The DataBridge XML Toolkit supports the import / export of logical data models as XML schema at design time, and the export of data as XML at run-time. The XML export supports features that allow users to filter and ‘walk’ the logical data model to select individual data items, and build these into a compact ‘package’ for exchange.

DataBridge is also shipped with DataInspector. This allows you to perform a ‘health check’ on the quality, format and consistency of data sources.

Typical installation

DataBridge operates on a Microsoft platform, but can access data from a heterogeneous environment of different hardware, operating systems, and applications. This covers a typical mixed Microsoft, UNIX and Legacy scenario. DataBridge can be deployed on a client machine, or with an application or web server.



The n-tier model emphasises a logical architecture for applications rather than a physical one. Any service may invoke any other service and may reside anywhere. Applications are distributed, which means you can run the right components in the right places, benefiting users and optimising use of hardware resources.

How does DataBridge differ? FAQ's.

Here are some Frequently Asked Questions (FAQ's) that clarify what DataBridge is and what it *is not*.

Is DataBridge an EAI tool?

Enterprise Application Integration (EAI) is a very broad heading. It is possible to consider DataBridge under the banner of EAI, but DataBridge is radically different to traditional EAI tools (that move data from one place to another). DataBridge offers integrated data from different sources *without moving the data* and provides the flexibility for these sources to change and adapt over time.

How does DataBridge differ from ETL?

Extract-Transform-Load (ETL) software operates a batch 'upload' from point-to-point and is often used to move data from operational systems into a custom built Data Warehouse (DWH). Moving data from A to B is not the role of DataBridge. DataBridge operates in scenarios where you do not wish to move or replicate data from its original sources, or where you do not have a target system to put it in.

How does DataBridge differ from ‘Publish and Subscribe’?

‘Publish and Subscribe’ middleware technology implements a data ‘pipeline’ to support *process* integration of different applications – i.e. A passes to B, B passes to C, C passes to D and E, etc. Publish and Subscribe packages can provide guaranteed message delivery amongst their own agents, but cannot influence how the information is then used by the application or by users. Their suitability varies according to the nature of the integration, and the freedom that users have to ‘compromise’ the data in each application. Challenges in process integration are the loss of synchronisation between applications (because of what users do, i.e. lack of discipline – this is invisible to the process integration layer), difficulties in error monitoring and history tracking, and lack of data comparison facilities. DataBridge can work with process integration packages (if required) to provide data comparison and an integrated *data* interface across different applications through the process.

How does DataBridge differ from Business Intelligence (BI)?

Business Intelligence (BI) packages vary from report generation and user-presentation of data to vertically integrated suites of modules with interfaces to proprietary applications such as ERP. BI tools use data on a read only basis, for their own purposes. DataBridge does not provide reports – it provides consistent data that *any* other application can connect to or use programmatically.

What about Microsoft BizTalk?

DataBridge integrates with BizTalk – the two work in a complementary manner, but are distinct products. BizTalk enables the exchange and routing of data in XML file format, and a means to share agreed XML schema.

Technology FAQ's

What platform does DataBridge run on?

DataBridge operates on a Microsoft XP or 2000. It can access data from a heterogeneous environment of different hardware, operating systems, and applications - MS, Unix and Legacy.

What sources can DataBridge connect to?

DataBridge can connect to any source with an OLE DB interface (leveraging Microsoft's Universal Data Access (UDA) framework). It can also connect via ODBC, and includes native connections for XML, Flat File, Delimited File as well as optimised native connectors for Oracle, SQL Server, DB2, Sybase, Informix, Btrieve, and Ingress.

What interfaces does DataBridge support?

DataBridge supports ANSI SQL92 through an OLE DB interface, and also supports an XML interface. The DataBridge XML Toolkit can connect to any OLE DB source, but is not sold separately.

How does DataBridge scale?

DataBridge scales by leveraging multiple clustered machines. It is multi-threaded, stateless and fully tested on multi-processor machines.

How is DataBridge called from within code?

DataBridge operates within a Microsoft WinDNA architecture, and is called via an OLE DB interface.

Is DataBridge read / write?

Yes, DataBridge is bi-directional.

Does DataBridge support two-phase commit?

Yes, DataBridge guarantees data integrity and consistency through a two-phase commit process for transactions spanning multiple sources. DataBridge can be enlisted within MTS (Microsoft Transaction Server) as a participant in a wider, external two-phase commit process.

How are the logical models defined?

Logical models are defined according to UML Static Class Diagrams (supporting Object Oriented concepts such as inheritance, classification and associations). These can be defined within BridgeConstructor, loaded directly from XML schema, or using CASE tools and formal methodologies such as Rational Rose.

What training is required?

DataBridge requires minimal training - it is designed to leverage standard technologies and interfaces throughout. A one-day introductory course is currently available, and is run on a regular basis.

Glossary

Structured / Unstructured Data

We used an analogy of a box of chocolates earlier – think of structured data as chocolates arranged neatly within a box, and unstructured data as the same chocolates tipped into a bag.

Structure data is data represented in context. Each distinct piece of information is ‘labelled’ and can be treated individually. However, the labels are entirely arbitrary - they vary from source to source, and the granularity and level of detail of labelling varies.

Unstructured data has no internal layout of cannot be broken into smaller distinct pieces of information. The contents of unstructured documents or sources are not labelled, and have no meaning.

Physical Data Source

A repository of data, as physically implemented – e.g. a database.

Logical Data Model

The consistent structure / semantics used by DataBridge to pull together data from different sources. The logical model is ‘virtual’ as DataBridge does not store any data [persistently] in this structure.

XML

Like HTML, eXtensible Markup Language is a subset of Standard Generalised Markup Language, a standard for defining descriptions of structure and content in documents. HTML defines layout and presentation only – XML labels content within a structure of ‘tags’.

SOAP

Simple Object Access Protocol – a means of calling a function (‘web service’) from a remote web site – allows you to ‘program the web’.

ADO / OLE DB

Both are de facto standard Microsoft programming interfaces. OLE DB is just used as a name nowadays. ADO – ActiveX Data Objects.

RDB / RDBMS

Relational Database [Management System] – the most common type of database used commercially.

SQL

Structured Query Language – the de facto query language for relational databases. The standard is ANSI SQL92, but as ever, all applications display subtle ‘native’ variations.

UML & OMG

UML – Unified Modelling Language – a standard Object Oriented data modelling and design methodology from the Object Management Group (OMG).

Further Details

Visit Metabula at www.metabula.com

Contact the author at info@metabula.com

For sales enquires contact sales@metabula.com

For press enquiries contact press@metabula.com

For partner and OEM enquires contact partners@metabula.com

