Data-modelling terminology and P_META

EBU Project Group P/Meta

Data analysis and modelling are relatively new disciplines in the broadcasting industry, but have become increasingly important because the digital convergence of media and information systems has raised the profile and value of "metadata" to organizations. Professional data analysts, entering the broadcasting industry from the information systems industry, have brought with them some well-established terms and techniques which are now entering regular use – and they need to be clearly understood.

This article – written by Andy Carter, a Data Analyst in BBC Technology's Media Data Group – attempts to provide some clear definitions for common information industry terms and approaches, and it positions the P_META Scheme v1.0 within this context.

Introduction

EBU Project Group P/Meta has been working since 1999 to create effective standards for the exchange of metadata between broadcasters. Recently, the work on this project has culminated in the publication of P_META version 1.0 which is a business-to-business semantic scheme – comprising a flat list of attributes which may be used to pass information meaningfully from one organization to another. As the list is in itself unstructured, P_META also contains a number of sets that enable the attributes to pass specific types of information by putting them into a specific context. It is also envisaged that groups of P_META users will also be able to build additional sets for their own purposes by using a defined notation to combine attributes from the flat list into meaningful sets.

Work to create the published version of P_META has been a major task, involving a number of participants from major European broadcasters and detailed analysis carried out over a three-year period. The final product is commonly referred to as a "Semantic Scheme" by its creators, but there has been some confusion and debate over the exact terminology which is in use by P_META. This confusion is rooted in the general confusion in terminology that is unfortunately endemic in both the broadcasting and information systems (IS) industries. The scope for confusion is further compounded by differences in terminology across organizational and cultural boundaries. Finally – and inextricably linked to the question of terminology – is the question of what are the appropriate techniques to use in the publication and implementation of the P_META Scheme.

Two questions are often asked about P_META:

- is the P_META Scheme a data model?
- should the P_META Scheme be represented as a data model?

The answer to both questions depends on the exact meaning of the term "data model". To answer the questions accurately, we need to understand both the meaning of the term "data model" and the context in which the question is asked – remembering that the EBU's wish is to create an architecture for metadata exchange which exists in the semantic, technology and application layers. This means that we also have to be clear about what is meant by each of the terms listed below.

Architecture for Metadata Exchange [1]

This is an umbrella term which the EBU has adopted to refer to the elements that are required to fully support their aim of being able to exchange media asset metadata efficiently between member organizations. The architecture is a collective term used to refer to the semantic, technology and application layers.

The semantic layer

The semantic layer is also known as the "descriptive metadata layer". This is concerned with the exact definition and meaning of each element within the metadata exchange architecture [2]. This definition is based on the business meaning and understanding of each concept and is expressed in terms of normal human language. The published version of P_META represents this layer.

The technology layer

The technology layer is concerned with the coding for storage [3] and transmission of the attributes and sets defined in the semantic layer. In simple terms, this represents the translation of the conceptual definitions in the semantic layer into a specific form for use in a particular type of technology. Thus it would include such products as a representation of the P_META scheme as XML or as a Descriptive Metadata Scheme (DMS) for use with the Material eXchange Format (MXF).

The application layer

The application layer (also known as the "data interchange layer") will contain actual applications which have been coded using the products available in the technology layer. So this layer would contain an actual application which had been created (using XML, for example) for the transfer of a specific kind of information between organizations.

The published P_META version 1.0 is the semantic layer of the overall metadata exchange architecture. Having understood that it represents an entirely semantic business-based view of exchange metadata, we now need to consider how this ties in with the concept of a data model.

What is a data model?

There is no universally agreed definition of what is meant by the term "data model". In the widest sense, a data model is almost any model that is constructed by the analysis of data. This sense of the term is in widespread use by mathematicians, statisticians and meteorologists, amongst others.

However, in the IS industry, a data model is generally understood to apply to a representation of the structure of, and relationships between, data elements in a given business scope – usually an organization or a part of an organization being supported by a system. Data models have been in common usage throughout the IS industry for well over twenty years. During this time, many different modelling techniques and methodologies have been devised, but most of their main features are common. Generally speaking, a data model is understood to

[1] EBU Technical Statement D93-2002: **Conveyance of information defined by the P_META Metadata Exchange Scheme in media file formats**, Geneva, 2002.

[2] This should not to be confused with the term Metadata Exchange Architecture (MX) which refers to a proprietary data warehousing product from the Informatica Corporation.

[3] Note that in this context we solely mean storage for exchange purposes as opposed to long-term storage.

consist of entities, attributes, data types, relationships and business rules. A brief description of each of these concepts is probably helpful.

- **Entities**

  More properly these should be referred to as "entity types," but the term "entity" is in common use. In data modelling terms, an entity is anything about which we wish to record data and which has attributes and/or relationships with other entities. An entity in a data model is equivalent to a persistent or storage class in an object model. When creating a data model for a particular business or business process, it is usually helpful to think of the entities as the nouns which occur in a narrative description of that business or process (i.e. entities represent tangible things).

- **Attribute**

  An attribute is a property of an entity about which data may be held. It is helpful to think of attributes as the elements that describe and define a particular entity. Still thinking of entities as broadly representing the nouns, attributes can be thought of as representing the adjectives. An entity representing a car might have attributes such as colour, engine size, model name and body type, for example.

- **Data types**

  A data type defines the way in which the value of an attribute should be physically stored. The data type defines the type of data an attribute represents (whether it is a character-based format, numeric, codified, a date and so on). This is of importance where the model is to be used as the basis for an IT system but is of less importance if the model is a purely semantic representation.

- **Relationship**

  In data modelling, a link between two entities is referred to as a relationship. This will generally have bi-directional names, and an indication of the optionality and cardinality (see below) involved at both ends. There should also be a definition of the relationship that states its purpose and includes examples. In object modelling, the same concept exists but is generally known as an "association". In many ways, relationships can be thought of as verbs in the same way that entities and attributes can be thought of as nouns and adjectives.

- **Business rules**

  Many data models contain business rules that lay down the exact nature of the way in which the data in a system interacts. Relationships are examples of a business's rules being represented on a data model. Similarly, the specific clusters of attributes, as part of an entity or class, reflect the sets of information that an organization expects to be held about something.

- **Optionality**

  This indicates the minimum number of entity instances that may occur for each end of a relationship. Although some tools and techniques do allow for specific values to be entered, the normal values are either zero or one. This indicates that the relationship is either optional or mandatory.

- **Cardinality**

  This indicates the maximum number of entity instances that may occur for each end of a relationship. Although some modelling tools and techniques do allow for specific values to be entered, the normal values are either one or many.

Until recently, a data model in IS-industry terms was any graphical representation of entities, relationships and attributes. Data models were either "logical" or "physical" with the former representing the semantic or business meaning and the latter acting as a step along the way to a physical table design. A logical model displays data grouped into entities in a manner that represents the way in which the data should ideally sit together, based solely on a business view. It does not take into account any of the constraints that an operating database

Abbreviations

EDI	Electronic Data Interchange	**RUP**	Rational Unified Process
IS	Information Systems	**UML**	Unified Modelling Language
MXF	Material eXchange Format	**XML**	Extensible Markup Language

management system might impose. This is reflected in the physical data model that reflects how data is arranged or implemented on a physical IT system. A physical model will also show entities, attributes and relationships but they will directly map to the tables, columns and keys in the system.

The rise of object-oriented modelling techniques in recent years has confused the exact meaning of the term "data model". The Rational Unified Process (RUP) and the Unified Modelling Language (UML) have become the dominant forces in object modelling and these techniques have gradually eroded the position of traditional data modelling. In the RUP and UML worlds, the term "data model" specifically applies to physical table designs. This represents a step along the way to building a specific physical application from an OO-class model. It should be noted that the equivalent steps of developing a logical (and therefore primarily semantic) class model, before moving to a physical data design, will still apply.

Is the P_META semantic scheme a data model?

The published version of P_META shares some features of some types of data model, but is not a data model in the normally accepted sense of the phrase, as used in the IS industry. Obviously, as a purely semantic scheme, P_META does not equate to a physical data model, or a data model in the sense meant by RUP/UML. P_META is a semantic scheme of data definitions available for business-to-business exchange. It contains no business rules, since the simple process of exchange (sending and receiving) does not imply any. The rules are specific to each exchange transaction and are agreed by the participants to that exchange. Therefore, there can be no entity definitions or data model relationships / associations included in the semantic scheme, since these are out of scope.

P_META has attributes similar to a data model, but they are not grouped together as entities and they are not held together by formal relationships and business rules. This is because the purpose of P_META is solely to facilitate common understanding between organizations for the purposes of transferring metadata from one to another. To achieve this, the attributes in P_META can be grouped together to form sets for specific exchange purposes. However, P_META is not concerned with data storage or persistence (which should support the business rules of the organization involved) and thus shies away from laying down any more rigid structures.

It is possible to create an entity relationship model with a full range of entities, attributes and relationships which would have the same data coverage as the current P_META scheme. However, the business rules being modelled in the entities and relationships may or may not be appropriate to any particular business. Certainly they would represent a complicated subset of the data model that would be required. Thus, such a model would present difficulties for many broadcasters because it would involve making assumptions about business rules and relationships that do not necessarily apply to all participating organizations.

Furthermore, the data model approach has the potential to be very divisive, as inevitably any standardized model would be similar to the models used internally by some organizations whilst being radically different from those used by others. Tensions might therefore arise over the extent of work required to create interfaces between the standardized model and each organization's own internal systems.

The method adopted by P_META – i.e. the use of a flat list of attributes and contextual sets – is felt to have the advantage of being less prescriptive and more flexible for users. Building an interface to P_META is likely to be easier for most organizations than building an interface from one rigid data structure to another, as individual data items can be more easily manipulated.

The precedents for the P_META approach are twofold.

- On the one hand, attempts have previously been made in a number of industries to have a common data architecture. Major players such as IBM and Unisys have attempted to build and sell generic data architectures for use across industries such as banking, insurance and petrochemicals. These architectures have failed because they cannot readily interface with the existing architectures of individual organizations, they lose clarity through over-generalisation and they cannot accurately represent the full range of business rules in use and the circumstances which apply across an entire industry.
- On the other hand, there is a long tradition of passing information between organizations using agreed data templates which do not require the support of full-blown data models. The latest and most popular

form of this kind of thing is XML, which lends itself very well to adopting the published P_META set structures. The use of XML, as a means of transferring metadata, follows in the well-established practice of Electronic Data Interchange (EDI).

Electronic Data Interchange

Electronic Data Interchange is *"the exchange of documents in standardized electronic form, between organizations, in an automated manner, directly from a computer application in one organization to an application in another"* [4]. It is most easily understood as a replacement of paper-based purchase orders with electronic equivalents, but is actually much broader in its application than the procurement process. The impact of EDI is thus far greater than mere automation. EDI offers the prospect of easy and cheap communication of structured information throughout the corporate community, and is capable of facilitating much closer integration among hitherto remote organizations.

History of EDI

By the standards of the IS industry, EDI has an extremely long and well-established history. The ideas which form the roots of EDI stretch back beyond the invention of computers to the nineteenth century when advances in railways, shipping and telegraphs meant that data needed to be collected and transmitted according to set patterns (for example in freight manifests). Electronic transmission commenced in the USA during the 1960s, initially in the rail and road transport industries. The standardization of documents was a necessary concomitant to that change. In 1968, the US Transportation Data Coordinating Committee (TDCC) was formed to coordinate the development of translation rules among four existing sets of industry-specific standards. Further significant moves towards standardization came with the X12 standards of the American National Standards Institute (ANSI), and work done by SITPRO (the British Simplification of Trade Procedures Board), the United Nations Economic Commission for Europe (UNECE) and a United Nations Joint European and North American working party (UN-JEDI), which ultimately resulted in the development of the EDIFACT (Electronic Data Interchange for Administration, Commerce and Transport) document translation standards.

Architecture for EDI

EDI can be compared and contrasted with electronic mail (e-mail). E-mail enables free-format textual messages to be electronically transmitted from one person to another. EDI, on the other hand, supports structured business messages (those which are expressed in hard-copy, pre-printed forms or business documents), and transmits them electronically between computer applications, rather than between people. In this sense, an XML implementation of P_META is a classic EDI application.

The essential elements of EDI are:

- the use of **an electronic transmission medium** (originally a value-added network, but increasingly the open public Internet) rather than the despatch of physical storage media such as magnetic tapes and disks;
- the use of **structured, formatted messages based on agreed standards** (such that messages can be translated, interpreted and checked for compliance with an explicit set of rules);
- **relatively fast delivery** of electronic documents from sender to receiver (generally implying receipt within hours, or even minutes); and
- **direct communication between applications** (rather than merely between computers).

EDI depends on a moderately sophisticated information technology infrastructure. This must include: (i) data processing, data management and networking capabilities, to enable the efficient capture of data into elec-

[4] Roger Clarke, Visiting Fellow, Dept Of Computer Science, Australian National University in Electronic Data Interchange (EDI): An Introduction, Canberra, 1998.

tronic form; (ii) the processing and retention of data; (iii) controlled access to this data and (iv) efficient and reliable data transmission between remote sites.

A common connection point is needed for all participants, together with a set of electronic mailboxes (so that the organizations' computers are not interrupted by one another), and security and communications management features. It is entirely feasible for organizations to implement EDI directly with one another, but it can be advantageous to use a third-party network services provider.

Conclusions

Although the P_META Scheme shares some features with a traditional logical data model, it cannot be described as a data model and does not aspire to be one. Previous experiences in other industries suggest that an EDI type of approach is more likely to be successful in the development of industry-wide metadata exchange standards than following a rigid data-model-based approach.